BAB II
TINJAUAN PUSTAKA

Pada bab ini menjelaskan mengenai tinjauan pustaka yang menjadi dasar dalam penelitian ini. Tinjauan pustaka meliputi metode yang digunakan yaitu Amelia II, CART, dan Regresi Logistik. Di bab ini juga dijelaskan mengenai penelitian terdahulu serta kelebihan dan kekurangan dari metode yang digunakan.

2.1 Penelitian Terdahulu

Teknik klasifikasi dalam data mining seperti *Support Vector Machine (SVM)*, *Neural Network,* Regresi Logistik dan Decision Tree sudah sering digunakan untuk membangun model prediktif. Teknik klasifikasi *SVM* dan *Neural Network* pernah dilakukan oleh Khandani et al. [2] untuk membuat model risiko kredit konsumer. Namun, kelemahan dari kedua metode ini adalah outputnya yang sulit untuk dimengerti.

Regresi logistik dan decision tree adalah dua algoritma yang paling populer digunakan dalam penelitian dan memiliki performa yang baik [3] . Neslin Et al. [4] mengkategorikan beberapa pendekatan yaitu, “Logit”, “Trees”, “Discriminant”, dan “Explain” kemudian membandingkannya. Hasil penelitiannya menunjukkan bahwa “Logit” dan “Trees” merupakan dua pendekatan yang paling baik dalam hal ketepatan prediksi.

Metode regresi logistik adalah suatu metode analisis statistika yang mendeskripsikan hubungan antara variabel respon yang memiliki dua kategori atau lebih dengan satu atau lebih variabel penjelas berskala kategori atau interval.

Melalui metode ini akan dihasilkan peluang dari masing-masing kategori respon yang akan dijadikan sebagai pedoman pengklasifikasian dan suatu pengamatan akan masuk ke dalam respon kategori tertentu berdasarkan nilai peluang yang terbesar [5].

Decision Tree merupakan model klasifikasi yang berbentuk seperti pohon, dimana *Decision Tree* mudah dimengerti meskipun oleh pengguna yang belum ahli sekalipun dan lebih efisien dalam menginduksi data [6]. Kelebihan regresi logistik dibandingkan dengan *Decision Tree* adalah memiliki nilai rasio odds yang dapat menunjukkan seberapa besar pengaruh terhadap peubah respon dari suatu kategori dalam atribut yang dibandingkan dengan kategori referensinya

Pada kenyataannya sering terjadi data tidak lengkap *(missing value).* Decision Tree dapat menyesuaikan terhadap "noisy" data, termasuk *missing value*. Sedangkan Regresi Logistik tidak dapat menyesuaikan jika ada gangguan *missing value* [7] . Blankers et al [8] dalam penelitiannya membandingkan 4 perangkat lunak untuk imputasi *missing value* : NORM, MICE, Amelia II, dan SPSS MI. Hasilnya Amelia II menghasilkan akurasi yang lebih baik dibandingkan dengan ketiga perangkat lunak lainnya.

2.2 *Missing Value*

Missing Value dapat terjadi dikarenakan informasi tentang obyek sulit dicari, tidak diberikan atau informasi tersebut memang tidak ada. *Missing value* adalah masalah yang sering terjadi dalam penelitian. Adanya *missing value* pada data yang dianalisis dapat menurunkan keakuratan dan kualitas data pada saat

diolah. Pendekatan yang paling sederhana dalam menangani *missing value* adalah dengan metode *listwise deletion* dan *pairwise deletion.* Kedua metode ini menyarankan untuk menghapus observasi yang mengandung *missing value* dan tidak melibatkannya dalam analisis selanjutnya. Jika *missing value* hanya sedikit, menghapus data yang mengandung *missing value* tidak akan berpengaruh terhadap hasil analisis. Namun, jika *missing value* berjumlah banyak menghapus data yang mengandung *missing value* akan menyebabkan informasi terbuang. Oleh karena itu, *missing value* ini harus diimputasi.

