PENGGUNAAN ALGORITMA RANDOM OVER SAMPLING UNTUK MENGATASI MASALAH IMBALANCE DATA PADA KLASIFIKASI GIZI BALITA

Syam Suryo Utomo[1], Triawan Adi Cahyanto[2], Bakhtiar Hadi Prakoso[3]
Jurusan Teknik Informatika, Fakultas Teknik
Universitas Muhammadiyah Jember
Jl. Karimata no.49 Jember, Indonesia
e-mail : syamsuryo15@gmail.com[1], triawanac@unmuhjember.ac.id[2], bahtiyar.hp@gmail.com[3]

***Abstract*— Tegaldlimo Health Center is one of the health implementing units in the sub-district and has the function of one of which is to make efforts to fulfill the nutrition and health of children under the age of five (toddlers). In Tegaldlimo Health Center up to 2018 there were 1186 children under five, here there are four nutritional status according to the authors which are very far apart, namely "good nutrition", "undernutrition", "over nutrition", "poor nutrition", with details of good nutrition as many as 1101 toddlers , malnutrition as many as 39 toddlers, over nutrition as many as 39 toddlers and malnutrition as many as 7 toddlers. To find solutions to class imbalance problems in the dataset, a model with a data level approach is proposed. The proposed model design includes the design of ROS level data approaches, and Naive Bayes based classification algorithms. The results show that the Naive Bayes model produces Excellent accuracy with an average accuracy of 0.957844. While the accuracy with the ROS + NB model tends to be lower or can be said to be not optimal to increase the accuracy value by having an average of 0.528716 Fail (failing) in classifying the Puskesmas dataset, this is because the amount of data is very significant from many "nutritional data" good "and will affect the Accuracy results of Random Over Sampling + Naive Bayes.**

Keywords: Random Over Sampling, Class Imbalance, Naive Bayes, Accuracy.

Intisari— Puskesmas Tegaldlimo adalah salah satu unit pelaksana kesehatan yang ada didaerah kecamatan dan mempunyai fungsi salah satunya melakukan upaya pemenuhan akan gizi dan kesehatan anak usia bawah lima tahun (balita). Di Puskesmas Tegaldlimo sampai 2018 tercatat ada sebanyak 1186 balita, disini ada empat status gizi yang menurut penulis sangat jauh perbedaannya yaitu "gizi baik", "gizi kurang", "gizi lebih", "gizi buruk", dengan rincian gizi baik sebanyak 1101 balita, gizi kurang sebanyak 39 balita, gizi lebih sebanyak 39 balita dan gizi buruk sebanyak 7 balita. Untuk mencari solusi masalah ketidakseimbangan kelas pada dataset, diusulkan model dengan pendekatan level data. Perancangan model yang diusulkan meliputi perancangan pendekatan level data ROS, dan algoritma pengklasifikasian berbasis *Naive Bayes*. Hasilnya menunjukkan bahwa model *Naive Bayes* menghasilkan akurasi *Excellent* (Sangat Baik) dengan rata – rata akurasi yang dihasilkan adalah 0,957844. Sedangkan akurasi dengan model ROS + NB cenderung lebih rendah atau dibisa dikatakan belum maksimal untuk meningkatkan nilai akurasi dengan memiliki rata – rata yaitu 0,528716 *Fail* (Gagal) dalam mengklasifikasi *dataset* Puskesmas, hal ini dikarenakan jumlah data yang sangat signifikan banyak dari data "gizi baik" dan akan mempengaruhi hasil Akurasi dari *Random Over Sampling + Naive Bayes*.

Kata Kunci— Random Over Sampling, Ketidakseimbangan Kelas, Naive Bayes, Akurasi.

I. PENDAHULUAN

Puskesmas Tegaldlimo adalah salah satu unit pelaksana kesehatan yang ada didaerah kecamatan dan mempunyai fungsi salah satunya melakukan upaya pemenuhan akan gizi dan kesehatan anak usia bawah lima tahun (balita). Sebelumnya penulis sudah mendapat izin dari Dinas Kesehatan Kabupaten Banyuwangi dan diteruskan ke Puskesmas Tegaldlimo untuk mendapatkan data gizi balita. Di Puskesmas Tegaldlimo sampai 2018 tercatat ada sebanyak 1186 balita, disini ada empat status gizi yang menurut penulis sangat jauh perbedaannya yaitu "gizi baik", "gizi kurang", "gizi lebih", "gizi buruk", dengan rincian gizi baik sebanyak 1101 balita, gizi kurang sebanyak 39 balita, gizi lebih sebanyak 39 balita dan gizi buruk sebanyak 7 balita. Dengan adanya klasifikasi terhadap gizi balita maka dapat mempermudah dan membuat waktu pengerjaan lebih efisien. sehingga penulis mendapat suatu permasalahan yaitu data "gizi baik" jauh lebih banyak daripada data "gizi kurang", "gizi lebih", "gizi buruk".

