ISSN: 2339-2541
JURNAL GAUSSIAN, Volume 14, Nomor 1, Tahun 2025, Halaman 107 - 117
Online di: https://ejournal3.undip.ac.id/index.php/gaussian/

KLASIFIKASI KUALITAS KOPI ARABIKA DENGAN METODE *RANDOM FOREST* DAN *K-NEAREST NEIGHBOR* PADA *IMBALANCED DATASET*

Hagi Afdal Fatan[1*], Tatik Widiharih[2], Sudarno[3]
[1,2,3] Departemen Statistika, Fakultas Sains dan Matematika, Universitas Diponegoro
***e-mail**: hagifatan22@gmail.com

DOI: 10.14710/j.gauss.14.1.107-117

Article Info:
Received: 2024-07-10
Accepted: 2025-07-15
Available Online: 2025-06-21

Keywords:
Classification, *Arabica Coffee*, SMOTE, *K-Nearest Neighbor*, *Random Forest*

Abstract: Coffee is a superior plantation commodity in the export sector with high economic value. Coffee quality is the most important factor affecting the selling price, so coffee quality assessment is the main key in setting market prices and determining the export potential of coffee-producing countries. Coffee quality is divided into specialty, premium and regular based on bean defects and taste test values. Coffee quality prediction is needed to find out which coffee has the best quality. This study compares the Random Forest and K-Nearest Neighbor (KNN) methods to find out which algorithm is most effective in predicting coffee quality. The working principle of Random Forest is to build more than one decision tree and then determine the estimated value based on majority voting. KNN classifies data based on the distance between the data and other data. The coffee dataset used is sourced from the Coffee Quality Institute (CQI) Database. The data has problems to match resulting in a small recall value in the minority class, the SMOTE oversampling algorithm is used to improve classification performance. The advantage of oversampling compared to undersampling is that it does not lose data information. The results showed that the Random Forest method after SMOTE produced the best classification performance with accuracy and memory values of 80.26% and 80.59%, respectively.

1. PENDAHULUAN

Kopi berperan penting sebagai sumber devisa negara. Kopi termasuk salah satu komoditas perkebunan yang unggul dalam bidang ekspor dengan nilai ekonomi yang tinggi dan cakupan perdagangan yang luas di dunia. Menurut Davis *et al* (2011), kopi arabika dan robusta merupakan spesies biji kopi yang paling banyak diminati diantara 124 spesies biji kopi yang mereka identifikasi. Perdagangan dan permintaan pasar kopi dunia sebanyak 70% berasal dari arabika dan 26% dari robusta.

Kualitas kopi merupakan faktor terpenting yang mempengaruhi nilai jual, sehingga pengukuran kualitas kopi merupakan syarat penting untuk menentukan harga dan menentukan potensi ekspor dari negara produsen kopi. Penilaian kualitas kopi dilakukan melalui uji fisik dan uji cita rasa. Uji fisik dapat dinilai dari warna, ukuran dan bentuk biji kopi, sedangkan uji cita rasa dapat dinilai dari rasa, aroma, keasaman dan kebersihan cawan (Tolessa *et al.*, 2016).

Prediksi kualitas kopi sangat diperlukan untuk menentukan kualitas terbaik sehingga dapat digunakan sebagai tolak ukur memilih kopi bahkan untuk meningkatkan kualitas kopi itu sendiri. Algoritma *machine learning* dapat digunakan untuk prediksi kualitas kopi, namun tidak semua algoritma pada *machine learning* bekerja dengan baik (Arifin dan Sasongko, 2018). Menguji beberapa algoritma pada *machine learning* diperlukan untuk mengetahui algoritma mana yang dapat memberikan kinerja terbaik.

Penelitian ini menggunakan algoritma *Random Forest* dan *K-Nearest Neighbor* (KNN) untuk membandingkan kinerja kedua algoritma dalam melakukan klasifikasi kualitas kopi arabika. Hasil klasifikasi dari masing-masing algoritma akan dibandingkan berdasarkan

nilai akurasi dan *recall*. *Random Forest* bekerja dengan cara membangun lebih dari satu *decision tree* kemudian menggabungkan nilai dugaan dari setiap *tree* menjadi satu nilai melalui *majority voting*. Metode *Random Forest* memberikan akurasi klasifikasi yang baik, *robust* (kebal) terhadap *outlier*, serta efektif untuk mengatasi data yang tidak lengkap (Breiman, 2001). KNN mengklasifikasikan data objek berlandaskan jarak antara data objek tersebut dengan data objek lainnya (El Houby *et al.*, 2017). Metode KNN sangat sederhana dan mudah dipelajari diantara algoritma *machine learning* lainnya, dan juga tahan terhadap data pelatihan yang mengandung *noise* (Mutrofin *et al.*, 2014).

