# BAB II
# TINJAUAN PUSTAKA DAN DASAR TEORI

## 2.1 Tinjauan Pustaka

Pada penelitian-penelitian yang sudah dilakukan mengenai penerapan JST *Backpropagation* dalam melakukan prediksi dan algoritma *ReliefF* dalam pemilihan penyaringan fitur evaluasi individu telah menghasilkan hasil yang baik. Adapun penelitian lain yang dilakukan menggunakan algoritma *Relief* sebagai *feature selection* dengan metode klasifikasi lainnya selain JST. Metode selain JST seperti contohnya *Logistic Regression* (*RegLog*), SVM dan lain sebagainya.

Chiueng-Fang Wu (2017) melakukan penelitian dengan menggabungkan dua metode yaitu *chi-square* untuk *feature selection* dengan model MLP dan *Backpropagation learning rule* dalam memprediksi ketahanan hidup bebas penyakit (DFS-*disease-free survival*) 1 tahun dan 2 tahun dari pasien HCC yang menerima perawatan *Radiofrequency ablation* (RFA). Sebanyak 15 variabel klinis kanker hati digunakan pada model prediksi DFS jaringan syaraf tiruan. Hasil penelitian menunjukkan parameter penilaian kinerja prediksi DFS 1 tahun adalah sebagai berikut: akurasi 85.0% (70.0%), sensitivitas 75.0% (63.3%), spesifisitas 87.5% (71.8%), dan area di bawah kurva 0.84 (0.77) untuk validasi internal (validasi prospektif simulasi). Untuk prediksi DFS 2 tahun, nilai akurasi, sensitivitas, spesifisitas, dan area di bawah kurva masing-masing adalah 67,9% (63,9%), 50,0% (56,3%), 85,7% (70,0%), dan 0,75 (0,72), untuk validasi internal (simulasi prospektif validasi) (Wu dkk., 2017).



Yanfeng Wang, dkk. (2020) melakukan penelitian pada tumor ganas yaitu *Esophageal squamous cell carcinoma* (ESCC). Penelitian ini bertujuan untuk mengetahui faktor-faktor yang mempengaruhi ketahanan hidup pasien. Peneliti menggunakan *Cox regression univariate analysis* untuk menganalisis indeks darah dan *Spearman and Pearson Correlation Analysis* untuk skrining faktor analisis ketahanan hidup. Adapun metode untuk analisis *survival* menggunakan penggabungan berapa metode, yaitu *Standard Salp Swarm Optimization Algorithm* dan BP *neural network*. Penelitian menghasilkan model ASSA-BP

meningkatkan akurasi prediksi, lebih efektif dalam memprediksi waktu ketahanan hidup pasien ESCC, dan mempersingkat waktu pelatihan (Wang dkk., 2020).

Dewi Nasien, dkk. (2021) melakukan penelitian pada kanker payudara untuk mengkategorikan kanker jinak dan ganas. Peneliti menggunakan metode *Backpropagation*. Tujuan dari penelitian ini adalah untuk memecahkan masalah kompleks dalam identifikasi, prediksi pengenalan pola, menyelidiki tingkat akurasi, dan kinerja BP dalam prediksi kanker payudara. Penelitian menggunakan simulasi dari MATLAB R2016a. Hasil dari penelitian ini menunjukan akurasi 96% dengan kombinasi parameter pelatihan dengan *epoch* 1000, *learning rate* 0.01, *goal* 0.001, dan *hidden layer* 5 (Nasien dkk., 2022).

Pada penelitian untuk memprediksi kanker endometrium pada wanita *pascamenopause* dengan perdarahan vagina atau ketebalan endometrium end 5 mm, yang ditentukan dengan pemeriksaan ultrasonografi, peneliti mengkomparasikan tiga metode yaitu *logistic regression*, ANN dan CART. Penelitian menghasilkan *logistic regression* memiliki sensitivitas 76,4%, spesifisitas 66,7%, dan akurasi keseluruhan (OA) 72,5%. ANN memiliki sensitivitas 86,8%, spesifisitas 83,3%, dan OA 85,4%. CART memiliki sensitivitas 78,3%, spesifisitas 76,4%, dan OA 77,5%. Dari ketiga metode yang digunakan, dapat disimpulkan bahwa ANN memiliki hasil akurasi yang lebih unggul dari dua metode lain dengan akurasi keseluruhan 85.4% (Pergialiotis dkk., 2018).