Proses klasifikasi dengan berbagai algoritma *machine learning*, yaitu algoritma yang mempelajari pengelompokan data berdasarkan pengolahan data yang telah ada sebelumnya, yang mempunyai tujuan untuk mendapatkan target kelas yang akurat. Namun kenyataannya muncul suatu permasalahan dalam proses klasifikasi tersebut ketika salah satu kelasnya mempunyai jumlah data yang jauh lebih banyak pada *training dataset*-nya, yaitu kumpulan data yang dijadikan bahan pengolahan data. Permasalahan tersebut disebut juga dengan *imbalance dataset problem*. Kondisi tersebut akan berpengaruh pada proses klasifikasi data yang akan dilakukan untuk menentukan kelas suatu data. Jika kondisi data tidak seimbang *(imbalance)* maka kecenderungan kelas data tidak stabil karena data akan lebih condong ke bagian data yang memiliki komposisi data lebih besar *(Majority class)* (Riswanto, dkk, 2007).

Untuk menyelesaikan permasalahan tersebut salah satu

pendekatan yang sangat populer adalah dengan metode sampling. Metode *sampling* secara umum bekerja untuk menyeimbangkan data sehingga komposisi data yang akan dihasilkan akan berbentuk sama besar. Terdapat dua metode sampling yaitu, *Random Over Sampling* dan *Random Under Sampling*. (Riswanto, dkk, 2007). Tugas Akhir ini akan menggunakan metode *random over-sampling* dalam proses *rebalance* data, yaitu proses penyeimbangan data. Dan selanjutnya untuk memperoleh hasil pengukuran nilai akurasi, analisis ini menggunakan metode *Naive Bayes (Nb)* sebagai algoritma klasifikasi.

Berangkat dari masalah tersebut, maka penelitian ini bertujuan untuk mengetahui tingkat akurasi algoritma *Random Over Sampling* untuk mengatasi terjadinya ketidakseimbangan kelas dan Mengetahui hasil analisa penggunaan algoritma *Random Over Sampling* dalam klasifikasi balita.

II TINJAUAN PUSTAKA

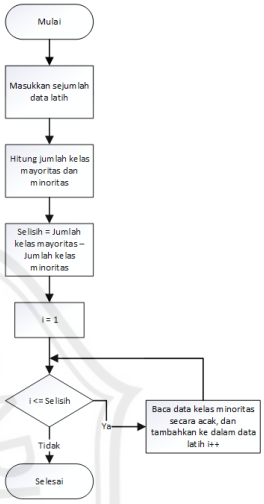
Data mining merupakan proses pengekstraksian informasi dari sekumpulan data yang sangat besar melalui penggunaan algoritma dan teknik penarikan dalam bidang statistik, pembelajaran mesin dan sistem basis data. Data mining adalah proses menganalisa data dari perspektif yang berbeda dan menyimpulkannya menjadi informasi – informasi penting yang dapat dipakai untuk meningkatkan keuntungan, memperkecil biaya pengeluaran, atau bahkan keduanya. Data mining sangat diperlukan terutama dalam mengelola jumlah data yang besar agar dapat memberikan informasi yang akurat untuk penggunanya. Dalam menambang data terdapat beberapa algortima klasifikasi untuk memproses data dalam jumlah besar tersebut. Proses penilaian terhadap objek data lalu memilah dan mengelompokkan ke dalam suatu kelas tertentu yang telah tersedia disebut klasifikasi (prasetyo, 2012).