Data milik peneliti tidak selalu dapat digunakan untuk analisis secara langsung, terkadang terdapat permasalahan pada data tersebut, seperti ketidakseimbangan data. Suatu data dikatakan tidak seimbang apabila proporsi antar kelas respon tidak ekuivalen (Chawla, *et al.*, 2002). Data tidak seimbang menyebabkan metode *machine learning* lebih mudah mengklasifikasikan kelas mayoritas dibandingkan kelas minoritas. Untuk mengatasi *imbalanced dataset* bisa dilakukan penyeimbangan data dengan *oversampling* SMOTE yang mesintetis sampel baru dari kelas minor.

2. TINJAUAN PUSTAKA

Specialty Coffee Assosiation of America (SCAA) menilai kualitas kopi berdasarkan hubungan antara kecacatan biji dan cita rasa kopi. Cacat biji diukur melalui uji fisik (warna, ukuran, dan bentuk), sedangkan untuk menilai cita rasa dilakukan melalui *cup quality* (rasa, aroma, keasaman, kemanisan). *Q Grader* melakukan serangkaian uji fisik dan uji cita rasa di laboratorium. Kriteria penilaian kulaitas biji kopi spesialti dan premium dirinci pada Tabel 1 berikut:

Tabel 1 Kriteria kualitas biji kopi spesialti dan premium

Kualitas	Persyaratan	Kualitas	Persyaratan
Spesialti		**Premium**	
Cacat primer	0 (Nol)	Cacat primer+sekunder	Maksimum 8
Cacat sekunder	Maksimum 5	Kadar air	10-12%
Kadar air	10-12%	Cacat biji "Quaker"	Maksimum 3
Cacat biji "Quaker"	0 (Nol)	Nilai uji citarasa	<80
Nilai uji cita rasa	≥ 80		

Data yang tidak lengkap adalah masalah yang sering terjadi dan sulit untuk dihindari pada permasalahan pengolahan data di berbagai kasus. Data yang memiliki *missing value* bisa diatasi dengan menghapus data yang mengandung *missing value* atau menggantinya dengan imputasi nilai mean untuk data kontinyu dan nilai modus untuk untuk data kategorik.

Pemilihan parameter yang akan digunakan sangatlah penting karena akan mempengaruhi kinerja klasifikasi. Pencarian parameter terbaik bisa dilakukan dengan *hyperparameters tuning* menggunakan algoritma *randomized search*. Algoritma *randomized search* akan mengambil secara acak dari sebagian kombinasi parameter yang telah ditentukan peneliti, kemudian kombinasi terbaik dipilih berdasarkan rata-rata nilai *cross validation* yang tertinggi. Misalkan terdapat *hyperparameter* Z = {2,5,7,10,11} dan T = {0,3}, dari semua kombinasi Z dan T dengan hasil {2,0}, {2,3}, {5,0}, {5,3}, {7,0}, {7,3}, {10,0}, {10,3}, {11,0}, dan {11,3}, maka secara otomatis algoritma *randomized search* akan mengambil beberapa kombinasi untuk diuji.

Data penelitian umumya tidak digunakan semuanya dalam proses pelatihan. Model yang dibentuk harus bisa melakukan prediksi sama baiknya pada data baru agar menghindari *overfitting*, yaitu dengan cara membagi data pelatihan menjadi dua bagian. *Holdout validation* dapat digunakan untuk membagi data latih dan data uji menjadi dua bagian dengan rasio yang ditetapkan peneliti. Rasio yang biasanya digunakan peneliti yaitu 60/40,

70/30, atau 80/20 (Raschka, 2018). Metode *k-fold cross validation* juga dapat dipakai untuk membagi data, yaitu dengan cara membagi data latih dan data uji menjadi kelompok-kelompok "*k*", maka proses pelatihan menjadi sejumlah "*k*", dimana *performance* model yang dihasilkan adalah rata-rata dari semua proses pelatihan.

Metode *Random Forest* adalah metode klasifikasi berupa kumpulan pohon keputusan. Algoritma *Random Forest* merupakan pengembangan dari *Classification and Regression Tree* (CART) yang menggunakan metode *bootsrtrap aggregation* dan *random feature selection* (Breiman, 2001).

Proses pembangunan pohon klasifikasi di *Random Forest* sama seperti proses di CART, hanya saja tidak dilakukan pemangkasan. Untuk membangun pohon klasifikasi pada *Random Forest* dijelaslkan pada langkah-langkah berikut (Breiman, 2001; Breiman dan Cutler, 2003):

1. Tentukan nilai *n* (jumlah pohon) dan *m* (jumlah variabel pemilah). Nilai *m* yang disarankan adalah $\frac{1}{2}\sqrt{p}$, $\sqrt{p}$, $2\sqrt{p}$, dengan *p* adalah banyaknya variabel prediktor. Sutton (2005) merekomendasikan untuk menggunakan nilai $n \geq 100$ karena nilai tersebut condong menghasilkan hasil kesalahan klasifikasi yang konstan.
2. Menjalankan proses *bootstrap* yaitu pengambilan sampel acak berukuran a_n dengan pengembalian pada data latih.
3. Bangun pohon klasifikasi tunggal menggunakan data sampel yang dihasilkan selama proses *bootstrap*. Pohon klasifikasi dibangun dengan menerapkan *random feature selection*, yaitu pemilihan peubah penjelas secara acak pada setiap simpul (*node*) dengan $m < p$. Misalkan simpul T dipisah menjadi dua simpul cabang T_1 dan T_2, proses pemecahan simpul menggunakan indeks gini:

$$Gini(T) = 1 - \sum_{j=1}^{c}\left(P_j\right)^2 \quad (1)$$

$$Gini_a(T) = \frac{n_1}{n} Gini(T_1) + \frac{n_2}{n} Gini(T_2) \quad (2)$$

$$Gini_{split}(T) = Gini(T) - Gini_a(T) \quad (3)$$

   Dengan P_j adalah frekuensi relatif dari kelas *j* dan *u* adalah banyaknya kategori dalam variabel respon. Pemilah terbaik dipilih berdasarkan nilai $Gini_{split}(T)$ yang paling besar. Setelah terpilih pemilah terbaik, maka dilakukan pencarian gini untuk setiap simpul dan akan terus berulang hingga batas kriteria penghentian atau nilai $Gini_a(T) = 0$.
4. Ulangi langkah 2 dan 3 hingga *n* kali untuk mendaparkan *n* pohon klasifikasi. Setiap pohon klasifikasi memberikan satu suara sehingga, sehingga menghasilkan *n* suara. Penentuan klasifikasi dilakukan melalui *majority vote*.

KNN mengklasifikasin data dengan menghitung kedekatan data baru dengan data lama (Nofriansyah & Nurcahyo, 2015). Prinsip kerja KNN yaitu mencari jarak terpendek antara data yang akan diestimasi dengan *k* tetangga terdekatnya dari data pelatihan. KNN menggunakan parameter jarak untuk mengukur seberapa dekat data dengan tetangganya. Penelitian ini menggunakan *euclidean distance* dengan persamaan sebagai berikut (Han *et al*, 2011):

$$d_{ij} = \sqrt{\sum_{s=1}^{p}\left(x_{is} - x_{js}\right)^2} \quad (4)$$

Dengan d_{ij} adalah *euclidean distance* objek ke-i dan objek ke-j, dan p adalah jumlah variabel faktor. x_{is} merupakan objek ke-i pada variabel ke-s, sedangkan x_{js} objek ke-j pada variabel ke-s. Algoritma KNN memiliki tahapan sebagai berikut:

1. Bagi data menjadi data latih dan data uji.
2. Hitung e*uclidean distance* antara setiap data pelatihan dan setiap data uji.
3. Urutkan e*uclidean distance* dari yang terkecil ke terbesar.
4. Tentukan nilai $k = 1,2,3,4,\ldots,\sqrt{n}$, dengan *n* adalah jumlah data latih.
5. Perika kelas dari *k* tetangga terdekat.
6. Tetapkan kelas terbanyak dari *k* sebagai kelas data uji.
7. Evaluasi hasil klasifikasi dengan mengukur nilai akurasi.

Synthetic Minority Oversampling Technique (SMOTE) menerapkan prinsip *oversampling* yang menambahkan data kelas minoritas sehingga jumlahnya seimbang dengan kelas mayoritas. Cara kerja SMOTE yaitu mereplikasi data kelas minoritas dengan pendekataan ketetanggaan (Chawla *et al*, 2002). Untuk mendapatkan hasil replikasi dari data minor maka digunakan persamaan sebagai berikut:

$$x_{syn_k} = x_i + (x_{knn} - x_i)\beta_k, i = 1,2,\ldots,m \quad (5)$$

dengan,

x_{syn_k} : data hasil replikasi (data *syntetic*) ke-k
x_i : data ke-i yang akan direplikasi
x_{knn} : data yang memiliki jarak terdekat dengan x_i
β_k : bilangan *random* bangkitan ke-k

Bilangan random β dibangkitkan secara acak dengan bantuan *software* yang bernilai antara 0 hingga 1.

Metrik kinerja algoritma *machine learning* umumnya diukur menggunakan *confusion matrix*, seperti yang ditunujukkan pada tabel di bawah ini (untuk variabel respon dengan 3 kelas):

Tabel 2. Confusion Matrix **3×3**

Label Asli	Label Prediksi		
	Kelas 1	Kelas 2	Kelas 3
Kelas 1	TP	FN	FN
Kelas 2	FNR	TNR	FNR
Kelas 3	FN	FN	TN

dengan,

TP (*True Positive*) : banyaknya kelas ‘1’ yang benar diprediksi sebagai ‘1’.
TNR (*True Netral*) : banyaknya kelas ‘2’ yang benar diprediksi sebagai ‘2’.
TN (*True Negative*) : banyaknya kelas ‘3’ yang benar diprediksi sebagai ‘3’.
FP (*False Positive*) : banyaknya kelas ‘1’ yang salah diprediksi sebagai ‘2/3’.
FNR (*False Netral*) : banyaknya kelas ‘2’ yang salah diprediksi sebagai ‘1/3’.
FN (*False Negative*) : banyaknya kelas ‘3’ yang salah diprediksi sebagai ‘1/2’.