Ada beberapa penelitian yang telah menggunakan algoritma *Relief* sebagai *feature selection* dengan beberapa metode klasifikasi. Salah satunya untuk mengenali batu bara dan klorit berdasarkan gambar. Penelitian ini menggunakan dua metode dalam dua tahap, yaitu pada tahap pertama fitur dihapus satu per satu berdasarkan bobot dalam urutan menaik menggunakan *Relief*. Tahap selanjutnya menggunakan metode SVM untuk mengetahui akurasi klasifikasi dari set data pengujian yang digunakan untuk menentukan jumlah fitur yang dipertahankan. Pengklasifikasian menggunakan sampel batubara dari tambang batubara Dafeng dan Baijigou menghasilkan akurasi rata-rata terkait dengan 92,57% dan 92% (Dou dkk., 2019).

Pada tahun yang sama, Turker melakukan penelitian untuk mengklasifikasikan penyakit HCC. Penelitian yang dilakukan berfokuskan pada *missing feature completion, feature reduction* dan *feature classification phases* dan mengkomparasikan dua algoritma Neighborhood Component Analysis (NCA) dan *ReliefF*. Pada tahap klasifikasi, metode klasifikasi yang digunakan adalah metode conventional machine learning yang ada pada learner toolbox MATLAB. Hasil akurasi terbaik untuk metode berbasis NCA dan *ReliefF* dihitung masing-masing sebesar 92,12% dan 83,03% (Tuncer & Ertam, 2020).

Penelitian lain menggunakan *Relief* sebagai *feature selection*, dilakukan pada objek gambar *X-ray* sehat dan gambar *X-ray* terdiagnosis Covid-19. Metode klasifikasi yang digunakan adalah lima metode, yaitu *Decision tree (DT), Linear Discriminant (LD), K-nearest neighborhood (kNN), Support Vector Machines (SVM) dan Subspace Discriminant (SD)*. Metode *hybrid* yang diusulkan adalah ekstraksi fitur oleh ResExLBP dan metode pemilihan fitur *Relief*. Fitur yang dipilih diklasifikasikan oleh lima pengklasifikasian dengan metode *Leave one out cross-validation* (LOOCV), *cross-validation* (CV) 10 kali lipat, dan validasi pisahan. Hasil dari penelitian menunjukkan akurasi klasifikasi 99,69% dan 100,0% dengan menggunakan SVM dengan LOOCV dan CV 10 kali lipat (Tuncer dkk., 2020).

Penelitian lain dengan tujuan mengembangkan *fault diagnosis model* (FDM) yang efisien, menggunakan algoritma *ReliefF* untuk peringkat fitur dan menerapkan JST untuk diagnosis kesalahan. Penelitian dilakukan dengan langkah pertama, model JST dibuat di atas subset data fitur terbaik N dan dioptimalkan oleh algoritma *Bayesian regularization*. Langkah kedua, model sebelumnya diverifikasi dengan cara menguji subset data, untuk mendapatkan *correct diagnosis rates* (CDR). Hasil penelitian menunjukkan bahwa CDR FDM berdasarkan 6 fitur terbaik hasilnya cukup tinggi dibandingkan dengan 22 fitur dan waktu pelatihan berkurang hingga 98,8% (Shi dkk., 2017).

Sebuah penelitian telah dilakukan dengan tujuan untuk mempelajari hubungan potensial antara *Parkinson's disease* (PD) dan *scans without evidence of dopaminergic deficit* (SWEDD). Penelitian ini menggunakan algoritma *Relief*

sebagai *feature selection* dan algoritma SVM sebagai metode klasifikasi. Hasil menunjukkan bahwa hasil klasifikasi baik dengan tingkat akurasi tertinggi 81,25% (Jin dkk., 2019).