Menurut He & Garcia, (2009, p. 1264) yang dikutip dari (Saifudin, 2014) Ketidakseimbangan kelas belajar mengacu pada belajar dari dataset yang menunjukkan ketidakseimbangan yang signifikan di antara atau di dalam kelas. Pemahaman umum tentang "ketidakseimbangan" dalam literatur berkaitan dengan situasi, di mana beberapa kelas sangat kurang terwakili dibandingkan dengan kelas lainnya. Secara konvensi, kelas yang memiliki lebih banyak contoh disebut kelas mayoritas, dan yang memiliki sedikitcontoh disebut kelas minoritas. Menurut (Wang & Yao, 2013) yang dikutip dari (Sabaruddin, 2017) Kesalahan klasifikasi dari contoh kelas minoritas biasanya lebih mahal. Untuk *software* prediksi cacat, karena sifat dari masalah, kasus cacat jauh lebih kecil kemungkinannya untuk terjadi daripada kasus tidak cacat. Kelas cacat sebagai minoritas. Pengenalan kelas ini penting, karena kegagalan menemukan cacat dapat sangat menurunkan kualitas perangkat lunak.

Menurut (Laurikkala, 2001) yang dikutip dari (ZK. Abdurahman, dkk, 2009) *Sampling* merupakan bagian dari ilmu statistik yang memfokuskan peneliatan terhadap pemilihan data yang dihasilkan dari satu kumpulan populasi data. Metode *sampling* atau yang lebih dikenal dengan *resample* adalah metode umum yang digunakan untuk menyelesaikan permasalahan *imbalance* data. Dengan adanya penerapan *sampling* pada data yang *imbalance*, tingkat *imbalance* semakin kecil dan klasifikasi dapat dilakukan dengan tepat. Sedangkan metode *Oversampling* dilakukan dengan menyeimbangkan jumlah distribusi data dengan meningkatkan jumlah data kelas minoritas.

Menurut Afzal & dan Torkar, (2008, p. 38) yang dikutip dari (Saifudin, 2014). *Resampling* ini sangat penting bagi penelitian validitas prediktif rekayasa perangkat lunak sejak dataset rekayasa perangkat lunak dalam keadaan langka dan datanya terbatas. Hal ini berkaitan dengan kesulitan dalam mendapatkan dataset yang besar karana datanya rahasia atau datanya belum lengkap di alam.

Pada algoritma ROS, data kelas minoritas dipilih secara acak, kemudian ditambahkan ke dalam data latih. Proses pemilihan dan penambahan ini diulang-ulang sampai jumlah data kelas minoritas sama dengan jumlah kelas mayoritas. Algoritma ROS digambarkan menggunakan flowchart pada gambar 2.1. pertama dihitung selisih antara kelas minoritas. Kemudian perulangan sebanyak hasil perhitungan selisih sambil membaca data kelas minoritas secara acak, dan ditambahkan ke dalam data latih.



Sumber: (Saifudin, 2015)

Gambar 2.1 *Flowchart* Algoritma ROS *(Random Over-Sampling).*

Menurut Zhang & Wang, (2011) yang dikutip dari (Saifuddin, 2014) Keseluruhan akurasi pada pengujian dataset umumnya digunakan untuk mengevaluasi kinerja pengklasifikasian. Tetapi untuk data yang tidak seimbang, akurasi yang mendalam didominasi oleh kelas minoritas, sehingga *alternative* evaluasi metrik digunakan. Metrik evaluasi yang tepat termasuk, semua akurasi dan rata-rata akurasi untuk minoritas. Untuk melakukan evaluasi dan validasi terhadap model yang diusulkan, maka dilakukan beberapa pengujian menggunakan *confusion matrix.*

Menurut Refaeilzadeh, Tang, & Liu, (2009) yang dikutip dari (Sabaruddin, 2017) *Cross validation* merupakan metode statistik untuk mengevaluasi dan membandingkan algoritma pembelajaran *(learning algorithms)* dengan membagi data menjadi dua segmen, satu segmen digunakan untuk belajar atau data latih, dan yang lain digunakan untuk memvalidasi model. Dalam *cross validation* harus *crossover* berturut-turut sehingga setiap data memiliki kesempatan tervalidasi.

