BAB 2
LANDASAN TEORI

2.1 Penelitian Terdahulu

Sebagai landasan penelitian, dilakukan kajian terhadap berbagai penelitian terdahulu yang mengkaji penggunaan metode *machine learning* dalam prediksi keberhasilan kampanye *crowdfunding*. Rangkuman penelitian yang relevan dengan topik penelitian ini dapat dilihat pada Tabel 2.1.

Tabel 2.1. Ringkasan penelitian terdahulu

Penulis	Judul	Topik	Hasil
Aashay Shah, Prithvi Shah, Umang Savla, Yash Rathod, Nirmala Baloorkar	*Predictive Analysis of Crowdfunding Projects. In Disruptive Technologies and Digital Transformations for Society 5.0*	BERT/TF-IDF dan Random Forest	Random Forest mencapai akurasi mendekati 90% dalam klasifikasi keberhasilan kampanye.
Ramy Elitzur, Noam Katz, Peri Muttath, David Soberman	*The Power of Machine Learning Methods to Predict Crowdfunding Success: Accounting for Complex Relationships Efficiently*	Random Forest dan XGBoost	Metode *machine learning* secara signifikan mengungguli Nominal Logistic.
Lanjut pada halaman berikutnya			

Tabel 2.1 Ringkasan Penelitian Terkait Prediksi Keberhasilan Crowdfunding (lanjutan)

Penulis	Judul	Topik	Hasil
Jiayu Tian	*Do You Want to Foresee Your Future? The Best Model Predicting the Success of Kickstarter Campaigns*	Random Forest	Random Forest memperoleh akurasi 87,85% dan menjadi model dengan performa terbaik dibanding metode lain.
Somboon Prasobpiboon, Roongkiat R., Achara Chandrachai, Sipat Triukose	*Innovative Reward-Based Crowdfunding Decision Model*	Logistic Regression, Decision Trees	Logistic Regression mampu memprediksi keberhasilan kampanye *crowdfunding* dengan akurasi hingga 94,1% serta mengidentifikasi fitur-fitur penting yang memengaruhi kesuksesan proyek.
Indria Agustina, Yessi Mulyani, Trisya Septiana, Mardiana	Analisis Pengembangan Model Prediksi Kesuksesan Kickstarter Menggunakan Algoritma Backpropagation dan Random Forest	ANN (*Back-propagation*), Random Forest	Random Forest mencapai akurasi 98% dan F1-score 98%. Variabel *pledge* dan *backers* meningkatkan performa model secara signifikan.

Penelitian pertama yang berjudul "*Predictive Analysis of Crowdfunding Projects*" oleh Shah *et al.* membahas prediksi keberhasilan kampanye *crowdfunding* melalui kombinasi algoritma Random Forest dan BERT. Temuan dari penelitian tersebut mengungkapkan bahwa Random Forest dapat menghasilkan prediksi yang cukup baik dengan tingkat akurasi yang hampir mencapai 90% [12].

Penelitian kedua yang berjudul "*The Power of Machine Learning Methods to*

Predict Crowdfunding Success: Accounting for Complex Relationships Efficiently" oleh Elitzur *et al.* menerapkan pendekatan *machine learning* dengan memanfaatkan algoritma Random Forest dan XGBoost untuk memprediksi keberhasilan kampanye *crowdfunding*. Hasil penelitian menunjukkan bahwa model berbasis *machine learning* memiliki performa yang lebih baik dibandingkan model logit konvensional dalam melakukan prediksi keberhasilan kampanye. Dari algoritma yang dievaluasi pada penelitian ini, algoritma Random Forest mencatatkan nilai presisi sebesar 84% dan terbukti melampaui performa metode lainnya, seperti DNN dan *Nominal Logistic* [5].

Penelitian ketiga yang berjudul "*Do You Want to Foresee Your Future? The Best Model Predicting the Success of Kickstarter Campaigns*" oleh Jiayu Tian mengkaji prediksi keberhasilan kampanye Kickstarter dengan menggunakan algoritma Random Forest. Berdasarkan hasil eksperimen yang dilakukan, Random Forest mampu mencapai tingkat akurasi sebesar 87,85% dan menunjukkan performa yang lebih unggul dibandingkan algoritma lain yang dievaluasi dalam penelitian tersebut. Temuan ini mengindikasikan bahwa Random Forest merupakan model yang paling efektif untuk memprediksi keberhasilan kampanye Kickstarter pada dataset yang digunakan [6].