## 2.2 Dasar Teori

### 2.2.1 *Hepatocellular Carcinoma* (HCC)

*Hepatocellular Carcinoma* (HCC) adalah penyakit yang berbahaya pada organ hati. Penyakit ini umumnya terjadi pada pasien yang memiliki penyakit hati kronis atau sirosis hati (Książek dkk, 2019). Virus hepatitis kronis dapat menyebabkan sirosis dan kanker hati (HCC). Hepatitis B dan C adalah penyebab paling umum dari hepatitis kronis di dunia. Penderita hepatitis B memiliki risiko 10%-25% dalam perkembangan HCC. Hepatitis B dapat ditularkan melalui suntikan intravena, transfusi darah yang terkontaminasi, kontak seksual, dan dari ibu ke janin. Penderita Hepatitis C (HCV) memiliki risiko berkembang menjadi hepatitis kronis sebesar 80% dan 20% akan berkembang menjadi sirosis. Selain penyakit hati kronis dan sirosis hati, asupan alkohol yang berlebihan menjadi faktor risiko terpenting lainya pada pengembangan kanker hati. Hubungan antara asupan alkohol dengan penyakit hati berkorelasi terhadap jumlah alkohol yang dikonsumsi seumur hidup. Penggunaan alkohol yang berlebihan menjadi risiko utama kanker hati (HCC) (G. dkk., 2017).

### 2.2.2 Risk Factor HCC

Ada beberapa r*isk factor* atau faktor risiko dari *Hepatocellular Carcinoma* (HCC). Faktor risiko yang paling utama di seluruh dunia adalah penyakit hati kronis dan sirosis yang merupakan perkembangan dari virus hepatitis dan akibat dari berlebihannya konsumsi alkohol. Faktor risiko HCC tersebut juga diakibatkan oleh beberapa faktor risiko seperti berikut (G. dkk., 2017):

1. Asupan alkohol berlebihan
2. Virus Hepatitis B (HBV) dan Hepatitis C (HCV)
3. Diabetes
4. Obesitas

5. Asupan Steroid Anabolik
6. *Aflatoxin* dari *Aspergillus* (jamur)
7. Penyakit Metabolik dan Genetik (*Hemochromatosis, Wilson's Disease, α-1 Antitrypsin Disease, Tyrosinemia, Glycogen-Storage Disease Types I* dan *II, porphyrias,* dan *thalassemia*)
8. Merokok
9. Kontrasepsi Oral (konsumsi >5 tahun)

Dari beberapa faktor risiko di atas akan berkembang menjadi penyakit hati kronis dan sirosis yang memiliki risiko dalam perkembangan HCC (G. dkk., 2017).

### 2.2.3 *Feature Selection*

*Feature selection* (FS) merupakan proses mengidentifikasi fitur relevan dan membuang fitur yang tidak relevan. Metode FS telah dikarakterisasi berdasarkan pemilihan bergantung pada skor yang ditetapkan ke fitur individu atau sebagai sebuah subset kandidat dari fitur (Urbanowicz dkk., 2018). Pendekatan FS dibagi menjadi tiga kategori, yaitu sebagai berikut:

1. *Filter Method*

Metode *filter* memiliki waktu pemrosesan jauh lebih cepat dan berfungsi secara independen dari algoritme induksi. Hasil dari fitur yang dipilih kemudian diteruskan ke algoritme pemodelan lainnya. Metode *filter* menggunakan *'proxy measure'* yang dihitung dari karakteristik umum data *training* untuk menilai fitur atau fitur subset sebagai langkah pemrosesan sebelum pemodelan. Metode filter secara kasar dapat didefinisikan sebagai langkah-langkah penyaringan untuk mendapatkan informasi, jarak, ketergantungan, konsistensi, kesamaan, dan ukuran statistik. Contoh dari metode ini adalah chi-square dan *Relief* (Urbanowicz dkk., 2018).