Metode Imputasi dibagi menjadi 2 yaitu *Single Imputation* dan *Multiple Imputation.*

1. *Single Imputation*

   Single Imputation adalah mengisi *missing value* dengan nilai dugaan seperti *Mean Imputation*, *Cold Deck Imputation,* atau *Hot Dect Imputation.* Kekurangan dari *Single Imputation* adalah tergantung pada faktor- faktor yang tidak nyata, seperti urutan pada dataset.

2. *Multiple Imputation*

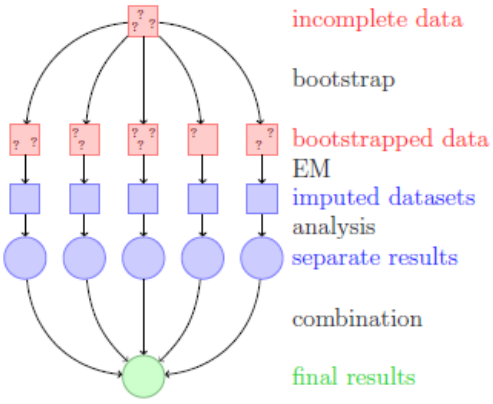
   Dengan keterbatasan dari *single imputation,* maka dikembangkan metode *multiple imputation.* Metode ini mengisi *missing value* sebanyak beberapa kali imputasi. Imputasi dilakukan maksimal sebanyak 5 kali kemudian akan dipilih nilai yang paling mendekati dengan nilai sebenarnya.

Sebagian besar metode analisis statistika mengasumsikan tidak adanya *missing value*, dan hanya bisa dijalankan apabila data yang diolah adalah data lengkap. Sehingga diperlukan metode untuk menangani masalah *missing value* ini

agar tidak kehilangan informasi dari berharga dari dataset yang akan dianalisis. Amelia II adalah perangkat lunak untuk mengimputasi *missing value* dari dataset yang tidak lengkap sehingga metode analisis yang memerlukan data lengkap dapat dengan tepat menggunakan seluruh informasi yang ada dalam data. Amelia II menggunakan algoritma *Expectation-Maximization with Bootstrapping* (EMB *bootstrap*) yang merupakan salah satu metode *multiple imputation*.

Amelia II memiliki beberapa keunggulan dibandingan dengan Amelia versi pertama. Amelia II dapat melakukan imputasi variabel dan data yang lebih banyak dengan waktu yang lebih singkat. Keunggulan ini dikarenakan algoritma EMB *bootstrap* yang digunakan dalam metode Amelia II ini. Amelia II juga memiliki fitur untuk melakukan imputasi yang valid dan jauh lebih akurat untuk data numerik dan kategorik. Selain itu, Amelia II juga menyediakan banyak fungsi diagnostik yang membantu memeriksa akurasi hasil imputasi [9].

Metode EMB *Bootstrap* melakukan sejumlah m imputasi untuk setiap *missing value* dalam matriks data dan membuat sejumlah m dataset lengkap. Imputasi dari setiap dataset dilakukan dengan algoritma EMB *bootstrap*. Seluruh m dataset lengkap kemudian dikombinasikan. Tahapan Amelia II dapat dilihat pada gambar II-1.



Gambar II-1 Skema EMB *Bootstrap* pada Amelia II

Sumber : Amelia II: A Program for Missing Data [9]

2.3 *Decision Tree*

Decision Tree adalah teknik klasifikasi yang menentukan setiap objek (dalam penelitian ini debitur) kedalam dua kelas yaitu "baik" dan "buruk" berdasarkan profil dari debiturnya. Output dari decision tree ini berbentuk seperti pohon, dimana *Decision Tree* mudah dimengerti meskipun oleh pengguna yang belum ahli sekalipun dan lebih efisien dalam menginduksi data. Proses pembentukan *Decision Tree* dibagi menjadi 3 yaitu (1) pembentukan pohon, (2) *pruning*, (3) mengekstrak aturan (rule) dari pohon keputusan yang terbentuk [10].