Penelitian keempat yang berjudul "*Innovative Reward-Based Crowdfunding Decision Model*" oleh Prasobpiboon *et al.* mengkaji penerapan metode *machine learning* pada data historis *crowdfunding* untuk mengidentifikasi faktor-faktor yang berkontribusi terhadap keberhasilan kampanye. Hasil penelitian menunjukkan bahwa algoritma Logistic Regression memiliki performa yang kompetitif dan stabil, dengan tingkat akurasi mencapai 94,1% pada pengujian eksternal. Selain itu, model tersebut mampu memberikan gambaran yang jelas mengenai pengaruh masing-masing fitur, sehingga dapat dimanfaatkan sebagai dasar dalam penyusunan strategi kampanye yang lebih efektif [11].

Penelitian kelima yang berjudul "Analisis Pengembangan Model Prediksi Kesuksesan Kickstarter Menggunakan Algoritma Backpropagation dan Random Forest" oleh Agustina *et al.* membahas pengembangan model klasifikasi biner untuk memprediksi keberhasilan kampanye Kickstarter dengan memanfaatkan algoritma Artificial Neural Network (ANN) berbasis *backpropagation* dan Random Forest. Penelitian tersebut menggunakan sebanyak 5.723 data kampanye Kickstarter pada kategori teknologi sebagai dataset penelitian. Hasil evaluasi pada penelitian tersebut memperlihatkan bahwa Random Forest menampilkan performa sebagai model terbaik dengan capaian akurasi dan *F1-score* yang

keduanya mencapai 98%. Selain itu, penelitian ini mengungkap bahwa variabel *pledge* dan jumlah *backers* memiliki peran yang penting dalam proses prediksi, yang ditunjukkan oleh penurunan performa model secara signifikan setelah kedua variabel tersebut dihilangkan [14].

2.2 Crowdfunding

Crowdfunding berasal dari konsep *crowdsourcing*, yaitu suatu proses pemanfaatan kontribusi dari sekelompok besar orang untuk memperoleh sumber daya tertentu. Penelitian sebelumnya menyatakan bahwa partisipasi kolektif dari kerumunan dapat menghasilkan kinerja yang lebih efisien dibandingkan dengan upaya yang dilakukan oleh individu secara terpisah [15]. *Crowdfunding* merupakan model pembiayaan yang sering disebut sebagai pendanaan demokratis, karena mekanismenya menghimpun dana dalam jumlah relatif kecil dari banyak individu hingga menghasilkan total pendanaan yang besar [16].

2.3 Kickstarter

Kickstarter merupakan salah satu media *crowdfunding* terbesar di dunia yang berfokus pada pendanaan proyek kreatif, seperti seni, desain, teknologi, film, dan permainan. Kickstarter menggunakan mekanisme *all-or-nothing* (AON), yang mendorong pembuat proyek untuk menetapkan target pendanaan yang masuk akal agar peluang keberhasilan lebih besar [13]. Selain itu, media Kickstarter termasuk ke dalam *reward-based* yang merupakan penggalangan dana yang memberikan sebuah imbalan atau hadiah sebagai apresiasi dalam proyek bisa berupa barang ataupun jasa [17]. Dukungan dari *backer* juga dipengaruhi oleh skema hadiah yang ditawarkan. Hal ini dikarenakan *backer* dapat menawarkan variasi pilihan penghargaan, termasuk adanya diskon atau jumlah terbatas, dapat meningkatkan kinerja kampanye. Namun demikian, harga penghargaan yang terlalu rendah atau terlalu tinggi justru dapat menurunkan tingkat keberhasilan, sehingga diperlukan penetapan harga yang seimbang [18].

2.4 Ensemble Learning

Ensemble learning adalah suatu pendekatan yang menggabungkan beberapa model *machine learning* guna menghasilkan prediksi yang lebih akurat dibandingkan apabila hanya mengandalkan satu model saja. Konsep utama metode

ini adalah membangun sejumlah model dasar (*base learner*) dan menggabungkan hasil prediksinya guna mengurangi varians, menekan bias, serta meningkatkan performa prediksi secara keseluruhan [19].