2. *Wrapper Methods*

Metode *wrapper* menggunakan algoritma pemodelan yang berdiri sendiri untuk melatih model prediktif yang menggunakan subset fitur kandidat. Metode ini pemrosesannya secara berulang dan intensif secara komputasi dan dapat

mengidentifikasi fitur set dengan performa terbaik untuk algoritme pemodelan tertentu. Setiap iterasi *wrapper*, subset fitur dibuat berdasarkan strategi pencarian yang dipilih, seperti *forward, backward* atau *heuristic* fitur pemilihan subset (Urbanowicz dkk., 2018).

3. *Embedded Methods*

Metode *embedded* melakukan pemilihan fitur sebagai bagian dari eksekusi algoritma pemodelan. Metode ini cenderung lebih efisien secara komputasi daripada metode *wrapper* karena mereka secara bersamaan mengintegrasikan pemodelan dengan pemilihan fitur. Contoh metode *embedded* adalah *Lasso, Elastic Net,* dan berbagai algoritma berbasis *decision tree* (Urbanowicz dkk., 2018).

Selain ketiga kategori yang sudah disebutkan, ada metode *hybrid* yang merupakan kombinasi dari beberapa metode untuk mendapatkan hasil yang unggul. Metode *hybrid* inilah yang sekarang sedang banyak diusulkan oleh peneliti.

### 2.2.4 *ReliefF*

Algoritma *Relief* terinspirasi oleh pembelajaran *instance-based*. Kira dan Rendell mengusulkan *Relief* pada tahun 1992. Algoritme *Relief* merupakan satu-satunya algoritma filter evaluasi individual yang mampu mendeteksi ketergantungan fitur. Algoritma ini tidak mencari melalui kombinasi fitur, tetapi menggunakan konsep tetangga terdekat untuk mendapatkan statistik fitur yang secara tidak langsung menjelaskan interaksi (Urbanowicz dkk., 2018). *ReliefF* merupakan peningkatan dari *Relief* untuk *feature selection* (Tuncer & Ertam, 2020). *Relief* dirancang untuk menangani masalah dua kelas tanpa *missing value* dan sensitif terhadap *noise data* sedangkan *ReliefF* dirancang untuk dapat menangani *incomplete data, noise data* dan mengevaluasi kualitas fitur pada masalah *multi-class* (Banerjee, 1985).

*Pseudo-code* dari algoritma *ReliefF* sebagai berikut:

**Input:** M contoh pembelajaran (*learning instance*) $x_k$ (N fitur dan kelas C); Probabilitas kelas $p_y$; Parameter pengambilan sampel *m*; Jumlah *n* *instance* terdekat dari setiap kelas;

**Output:** untuk setiap fitur $F_i$ bobot kualitas $-1 \leq$ W $[i] \leq 1$;

**for** i=1 **to** N **do** W[i] = 0.0; **end for**;

**for** l = 1 **to** m **do**

secara acak pilih *instance* $x_k$ (dengan kelas $y_k$);

**for** y = 1 **to** C **do**

temukan n *instances* terdekat x[*j*,*y*] dari kelas *y*, *j* = 1..*n*;

**for** i=1 **to** N **do for** j = 1 **to** n **do**

**if** *y* = $y_k$ {*hit* terdekat?}

**then** W[i] = W[i] – diff(i,$x_k$,x[*j*, *y*])/(m * n);

**else** W[i] = W[i] + $p_y$/(1 − $p_{yk}$ ) * diff(i,$x_k$,x[*j*, *y*])/(m * n);

**end if**;

**end for**; { *j* } **end for**; { *i* }

**end for**; { *y* }

**end for**; { *l* }

return(W);

---

- *Missing feature values*: *ReliefF* dapat menangani data yang tidak lengkap atau *incomplete data* dengan menggeneralisasi fungsi *diff* yang berfungsi sebagai penghitung probabilitas dua *instance* nilai berbeda dari fitur yang telah diberikan. Berikut persamaannya:

-> Satu *instance* ($x_l$) dengan nilai fitur yang tidak diketahui $F_i$:

$$diff\ (F_i x_l x_k) = 1 - p\left(F_i = x_{k,i} | y = y_l\right) \quad (2.1)$$

-> Dua *instance* dengan nilai fitur yang tidak diketahui:

$$diff\ (F_i x_l x_k) = 1 - \sum_{j=1}^{n_i} (p\ (F_i = j | y = y_l) \times p\ (F_i = j | y = y_k)) \quad (2.2)$$