Accuracy (akurasi) adalah tingkat ketepatan antara nilai prediksi dengan nilai aslinya (Deolika *et all*., 2019). *Accuracy* dapat dihitung dengan rumus:

$$Accuracy = \frac{TP+TNR+TN}{Jumlah\ data} \quad (6)$$

Recall TP (*True Positive Rate*) adalah rasio dari kelas positif yang diidentifkasi secara benar dengan jumlah kelas positif yang sebenarnya (Deolika *et all*., 2019). *Recall* dihitung dengan rumus:

$$recall = \frac{TP}{TP+FP} \quad (7)$$

3. METODE PENELITIAN

Data penelitian yang digunakan adalah data kualitas kopi arabika di dunia pada tahun 2018. Data tersebut merupakan data sekunder dari *Coffee Quality Institute* (CQI) yang diperoleh melalui platform Github. Data berjumlah 1312 data kopi arabika.

Variabel penelitian terdiri dari variabel respon (Y) dan beberapa variabel prediktor (X). Adapun variabel-variabel tersebut adalah sebagai berikut:

1. Variabel Terikat
   Y = Kualitas Kopi, 1 = Spesialti; 2 = Premium; 3 = Reguler
2. Variabel Bebas
   X_1 = *Altitude*, X_2 = *Processing Method*, X_3 =*Aroma*, X_4 = *Flavor*, X_5 = *Aftertaste*, X_6 = *Acidity*, X_7 = *Body*, X_8 = *Balance*, X_9 = *Cupper Points*.

Data diolah dengan bantuan *software python.* Tahapan analisis data dirinci sebagai berikut:

1) Periksa *missing value*. Jika terdapat *missing value* pada data penelitian, maka *missing value* diganti dengan nilai mean (variabel kontinu) atau nilai modus (variabel diskrit).
2) Salin data sehingga ada 2 data yang persis sama.
   a. Data pertama merupakan data asli setelah pre-*processing*. Berikut langkah-langkah penelitian selanjutnya:
      i Bagi data latih dan data uji menggunakan rasio 80:20.
      ii Lakukan *hyperparameters tuning* untuk *Random Forest* dan KNN menggunakan *randomized search*.
      iii Bangun model pohon klasifikasi *random forest*.
      iv Membuat prediksi pada data uji menggunakan metode KNN dan model pohon klasifikasi *random forest* yang dihasilkan.
      v Ukur akurasi dan *recall* prediksi klasifikasi *random forest* dan KNN.
   b. Data kedua diseimbangkan dengan SMOTE sehingga kelas spesialti, premium dan reguler jumlah datanya menjadi sama besar. Kemudian dilakukan langkah-langkah penelitian sebagai berikut:
      i Bagi data latih dan data uji menggunakan rasio 80:20.
      ii Lakukan *hyperparameters tuning* untuk SMOTE *Random Forest* dan SMOTE KNN menggunakan *randomized search*.
      iii Bangun model pohon klasifikasi SMOTE *random forest*.
      iv Memembuat prediksi pada data uji menggunakan metode SMOTE KNN dan model pohon klasifikasi SMOTE *random forest* yang dihasilkan.
      v Ukur akurasi dan *recall* prediksi klasifikasi SMOTE *random forest* dan SMOTE KNN.
3) Tentukan model terbaik dari *random forest*, KNN, SMOTE *random forest* dan SMOTE KNN berdasarkan akurasi dan *recall* tertinggi.

4. HASIL DAN PEMBAHASAN

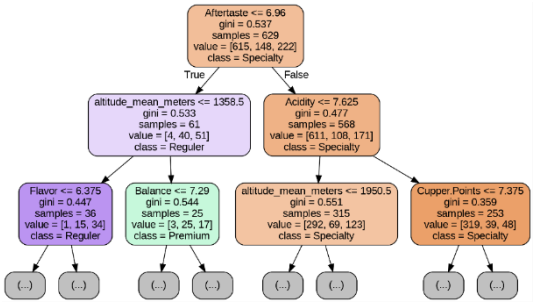
Pada proses pre-*processing* dilakukan pengecekan *missing value*. Hasil pengamatan menunjukkan bahwa ditemukan *missing value*, maka data tersebut diganti dengan nilai perkiraan menggunakan mean (data kontinu) dan modus (data kategorik) untu mengatasinya. Pada variabel *altitude* (X_1) dilakukan penghapusan *outliers* yaitu data yang <500. Jumlah data setelah dilakukan pre-*processing* menjadi 1232 data.

Pemilihan parameter terbaik dilakukan dengan *hyperparameters tuning* menggunakan *randomized search* dan *resampling* menggunakan 10-*fold cross validation*. Parameter yang diuji disajikan pada tabel berikut:

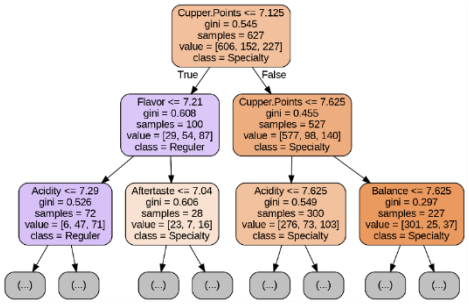
Tabel 3. Parameter *Random Forest*

Parameter	Nilai Parameter
m	2, 3, 4, 5, 6, 7
n	50, 75, 100, 150 200, 250, 300, 500, 750, 1000

Tuning parameter menghasilkan model terbaik pada parameter dengan nilai $m = 3$ dan $n = 750$. Pohon keputusan yang terbentuk dari algoritma *random forest* sebanyak 750 pohon. Model pohon keputusan *random forest* untuk pohon ke-1 dan pohon ke-750 dapat dilihat pada Gambar 1 dan Gambar 2.