Salah satu metode dalam *ensemble learning* adalah *bootstrap aggregating* atau *bagging*. Metode ini bertujuan untuk mengurangi varians yang dimiliki model dengan cara melatih beberapa model pada subset data yang berbeda yang diperoleh melalui proses *bootstrap sampling*. Pendekatan tersebut nantinya dapat menghasilkan prediksi yang lebih stabil dan mengurangi sensitivitas model terhadap variasi data pelatihan [20]. Di samping itu, *bagging* dikenal tangguh dalam menghadapi *noise* karena galat yang dihasilkan oleh tiap model cenderung saling menetralkan satu sama lain, sehingga prediksi yang dihasilkan menjadi lebih robust [21]. Random Forest merupakan salah satu algoritma yang menerapkan teknik *bagging*.

Pendekatan lain dalam *ensemble learning* adalah *boosting*, yaitu metode yang membangun model secara bertahap dan berurutan, di mana setiap model baru difokuskan untuk memperbaiki kesalahan yang dihasilkan oleh model sebelumnya [20]. Teknik ini terbukti efektif dalam meningkatkan akurasi prediksi serta mampu menangani dataset yang tidak seimbang (*imbalanced data*) dengan baik [21]. Beberapa algoritma yang menerapkan pendekatan *boosting* antara lain Gradient Boosting dan XGBoost.

Selain *bagging* dan *boosting*, terdapat pula metode *stacking*, yaitu teknik yang menggabungkan prediksi dari beberapa model berbeda menggunakan sebuah *meta-learner* untuk menghasilkan prediksi akhir yang lebih optimal. Pendekatan ini memungkinkan pemanfaatan keunggulan dari berbagai algoritma secara bersamaan sehingga mampu menangkap pola data yang lebih kompleks [20]. Penelitian terdahulu menunjukkan bahwa *stacking* memiliki performa yang baik pada data berdimensi tinggi maupun permasalahan yang kompleks karena memanfaatkan karakteristik unik dari setiap model yang digunakan [22]. Dalam implementasinya, berbagai algoritma dapat digunakan sebagai *meta-learner*, termasuk Logistic Regression.

2.5 Logistic Regression

Logistic Regression adalah metode klasifikasi yang umum digunakan untuk memprediksi keluaran biner dengan memodelkan probabilitas suatu kejadian berdasarkan sekumpulan variabel independen. Metode ini memanfaatkan fungsi

logit untuk menggambarkan hubungan antara variabel prediktor dan probabilitas terjadinya suatu peristiwa [13]. Dalam prosesnya, kombinasi linier dari variabel independen ditransformasikan menggunakan fungsi sigmoid sehingga menghasilkan nilai probabilitas dalam rentang 0 hingga 1, yang sesuai untuk permasalahan klasifikasi biner [23]. Selain memiliki struktur yang relatif sederhana, Logistic Regression juga menghasilkan nilai probabilitas yang mudah diinterpretasikan, sehingga sering digunakan sebagai *baseline model* dalam berbagai penelitian klasifikasi, khususnya ketika hubungan antar fitur cenderung bersifat linier.

Meskipun Logistic Regression menawarkan tingkat interpretasi yang baik dan efektif digunakan pada data yang memiliki hubungan linier, metode ini memiliki keterbatasan dalam merepresentasikan pola hubungan non-linear yang kompleks. Kondisi seperti *threshold effects* maupun interaksi antar fitur sering kali sulit ditangkap secara optimal oleh model ini, terutama pada kasus prediksi keberhasilan kampanye *crowdfunding* [5]. Akibatnya, performa Logistic Regression pada beberapa permasalahan dapat berada di bawah metode *machine learning* yang lebih kompleks. Namun, berbagai penelitian menunjukkan bahwa dengan pemilihan fitur yang relevan dan pemanfaatan fitur tambahan dari sumber eksternal, Logistic Regression tetap mampu menghasilkan kinerja yang kompetitif dengan tingkat akurasi yang dapat mencapai 94,1% [11].