Beberapa algoritma yang telah dikembangkan berdasar *Decision Tree* adalah (1) CHAID (*Chi-Square Automatic Interaction Detection*) yang mana *split* tiap node berdasakan pada *Chi-Square test* pada masing-masing atribut, (2) CART *(Classification And Regression Tree)* yang membentuk *Decision Tree* dengan perhitungan *Gini Index* untuk kriteria *split* , (3) C4.5 yang merupakan variasi pengembangan dari ID3 menggunakan *Gain Ratio* untuk kriteria *split.* CHAID

dan C4.5 hanya bisa digunakan untuk data kategorik, sedangkan CART bisa digunakan untuk data numerik dan kategorik [11].

CART digunakan untuk mengelompokan data sebagai penciri dari suatu pengklasifikasian, selain itu CART juga dapat digunakan untuk menggambarkan hubungan antara peubah respon dengan satu atau lebih atributnya. CART dapat digunakan untuk data numerik dan data kategorik. Model pohon yang dihasilkan tergantung kepada peubah responnya, jika peubah respon berupa data numerik maka pohon yang dihasilkan adalah pohon regresi sedangkan jika peubah respon berupa data kategorik maka pohon yang dihasilkan adalah pohon klasifikasi [12].

CART hanya memisahkan variabel bebas menjadi dua kategori sehingga metode ini menghasilkan pohon keputusan yang mudah dibaca. CART mempunyai beberapa kelebihan dibandingkan metode klasifikasi lainnya, yaitu hasilnya lebih mudah diinterpretasikan, lebih akurat dan lebih cepat penghitungannya. Selain itu CART bisa diterapkan untuk himpunan data yang mempunyai jumlah besar, variabel yang sangat banyak dan dengan skala variabel campuran melalui prosedur pemilahan biner [13].

Penentuan atribut akan dijadikan simpul atau tidak dilihat berdasarkan keheterogenannya. Jika penambahan atribut untuk dijadikan simpul dapat menurunkan keheterogenan maka atribut tersebut akan dijadikan simpul. Sebaliknya jika penambahan simpul hanya menambah keheterogenan maka pembuatan pohon klasifikasi dihentikan. Langkah – langkah algoritma CART adalah sebagai berikut :

1) Pengkategorian Ulang Atribut

Pengkategorian ulang atribut yang akan digunakan dalam pembentukan pohon klasifikasi diukur menggunakan indeks gini. Indeks gini menunjukan tingkat keheterogenan dari suatu atribut yang dapat membantu dalam pengkategorian ulang nilai dalam atribut menjadi dua kategori. Fungsi indeks Gini adalah sebagai berikut :

$$i(t) = -\sum_{i \neq j} p(i|t)p(j|t) \quad (2.1)$$

Dimana p(i|t) adalah peluang i pada simpul t dan p(j|t) adalah peluang j pada simpul t. Pemilahan dilakukan untuk mereduksi tingkat keheterogenan yang didefinisikan sebagai berikut :

$$\emptyset(s,t) = \Delta i(s,t) = i(t) - p_L i(t_L) + p_R i(t_R) \quad (2.2)$$

Nilai $\Delta i(s,t)$ yang tinggi menunjukkan bahwa pembagian kategori mereduksi keheterogenan secara signifikan. Pembagian kategori yang menghasilkan nilai $\Delta i(s,t)$ yang paling tinggi adalah pemilah yang baik untuk digunakan dalam pembentukan pohon klasifikasi.

2) Seleksi Atribut

Suatu atribut akan diikutsertakan dalam pembentukan pohon klasifikasi jika atribut tersebut dapat menurunkan keheterogenan secara berarti.

Ukuran pohon klasifikasi optimal dapat dilihat dari nilai *Cost Complexity Minimum* yang didefinisikan sebagai berikut :

$$R_\alpha(t) = R(t) + \alpha|\tilde{T}| \tag{2.3}$$

dimana :

R(T) = *Resubstitution Estimate* (Proporsi kesalahan pada sub pohon)

α = Kompleksitas parameter

$|\tilde{T}|$ = Ukuran banyaknya atribut pada pohon T

Pohon klasifikasi dengan banyak atribut menyebabkan nilai kompleksitas tinggi. Pohon yang optimal adalah pohon dengan atribut yang sedikit namun memberikan nilai akurasi yang baik.