Gambar 1. Pohon Klasifiksi Ke-1 Metode *Random Forest*



Gambar 2. Pohon Klasifiksi Ke-750 Metode *Random Forest*

Tabel 4 menampilkan hasil *confusion matrix* dari metode *random forest*, kemudian dilakukan perhitungan manual untuk mengukur kinerja *random forest* dalam melakukan klasifikasi kualitas kopi arabika berdasarkan akurasi dan *recall*.

Tabel 4. *Confusion Matrix* Metode *Random Forest*

Aktual	Prediksi		
	Spesialti (1)	Premium (2)	Reguler (3)
Spesialti (1)	142	0	5
Premium (2)	31	2	13
Reguler (3)	37	1	16

$$Accuracy = \frac{TP+TNR+TN}{Jumlah\ data} = \frac{142+2+16}{247} = 0{,}6478$$

$$Recall(1) = \frac{TP}{TP+FP} = \frac{142}{142+(0+5)} = 0{,}9660$$

$$Recall(2) = \frac{TNR}{TNR+FNR} = \frac{2}{2+(31+13)} = 0{,}0435$$

$$Recall(3) = \frac{TN}{TN+FN} = \frac{16}{16+(37+1)} = 0{,}2963$$

$$Recall = \frac{Recall(1)+Recall(2)+Recall(3)}{3} = \frac{0{,}9660+0{,}0435+0{,}2963}{3} = 0{,}4353$$

Klasifikasi mengunakan *random forest* memiliki nilai akurasi sebesar 64,78% dan nilai rata-rata *recall* 43,53%. Kelas spesialti (1) memiliki *nilai recall* yang besar mencapai 96,60%, artinya proporsi dari kelas spesialti yang dapat diklasifikasikan secara benar mencapai 96,69%, sedangkan kelas premium (2) dan regular (3) memiliki nilai *recall* yang kecil yaitu 11,11% dan 26,67%. Nilai *recall* yang kecil menunjukkan bahwa model tidak dapat mengklasifikasikan kopi yang tergolong ke dalam kelas premium dan regular dengan benar, hal ini disebabkan karena ketidakseimbangan jumlah data antar kelas.

Metode KNN memiliki parameter jarak dan parameter *k*. Nilai *k* paling sedikit adalah 1 dan nilai *k* paling besar adalah hasil akar kuadrat dari jumlah data latih (Hassanat *et al.*,

2014). Banyaknya data latih yang digunakan adalah 985 yang merupakan hasil dari 80% jumlah data. Nilai *k* paling besar adalah akar kuadrat dari 985 yaitu 31, sehingga penelitian ini menggunakan nilai $k = 1,2,3, \ldots ,31$ untuk mendapatkan nilai *k* terbaik.

Penentuan *k* terbaik dilakukan dengan algoritma *random search* dan *resampling* menggunakan 10-*fold cross validation*. *Tuning* parameter menghasilkan model terbaik pada parameter dengan nilai $k = 12$. Nilai $k = 12$ menjadi parameter untuk mencari tertangga terdekat sejumlah 12. Data uji dihitung jaraknya satu per satu ke seluruh data latih menggunakan rumus *euclidean distance* seperti pada persamaan (2). Hasil untuk *confusion matrix* dari metode KNN dapat dilihat pada Tabel 5, kemudian dilakukan perhitungan manual untuk mengukur kinerja KNN dalam melakukan klasifikasi kualitas kopi arabika berdasarkan akurasi dan *recall*.

Tabel 5. *Confusion Matrix* Metode KNN

Aktual	Prediksi		
	Spesialti (1)	Premium (2)	Reguler (3)
Spesialti (1)	136	5	6
Premium (2)	33	5	8
Reguler (3)	47	2	5

$$Accuracy = \frac{TP+TNR+TN}{Jumlah\ data} = \frac{136+5+6}{247} = 0{,}5911$$

$$Recall(1) = \frac{TP}{TP+FP} = \frac{136}{136+(5+6)} = 0{,}9252$$

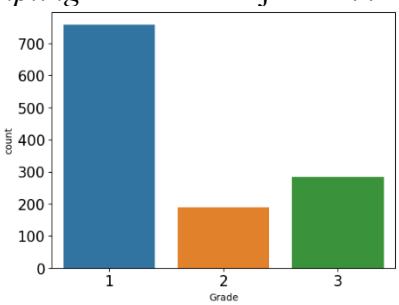
$$Recall(2) = \frac{TNR}{TNR+FNR} = \frac{5}{5+(33+18)} = 0{,}1087$$