- *Noisy Data*: Pencarian *hit* dan *miss* terdekat pada *Relief* merupakan bagian terpenting sehingga adanya *noise* dalam kelas (*class*) dan nilai fitur secara

signifikan akan memengaruhi pemilihan hit dan miss terdekat. *ReliefF* menggunakan *n hit* dan *miss* terdekat dan rata-rata kontribusinya pada perkiraan kualitas fitur. *n* merupakan parameter yang ditentukan pengguna dengan nilai tipikal n ∈ [5 ... 10].

- *Multi-class problems*: *ReliefF* mencari n *instance* terdekat dari setiap kelas. Kontribusi kelas berbeda berbobot dengan probabilitas sebelumnya. Faktor pembobotnya adalah $p_y$ / (1−$p_{yk}$). Kelas sebuah *instance* adalah $y_k$, sedangkan y adalah kelas *miss* terdekatnya. Faktor tersebut oleh karenanya sebanding dengan probabilitas kelas *y*, dinormalisasi dengan jumlah probabilitas semua kelas, berbeda dari $y_k$ (Banerjee, 1985).

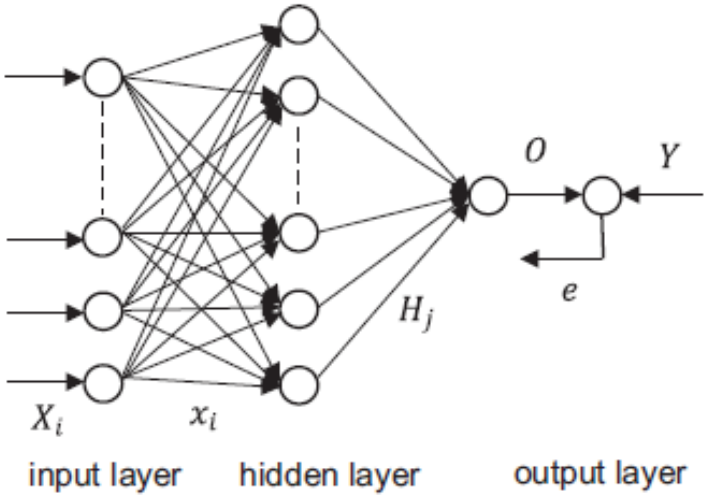
### 2.2.5 Jaringan Syaraf Tiruan

Jaringan Syaraf Tiruan (JST) adalah model yang terinspirasi secara biologis yang dapat belajar melalui contoh. Terdiri dari neuron buatan yang disusun berlapis-lapis dan saling berhubungan dengan bobot sinaptik, JST dapat memperoleh pengetahuan berdasarkan adaptasi berat selama proses pelatihan. Adopsi JST untuk memecahkan masalah tertentu melibatkan dua fase utama: pelatihan dan operasi. Pada awalnya, bobot sinaptik jaringan diadaptasi untuk menyelesaikan tugas tertentu, mengekstraksi pengetahuan dari data pelatihan. Setelah fase pelatihan selesai, JST beroperasi hanya menghasilkan output berdasarkan pada pengetahuan yang tersimpan (fase operasi) (João dkk., 2016).

### 2.2.6 *Backpropagation*



*Backpropagation Neural Network* (BPNN) merupakan jaringan neural feedforward multilayer yang terdiri dari lapisan input (*input layer*), lapisan implisit (*implicit layer*), dan lapisan output (*output layer*) yang ditunjukan pada Gambar 1 (Ye dkk., 2019). BPNN didasarkan pada algoritma pelatihan *Backpropagation* kesalahan. BPNN menggunakan pencarian *gradien* untuk meminimalkan kesalahan dari output *actual* dan output *expected* (Ling dkk., 2020).