Dalam penerapan Logistic Regression, proses *feature scaling* umumnya dilakukan untuk meningkatkan stabilitas dan efektivitas selama tahap pelatihan model. Sejumlah penelitian menunjukkan bahwa teknik *scaling*, seperti standardisasi maupun normalisasi, dapat memberikan kontribusi positif terhadap performa Logistic Regression pada berbagai permasalahan klasifikasi [24]. Teknik ini bekerja dengan menyelaraskan rentang nilai pada fitur-fitur numerik sehingga setiap fitur memiliki skala yang lebih sebanding dan model dapat mempelajari pola dalam data secara lebih optimal.

2.6 Random Forest

Random Forest adalah metode *ensemble learning* yang dibangun berdasarkan konsep *bagging*, yaitu dengan membentuk sejumlah *decision tree* menggunakan sampel data hasil *bootstrap sampling* yang berbeda-beda. Pada setiap percabangan pohon, algoritma ini juga memilih sebagian fitur secara acak untuk menentukan pemisahan terbaik. Mekanisme tersebut memungkinkan Random

Forest mengurangi variasi prediksi yang sering muncul pada *decision tree* tunggal tanpa meningkatkan bias secara signifikan [13][25]. Dengan menggabungkan hasil prediksi dari banyak pohon keputusan, Random Forest mampu menghasilkan model yang lebih robust, stabil, dan memiliki risiko *overfitting* yang lebih rendah dibandingkan penggunaan satu *decision tree* saja.

Proses kerja Random Forest diawali dengan pembentukan sejumlah *decision tree* yang masing-masing dilatih menggunakan subset data berbeda yang diperoleh melalui teknik *bagging* [26]. Pada setiap tahap percabangan pohon, algoritma memilih sebagian fitur secara acak sebagai kandidat pemisahan, sehingga setiap pohon yang dihasilkan memiliki karakteristik yang berbeda dan tingkat korelasi yang rendah satu sama lain [26]. Selanjutnya, proses pemilihan pemisahan terbaik dilakukan dengan memanfaatkan ukuran *impurity*, seperti *Gini impurity* atau *entropy*, untuk menentukan fitur yang memberikan kemampuan pemisahan kelas paling baik [27]. Setelah seluruh pohon selesai dibangun, prediksi dari masing-masing pohon dikombinasikan melalui mekanisme *majority voting* guna menghasilkan keputusan akhir pada permasalahan klasifikasi [26]. Kombinasi banyak pohon keputusan tersebut memungkinkan Random Forest mengenali pola hubungan yang kompleks, termasuk hubungan non-linear dan interaksi antar variabel, tanpa memerlukan pemodelan secara eksplisit [28].

Pada permasalahan klasifikasi, Random Forest menawarkan berbagai keunggulan yang menjadikannya salah satu algoritma yang banyak digunakan. Algoritma ini mampu memodelkan hubungan non-linear antar variabel, menyediakan informasi mengenai tingkat kepentingan fitur (*feature importance*), serta memiliki kemampuan yang baik dalam mengurangi risiko *overfitting* [6]. Berkat karakteristik tersebut, Random Forest sering menunjukkan performa yang kompetitif pada berbagai kasus prediksi. Dalam penelitian terkait prediksi keberhasilan kampanye *crowdfunding*, algoritma ini dilaporkan mencapai tingkat akurasi sebesar 87,85% dan menghasilkan kinerja yang lebih baik dibandingkan beberapa metode klasifikasi lainnya [6][29].

2.7 Feature Engineering

Feature engineering merupakan proses pemilihan, transformasi, dan pembuatan fitur baru untuk meningkatkan kualitas prediksi model *machine learning*. Dalam prediksi keberhasilan *crowdfunding*, fitur yang digunakan umumnya dapat dikelompokkan menjadi tiga jenis utama, yaitu fitur numerik,

fitur kategorikal, dan fitur tekstual [13]. Fitur numerik meliputi target pendanaan, durasi kampanye, serta jumlah pendukung (*backers*), sedangkan fitur kategorikal mencakup kategori proyek dan negara asal kampanye. Sementara itu, fitur tekstual berasal dari konten kampanye seperti judul dan deskripsi proyek. Teknik *feature engineering* dapat mencakup pembuatan variabel turunan seperti rasio "pledge per person", faktor temporal, maupun ekstraksi fitur berbasis teks dari konten kampanye.