$$Recall(3) = \frac{TN}{TN+FN} = \frac{5}{5+(47+2)} = 0{,}0926$$

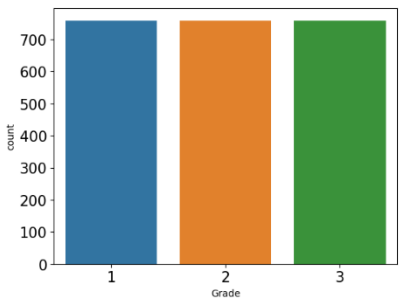
$$Recall = \frac{Recall(1)+Recall(2)+Recall(3)}{3} = \frac{0{,}9252+0{,}1087+0{,}0926}{3} = 0{,}4353$$

Klasifikasi mengunakan KNN memiliki nilai akurasi sebesar 59,11% dan nilai rata-rata *recall* yaitu 37,55%. Kelas spesialti (1) memiliki *nilai recall* yang besar mencapai 92,52%, artinya proporsi dari kelas spesialti yang dapat diklasifikasikan secara benar mencapai 92,52%, sedangkan kelas premium (2) dan regular (3) memiliki nilai *recall* yang kecil yaitu 10,87% dan 9,26%. Nilai *recall* yang kecil menunjukkan bahwa model tidak dapat mengklasifikasikan kopi yang tergolong ke dalam kelas premium dan regular dengan benar.

Hasil klasifikasi menggunakan metode *random forest* dan KNN mengahasilkan nilai *recall* yang rendah. Untuk mengatasinya maka dilakukan *oversampling* pada kelas minor menggunakan SMOTE. Penggunaan algoritma SMOTE membuat data pada kelas premium dan regular menjadi sama jumlahnya dengan data kelas spesialti yaitu 759. Total data setelah *oversampling* SMOTE menjadi 2277 data.

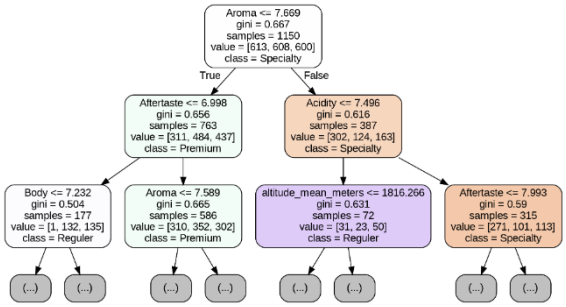


Gambar 3. Plot tiap kelas variabel Y sebelum dilakukan SMOTE

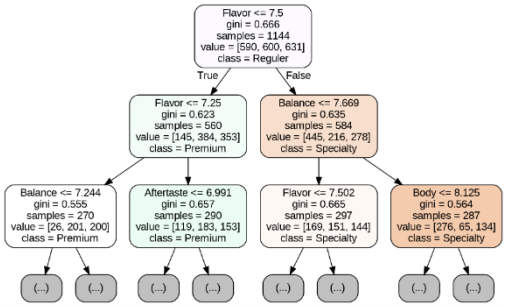


Gambar 4. Plot tiap kelas variabel Y setelah dilakukan SMOTE

Parameter yang dicobakan pada SMOTE *random forest* sama dengan parameter pada *random* forest. Kemudian dilakukan *hyperparameters tuning* untuk menentukan nilai *m* dan *n* terbaik menggunakan algoritma *random search* dan *resampling* menggunakan 10-*fold cross validation*. *Tuning* parameter menghasilkan model terbaik pada parameter dengan nilai *m* = 2 dan *n* = 300. Pohon keputusan yang terbentuk dari algoritma SMOTE *random forest* sebanyak 300 pohon. Model pohon keputusan *random forest* untuk pohon ke-1 dan pohon ke-300 dapat dilihat pada Gambar 5 dan Gambar 6.



Gambar 5. Pohon Klasifiksi Ke-1 Metode SMOTE *Random Forest*



Gambar 6. Pohon Klasifiksi Ke-300 Metode SMOTE *Random Forest*

Tabel 6 menampilkan hasil *confusion matrix* dari metode SMOTE *random forest*, kemudian dilakukan perhitungan manual untuk mengukur kinerja SMOTE *random forest* dalam melakukan klasifikasi kualitas kopi arabika berdasarkan akurasi dan *recall*.