2.4 Regresi Logistik

Analisis Regresi Logistik adalah analisis yang digunakan untuk melihat hubungan antara peubah respon yang berupa data kategorik dengan atributnya yang berupa data kategorik maupun data numerik. Analisis Regresi Logistik merupakan teknik pemodelan statistika yang sering digunakan. Regresi Logistik dapat membangun model dengan keluaran dikotomis dan telah terbukti sebagai algoritma yang ampuh dalam pemodelan. Regresi logistik pernah diaplikasikan dalam penelitian di berbagai bidang seperti penelitian sosial, penelitian medis, prediksi kebangkrutan, segmentasi pasar, dan pola perilaku pelanggan [14] [15] [16] [17] [18].

Misalkan terdapat p atribut $X' = (X_1, X_2, \ldots., X_p)$ dengan peubah respon Y yang bernilai 1 dan 0 maka peubah respon Y mengikuti sebaran Bernoulli dengan parameter π(X) dengan sebaran fungsi peluang sebagai berikut :

$$f(Y_i) = \pi(X_i)^{yi}[1 - \pi(X_i)]^{1-yi} \quad (2.4)$$

Fungsi regresi logistik antara π(X) dan X adalah :

$$\pi(x) = \frac{\exp[g(X)]}{1+\exp[g(X)]} \quad (2.5)$$

Dengan $g(X) = (\beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p)$ merupakan fungsi logit.

Fungsi di atas berbentuk curvilinier sehingga untuk membuatnya menjadi fungsi linier harus dilakukan transformasi logit sebagai berikut :

$$logit\,[\pi(x)] = \ln\left[\frac{\exp[g(X)]}{1+\exp[g(X)]}\right] = g(X) \quad (2.6)$$

Kemudian pendugaan parameter ($\beta_0,\ \beta_1,\ \beta_2, \ldots,\ \beta_p$) dilakukan menggunakan Metode Kemungkinan Maksimum.

Interpretasi koefisien pada model regresi logistik dapat dilakukan dengan melihat nilai rasio oddsnya. Rasio odds adalah ukuran untuk melihat seberapa besar kecenderungan pengaruh peubah bebas terhadap peubah responnya. Nilai koefisien model logit pada regresi logistik $\beta_i = g(x+1) -$

$g(x)$ menunjukan perubahan logit g(x) untuk setiap perubahan satu unit atribut X yang disebut log odds. Nilai odds (rasio antara Y=1 dan Y=0) untuk X=1 adalah $\left[\frac{\pi(1)}{\{1-\pi(1)}\right]$ sedangkan untuk X = 0 adalah $\left[\frac{\pi(0)}{\{1-\pi(0)}\right]$. Rasio odds adalah pembagian nilai odds X=1 dengan nilai odds X=0 sehingga :

$$Rasio\ Odds = \frac{\left[\frac{\pi(1)}{\{1-\pi(1)}\right]}{\left[\frac{\pi(0)}{\{1-\pi(0)}\right]} = \exp(\beta_i) \tag{2.7}$$

Model Regresi yang baik adalah model yang memiliki peluang salah klasifikasi yang minimal. Ketepatan klasifikasi Regresi Logistik dapat dilihat dari nilai *Correct Clasification Rate* (CCR). CCR merupakan persentase ketepatan nilai dugaan dengan pengamatannya. Nilai CCR dapat dihitung dengan persamaan berikut :

$$CCR = \frac{banyaknya\ dugaan\ yang\ tepat}{banyaknya\ pengamatan}\ x\ 100\% \tag{2.8}$$

Nilai CCR akan dihitung menggunakan data testing. Semakin besar nilai CCR yang dihasilkan maka ketepatan klasifikasinya semakin tinggi.