2.8 One-Hot Encoding

One-Hot Encoding (OHE) merupakan metode yang digunakan untuk mengubah data kategorikal ke dalam bentuk numerik sehingga dapat diolah oleh algoritma *machine learning*. Proses ini dilakukan dengan membuat variabel biner untuk setiap kategori yang tersedia, di mana nilai 1 menandakan keberadaan suatu kategori dan nilai 0 menunjukkan ketidakhadirannya. Berbagai penelitian menunjukkan bahwa *One-Hot Encoding* dapat mendukung kinerja model klasifikasi secara efektif, meskipun penerapannya berpotensi meningkatkan dimensi fitur pada dataset [30][31]. Dalam penelitian ini, *One-Hot Encoding* digunakan untuk mengonversi fitur-fitur kategorikal pada data latih dan data uji ke dalam format numerik sebelum dilakukan proses pelatihan model.

2.9 Teknik Pembagian Dataset

Pembagian data merupakan salah satu tahapan penting dalam pengembangan model *machine learning* karena berperan dalam proses evaluasi kemampuan model terhadap data yang tidak digunakan selama pelatihan. Pada umumnya, dataset dipisahkan menjadi data pelatihan (*training set*) dan data pengujian (*test set*), serta dapat ditambahkan data validasi (*validation set*) untuk mendukung proses penyesuaian parameter model. Melalui pembagian tersebut, model diharapkan mampu menghasilkan performa yang baik pada data baru, sehingga kemampuan generalisasinya meningkat dan risiko terjadinya *overfitting* dapat diminimalkan.

Metode pembagian tunggal seperti *holdout split* diketahui memiliki keterbatasan karena hasil evaluasi sangat bergantung pada pembagian data secara acak. Perbedaan pembagian data dapat menyebabkan variasi nilai evaluasi yang signifikan, sehingga estimasi performa model menjadi kurang stabil [32]. Selain

itu, karakteristik dataset dan algoritma juga memengaruhi hasil evaluasi. Jumlah data pelatihan dan jenis model dapat menentukan tingkat sensitivitas terhadap rasio pembagian data [33][34].

Untuk memperoleh hasil evaluasi yang lebih konsisten dan andal, banyak penelitian menerapkan metode validasi seperti *k-fold cross-validation*. Pada metode ini, dataset dibagi menjadi k subset (*fold*), kemudian setiap subset digunakan secara bergantian sebagai data validasi sementara subset lainnya digunakan sebagai data pelatihan. Proses tersebut diulangi hingga seluruh *fold* memperoleh kesempatan menjadi data validasi. Nilai $k = 5$ dan $k = 10$ merupakan konfigurasi yang paling sering digunakan karena mampu menghasilkan estimasi performa yang relatif stabil dengan keseimbangan yang baik antara bias dan varians, serta tetap mempertahankan efisiensi komputasi [35].

Selain metode validasi, pemilihan rasio pembagian data juga merupakan faktor penting dalam evaluasi model. Beberapa penelitian menunjukkan bahwa rasio 70:30 merupakan salah satu konfigurasi yang umum digunakan dan mampu memberikan performa yang optimal dibandingkan rasio lainnya [36]. Hasil penelitian lain juga menunjukkan bahwa penggunaan rasio 70:30 menghasilkan akurasi sebesar 89,05% dan nilai *Area Under the Curve* sebesar 98,10%, dengan tingkat kesalahan prediksi yang rendah [37].

2.10 Penanganan Ketidakseimbangan Kelas

Class imbalance atau ketidakseimbangan kelas terjadi ketika jumlah data pada setiap kelas dalam suatu dataset memiliki distribusi yang tidak proporsional. Kondisi ini dapat menyebabkan model cenderung lebih fokus pada kelas yang memiliki jumlah data lebih banyak sehingga performa prediksi terhadap kelas minoritas menjadi kurang optimal. Beberapa penelitian mengelompokkan rasio kelas sekitar 1:3 sebagai ketidakseimbangan kelas tingkat moderat yang tetap perlu diperhatikan dan ditangani dalam proses klasifikasi untuk memperoleh hasil yang lebih akurat dan seimbang [38][39].