Tabel 6. *Confusion Matrix* Metode SMOTE *Random Forest*

Aktual	Prediksi		
	Spesialti (1)	Premium (2)	Reguler (3)
Spesialti (1)	126	4	10
Premium (2)	22	127	15
Reguler (3)	24	15	113

$$Accuracy = \frac{TP+TNR+TN}{Jumlah\ data} = \frac{126+4+10}{456} = 0{,}8026$$

$$Recall(1) = \frac{TP}{TP+FP} = \frac{126}{126+(4+10)} = 0{,}9000$$

$$Recall(2) = \frac{TNR}{TNR+FNR} = \frac{127}{127+(22+15)} = 0{,}7744$$

$$Recall(3) = \frac{TN}{TN+FN} = \frac{113}{113+(24+15)} = 0{,}7434$$

$$Recall = \frac{Recall(1)+Recall(2)+Recall(3)}{3} = \frac{0{,}9000+0{,}7744+0{,}7434}{3} = 0{,}8059$$

Klasifikasi menggunakan SMOTE *random forest* meningkatkan nilai akurasi menjadi 80,59% dan nilai rata-rata *recall* menajdi 80,26%. Kelas spesialti (1) memiliki *nilai recall* 90%, terlihat menurun dibandingkan nilai *recall* pada metode *random forest* namun masih cukup baik mengklasifikasikan kelas spesialti dengan ketepatan klasifikasi sebesar 90%. Kelas premium (2) dan regular (3) mengalami peningkatan nilai *recall* menjadi 77,44% dan 74,34%, sehingga model yang terbentuk sudah mampu melakukan klasifikasi dengan baik pada kedua kelas.

Metode KNN memiliki parameter jarak dan parameter *k*. Nilai *k* paling sedikit adalah 1 dan nilai *k* paling besar adalah hasil akar kuadrat dari jumlah data latih (Hassanat *et al.*, 2014). Karena data hasil SMOTE bertambah, sehingga jumlah data latih yang digunakan pada SMOTE KNN pun berubah menjadi 1821 yang merupakan hasil dari 80% jumlah data.

Nilai *k* paling besar adalah akar kuadrat dari 1821 yaitu 43, sehingga penelitian ini menggunakan nilai *k* = 1,2,3, … ,43 untuk mendapatkan nilai *k* terbaik.

Penentuan *k* terbaik dilakukan dengan algoritma *random search* dan *resampling* menggunakan 10-*fold cross validation*. *Tuning* parameter menghasilkan model terbaik pada parameter dengan nilai *k* = 1. Nilai *k* = 1 menjadi parameter untuk mencari 1 tertangga terdekat. Data uji dihitung jaraknya satu per satu ke seluruh data latih menggunakan rumus *euclidean distance* seperti pada persamaan (2). Hasil untuk *confusion matrix* dari metode SMOTE KNN dapat dilihat pada Tabel 7, kemudian dilakukan perhitungan manual untuk mengukur kinerja SMOTE KNN dalam melakukan klasifikasi kualitas kopi arabika berdasarkan akurasi dan *recall*.

Tabel 7. *Confusion Matrix* Metode SMOTE KNN

Aktual	Prediksi		
	Spesialti (1)	Premium (2)	Reguler (3)
Spesialti (1)	86	26	28
Premium (2)	17	135	12
Reguler (3)	13	18	121

$$\text{Accuracy} = \frac{TP+TNR+TN}{Jumlah\ data} = \frac{86+135+121}{456} = 0{,}75$$

$$\textit{Recall(1)} = \frac{TP}{TP+FP} = \frac{86}{86+(26+28)} = 0{,}6143$$

$$\textit{Recall(2)} = \frac{TNR}{TNR+FNR} = \frac{135}{135+(17+12)} = 0{,}8223$$

$$\textit{Recall(3)} = \frac{TN}{TN+FN} = \frac{121}{121+(13+18)} = 0{,}7961$$

$$\textit{Recall} = \frac{Recall(1)+Recall(2)+Recall(3)}{3} = \frac{0{,}6143+0{,}8223+0{,}7961}{3} = 0{,}7445$$

Klasifikasi menggunakan SMOTE KNN meningkatkan nilai akurasi menjadi 75% dan nilai rata-rata *recall* menjadi 74,45%. Kelas spesialti (1) memiliki *nilai recall* 61,43%, terlihat menurun dibandingkan nilai *recall* metode KNN, namun masih cukup baik mengklasifikasikan kelas spesialti dengan ketepatan klasifikasi sebesar 61,43%. Kelas premium (2) dan regular (3) mengalami peningkatan nilai *recall* menjadi 82,23% dan 79,61%, sehingga model yang terbentuk sudah mampu melakukan klasifikasi dengan baik pada kedua kelas.

Hasil klasifikasi menggunakan 4 metode yaitu *random forest*, KNN, SMOTE *random forest*, dan SMOTE KNN kemudian dibandingkan kinerja klasifikasinya. Model terbaik dipilih berdasarkan nilai akurasi dan *recall* tertinggi.

Tabel 8. Perbandingan 4 Metode berdasarkan Akurasi dan *Recall*

Ukuran Kebaikan Model	*Random Forest*	KNN	SMOTE *Random Forest*	SMOTE KNN
Accuracy	64,78%	59,11%	80,26%	75%
Recall	43,35%	37,55%	80,59%	74,45%
Recall (1)	96,60%	92,52%	90%	61,43%
Recall (2)	4,35%	10,87%	77,44%	82,23%
Recall (3)	29,63%	9,26%	74,34%	79,61%

Dari Tabel 8, diperoleh informasi bahwa metode SMOTE *random forest* memiliki tingkat akurasi dan *recall* tertinggi yaitu sebesar 80,26% dan 80,59%. Nilai *recall* setelah dilakukan *oversampling* SMOTE lebih besar dibandingkan sebelum dilakukan penyeimbangan data. Jika dilihat dari nilai *recall* pada masing-masing kelas, metode