Gambar 2.1 Lapisan *Backpropagation* Neural Network (Ling dkk., 2020)

Proses BPNN seperti yang dilihat pada Gambar 1. Pertama, sinyal merambat ke depan, dari lapisan input melalui lapisan tersembunyi ke lapisan output. Langkah selanjutnya kesalahan dihitung, bergerak dari lapisan keluaran melalui lapisan tersembunyi ke lapisan input. Jaringan saraf mencapai bobot dan ambang batas optimal setelah proses pelatihan berulang untuk mengurangi kesalahan ke tingkat yang diinginkan (Ling dkk., 2020). Pada proses *Multilayer perceptron* (MLP), output dari satu lapisan menjadi input di lapisan berikutnya. Neuron di lapisan pertama menerima input eksternal, dan neuron di lapisan terakhir menampilkan output dari jaringan (Vishwakarma dkk., 2020).

Proses BPNN terdapat tiga tahap, yaitu *Feedforward* untuk input pola pelatihan (*training pattern*), *Backpropagation* untuk *associated error* dan penyesuaian bobot (*adjustment weight*) (Fausett & Fausett, 1994).



1. *Feedforward*

   Persamaan yang digunakan dalam tahap ini adalah,

   $$z\_in_j = v_{0j} + \sum_i x_i v_{ij} \quad (2.3)$$

   Keterangan:

   $i$ : *neuron* ke-$i$ ($i$=1,2,3, …, $n$) pada *input layer* (lapisan input).

   $j$ : *neuron* ke-$j$ ($j$=1,2,3, …, $p$) pada *hidden layer*.

   $v_{0j}$: bias pada *input layer*.

$x_i$ : bobot pada *i*.

$v_{ij}$: nilai *input* pada *i* ke *j*.

Persamaan fungsi aktivasi:

$$z_j = f(z\_in_j) \quad (2.4)$$

Keterangan:

$f(z\_in_j)$ : nilai aktivasi pada *hidden layer* ke *output.*

2. *Backpropagation*

Persamaan yang digunakan dalam tahap ini adalah,

$$\delta_k = (t_k - y_k) f'(y\_in_k) \quad (2.5)$$

Keterangan:

$\delta_k$ : informasi kesalahan.

$y_k$ : unit *output*.

$t_k$ : pola target.

3. *Adjustment weight*

Persamaan yang digunakan dalam tahap ini adalah,

$$w_{jk}(new) = w_{jk}(old) + \Delta w_{jk} \quad (2.6)$$

$$v_{ij}(new) = v_{ij}(old) + \Delta v_{ij} \quad (2.7)$$

Keterangan:

$w_{jk}(new)$: bobot baru antara *input layer* dan *hidden layer* yang akan dicari.

$w_{jk}(old)$ : bobot lama yang diperbaharui.

$v_{ij}(new)$ : bobot baru antara *hidden layer* dan *output layer* yang akan dicari.

$v_{ij}(old)$ : bobot lama yang diperbaharui.



### 2.2.7 Confusion Matrix

*Confusion matrix* merupakan pendekatan untuk memvalidasi akurasi klasifikasi. *Confusion matrix* menyajikan informasi tentang seberapa sering perilaku tertentu dideteksi dengan benar dan seberapa sering perilaku tersebut diklasifikasikan sebagai perilaku lain. Berikut tabel *confusion matrix* terlihat pada Tabel 2.1.

Tabel 2.1 *Confusion Matrix*

| ***Actual Class*** | | ***Assigned Class*** | |
|---|---|---|---|
| | | ***Positive*** | ***Negative*** |
| | ***Positive*** | TP | FN |
| | ***Negative*** | FP | TN |

Keterangan:

TP : *True Positives*

FN : *False Negative*

FP : *False Positive*

TN : *True Negative*

Pada perhitungan indikator kinerja klasifikasi untuk mencerminkan bagaimana kinerja pengklasifikasi dalam mendeteksi kelas yang diberikan dibutuhkan *precision*, *sensitivity*, *specificity*, dan *accuracy*. Berikut persamaan keempat indikator (Shultz dan Fahlman, 2017).

$$Precision = \frac{TP}{(TP + FP)} \quad (2.8)$$

$$Recall = \frac{TP}{(TP+FN)} \quad (2.9)$$

$$Accuracy = \frac{TP+TN}{TP + TN + FP + FN} \quad (2.11)$$