Salah satu pendekatan yang umum diterapkan untuk mengatasi permasalahan ketidakseimbangan kelas adalah penggunaan bobot kelas (*class_weight*). Pendekatan ini memberikan perhatian yang lebih besar kepada kelas minoritas dengan meningkatkan penalti terhadap kesalahan prediksi pada kelas tersebut selama proses pelatihan model [40]. Dengan demikian, model didorong untuk tidak terlalu berfokus pada kelas mayoritas. Dalam praktiknya,

penggunaan parameter *class_weight='balanced'* memungkinkan penentuan bobot kelas secara otomatis berdasarkan distribusi data yang tersedia, sehingga kelas dengan jumlah sampel yang lebih sedikit akan memperoleh bobot yang lebih tinggi dalam proses pembelajaran model [41].

2.11 Confusion Matrix

Salah satu metode evaluasi yang digunakan untuk menilai kinerja model klasifikasi adalah *confusion Matrix*. Metode ini menilai kinerja model klasifikasi dengan cara membandingkan label kelas yang sebenarnya dengan hasil prediksi yang dikeluarkan oleh model [42][43]. Evaluasi ini disajikan dalam bentuk tabel dan dapat diterapkan pada permasalahan klasifikasi biner maupun multikelas. Dalam penggunaannya, setiap baris akan mencerminkan kelas aktual data. Sementara itu, setiap kolom akan merepresentasikan kelas yang diprediksi oleh model. Melalui susunan tersebut, *confusion matrix* mampu memberikan informasi yang lebih rinci mengenai hasil klasifikasi berdasarkan beberapa komponen utama yang menyusun matriks tersebut [44].

1. *True Positive* (TP)
   True Positive merupakan kondisi di mana model secara tepat memprediksi data yang sesungguhnya berkelas positif sebagai kelas positif.

2. *False Positive* (FP)
   False Positive adalah situasi di mana model salah mengklasifikasikan data yang sesungguhnya berkelas negatif menjadi kelas positif. Kesalahan semacam ini dikenal sebagai *Type I Error*.

3. *True Negative* (TN)
   True Negative merupakan kondisi di mana model secara tepat mengenali data berkelas negatif dan mengklasifikasikannya sebagai kelas negatif sesuai kondisi aktualnya.

4. *False Negative* (FN)
   False Negative terjadi saat model keliru mengklasifikasikan data yang sesungguhnya berkelas positif menjadi kelas negatif. Kesalahan jenis ini disebut sebagai *Type II Error*.

TP, TN, FP, dan TN digunakan untuk mengevaluasi kinerja model klasifikasi. Metrik pertama yang digunakan adalah akurasi (*accuracy*), yaitu

nilai proporsi prediksi yang didapatkan dari keberhasilan klasifikasi yang benar dibandingkan dengan seluruh data yang dievaluasikan. Perhitungan nilai akurasi ditunjukkan pada Rumus 2.1.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.1}$$

Selain akurasi, penelitian ini juga menggunakan presisi (*precision*) untuk mengukur nilai ketepatan model dalam mengidentifikasi data yang termasuk ke dalam kelas positif. Formula perhitungan presisi ditunjukkan pada Rumus 2.2.

$$Precision = \frac{TP}{TP + FP} \tag{2.2}$$

Metrik lainnya adalah *recall*, yang digunakan untuk mengukur kemampuan model dalam menemukan seluruh data yang benar-benar berada pada kelas positif. Perhitungan nilai *Recall* dapat dilihat pada Rumus 2.3.

$$Recall = \frac{TP}{TP + FN} \tag{2.3}$$

Selanjutnya, digunakan *F1-score*, yaitu metrik yang menggabungkan presisi dan *recall* melalui rata-rata harmonis. Metrik ini banyak digunakan ketika distribusi kelas tidak seimbang karena mampu memberikan gambaran yang lebih seimbang mengenai performa model. Perhitungan *F1-Score* ditunjukkan pada Rumus 2.4.

$$F1\text{-}Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{2.4}$$

Di luar metrik kuantitatif yang telah disebutkan, *confusion matrix* berfungsi menyajikan gambaran visualisasi yang lebih mendetail terkait pola hasil klasifikasi yang dihasilkan model. Visualisasi ini memungkinkan identifikasi pola kesalahan prediksi pada setiap kelas sehingga proses evaluasi dapat dilakukan secara lebih mendalam [45]. Dalam penelitian ini, *confusion matrix* disajikan pada model dengan performa terbaik karena model tersebut dianggap paling representatif dalam menggambarkan kemampuan prediksi yang diperoleh.