SMOTE *random forest* mampu melakukan klasifikasi dengan baik pada semua kelas. Metode SMOTE KNN mampu melakukan klasifikasi sedikit lebih baik pada kelas premium (2) dan regular (3) dibandingkan SMOTE *random forest* karena memiliki nilai *recall* yang lebih tinggi, namun tidak untuk kelas spesialti (1) dengan nilai *recall* hanya 61,43%. Secara keseluruhan, berdasarkan perbandingan nilai akurasi dan *recall* maka dapat dikatakan bahwa metode SMOTE *random forest* merupakan metode terbaik untuk klasifikasi kualitas koi arabika.

5. PENUTUP

Berdasarkan hasil analisis yang telah dilakukan, maka diperoleh kesimpulan sebagai berikut:

1. Perbandingan metode *random forest* dan KNN sebelum dan sesudah dilakukan SMOTE diperoleh hasil bahwa metode SMOTE *random forest* memiliki tingkat akurasi dan *recall* tertinggi yaitu sebesar 80,26% dan 80,59%. SMOTE memperbaiki hasil klasifikasi sehingga nilai akurasi dan *recall* mengalami peningkatan ±20%.
2. Nilai *recall* dari kelas premium dan reguler pada SMOTE KNN lebih tinggi dibandingkan SMOTE *random forest* dengan selisih ±5%, jika peneliti ingin klasifikasi secara lebih akurat untuk kelas premium dan reguler maka bisa menggunakan SMOTE KNN dengan resiko ketepatan klasifikasi dari kelas spesialti menjadi berkurang.

DAFTAR PUSTAKA

Arifin, O., dan Sasongko, T. B. 2018. Analisa Perbandingan Tingkat Performansi Metode *Support Vector Machine* dan Naive Bayes *Classifier* Untuk Klasifikasi Jalur Minat SMA. Seminar Nasional Teknologi Informasi dan Multimedia 2018, 6(1), 67–72.

Breiman, L. dan Cutler, A. 2003. *Manual on Setting Up, Using, and Understanding Random Forest V4.0*. Tersedia di: Using_random_forests_v4.0.pdf (berkeley.edu) (diakses pada 1 Februari 2023).

Breiman, Leo. 2001. *"Random forests" in Machine Learning*. 45, 5–32.

Chawla, N. V., Bowyer, K. W., Hall, L. 0., dan Kegelmeyer, W. P. 2002. SMOTE: *Synthetic Minority Over-Sampling Technique*. Journal of Artificial Research Vol. 16, Hal. 321-357.

Davis, A. P., Tosh, J., Ruch, N., dan Fay, M. F. 2011. *Growing Coffee: Psilanthus (Rubiaceae) Subsumed on The Basis of Molecular and Morphological Data; Implications for The Size, Morphology, Distribution and Evolutionary History of Coffea.* Botanical Journal of The Linnean Society, 167(4), 357–377.

Deolika, A., Kusrini, dan Luthfi, ET. 2019. Analisis Pembobotan Kata pada Klasifikasi *Text Mining*. Jurnal Teknologi Informasi Vol.3, No.2, Hal: 179-184.

El Houby, E. M., Yassin, N. I., dan Omran, S. 2017. *A Hybrid Approach from Ant Colony Optimization and K-Nearest Neighbor for Classifying Datasets Using Selected Features*. Informatica, Vol. 41, No. 4.

Han, J., Kamber, M., dan Pei, J. 2011. *Data Mining: Concepts and Techniques (3rd ed.)*. Elsevier. https://doi.org/10.1016/B978-0-12-381479-1.00001-0 (diakses pada tanggal 20 Januari 2023)

Hassanat, A. B., Abbadi, M. A., dan Altarawneh, G. A. 2014. *Solving the Problem of the K Parameter in the KNN Classifier Using an Ensemble Learning Approach.* International Journal of Computer Science and Information Security (IJCSIS), Vol. 12, No. 8.

Mutrofin, S., Izzah, A., Kurniawardhani, A., dan Masrur, M. 2014. Optimasi Teknik Klasifikasi Modified *K-Nearest Neighbor* Menggunakan Algoritma Genetika. Jombang: Jurnal GAMMA, ISSN 0216-9037

Nofriansyah, D., dan Nurcahyom G.W. 2015. Algoritma Data Mining dan Pengujian. Sleman: Deepublish.

Raschka, S. 2018. *Model evaluation, model selection, and algorithm selection in machine learning*. arXiv.

Sutton C.D. 2005. *Classification and Regression Trees, Bagging, and Boosting. Handbook of Statistics* 24:303-329.

Tolessa, K., Rademaker, M., De Baets, B., dan Boeckx, P. 2016. *Prediction of Specialty Coffee Cup Quality Based on Near Infrared Spectra of Green Coffee Beans*. Talanta, 150, 367-374. Https://Doi.Org/10.1016/J.Talanta.2015.12.039 (diakses pada tanggal 1 Februari 2023)