Menurut Korb & Nicholson, (2011) yang dikutip dari (Sabaruddin, 2017) *K-fold cross validation* adalah teknik umum untuk memperkirakan kinerja pengklasifikasian. *K-fold cross validation* dilakukan dengan menggunakan kembali dataset yang sama, sehingga menghasilkan k perpecahan dari kumpulan data menjadi *non-overlapping* dengan proporsi pelatihan (k-1)/k dan 1/k untuk pengujian. Dengan menggunakan *Cross validation* akan digunakan percobaan sebanyak k. Data yang digunakan dalam

percobaan ini adalah data *training* untuk mencari nilai *error rate* secara keseluruhan. Secara umum pengujian nilai k dilakukan sebanyak 10 kali untuk memperkirakan akurasi estimasi. Dalam penelitian ini nilai k yang digunakan berjumlah 10 atau 10-*fold Cross validation*. Tiap percobaan akan menggunakan satu data *testing* dan k-1 bagian akan menjadi data *training*, kemudian data *testing* itu akan ditukar dengan satu buah data *training* sehingga untuk tiap percobaan akan didapatkan data *testing* yang berbeda-beda.

Menurut Witten & Frank, (2011) yang dikutip dari (Sabaruddin, 2017) Data training adalah data yang akan dipakai dalam melakukan pembelajaran sedangkan data testing adalah data yang belum pernah dipakai sebagai pembelajaran dan akan berfungsi sebagai data pengujian kebenaran atau keakurasian hasil pembelajaran.

Tabel 2.1. Ilustrasi 10-Fold Cross validation

Split 1	Split 2	Split 3	Split 4	Split 5	Split 6	Split 7	Split 8	Split 9	Split 10
Training									Test
Training								Test	Training
Training							Test	Training	
Training						Test	Training		
Training					Test	Training			
Training				Test	Training				
Training			Test	Training					
Training		Test	Training						
Training	Test	Training							
Test	Training								

Sumber: (Witten & Frank, 2011).

Menurut Gorunescu, (2011) yang dikutip dari (Sabaruddin, 2017) Alat ukur evaluasi berupa *confusion matrix* yang terdapat pada Anaconda 3 dengan tujuan mempermudah dalam menganalisis performa algoritma karena *confusion matrix* memberikan informasi dalam bentuk angka sehingga dapat dihitung rasio keberhasilan klasifikasi. *Confusion matrix* adalah salah satu alat ukur berbentuk 2x2 yang digunakan untuk mendapatkan jumlah ketetapan klasifikasi dataset terhadap kelas aktif dan tidak aktif pada algoritma yang dipakai.

Menurut Gorunescu, (2011) yang dikutip dari (Baharuddin, 2017) Tiap kelas yang diprediksi memiliki empat kemungkinan keluaran yang berbeda, yaitu *true positives* (TP) dan *true negatives* (TN) menunjukkan ketetapan klasifikasi. Jika prediksi keluaran bernilai positif sedangkan nilai aslinya adalah *false positif* (FP) negative sedangkan nilai aslinya adalah positif maka disebut dengan *false negative* (FN) salah. Berikut ini table 2.2 disajikan bentuk confusion matrix seperti yang telah disajikan sebelumnya.

Tabel 2.2 *Confusion Matrix*

Clarification	***Predicted Class***		
Observed Class		*Class = Yes*	*Class = No*
	Class = Yes	*A (True Positive –TP)*	*B (False Negative – FN)*
	Class = No	*C (False Positive-FP)*	*D (True Negative – TN)*

Sumber: (Khun, 2009).

Keterangan:

TP = True Positive, yaitu jumlah data positif yang terklasifikasi dengan benar oleh sistem.

TN = True Negative, yaitu jumlah data negatif yang terklasifikasi dengan benar oleh sistem.

FP = False Positive, yaitu jumlah data positif yang terklasifikasi dengan salah oleh sistem.

FN = False Negative, yaitu jumlah data negatif yang terklasifikasi dengan salah oleh sistem.

Menurut (Khun, 2008) yang dikutip dari (Baharuddin, 2017) Setelah dibuat *confusion matrix*, selanjutnya dihitung nilai akurasi. Rumus yang digunakan untuk melakukan perhitungan adalah:

a. Nilai *Accuracy* adalah seberapa akurat sistem dapat mengklasifikasi dengan benar. Dapat dihitung dengan menggunakan persamaan berikut:

$$\boldsymbol{Accuracy} = \frac{(A+D)}{(A+D+C+B)}$$

Naive Bayes merupakan sebuah pengklasifikasian probabilistik sederhana yang menghitung sekumpulan probalilitas dengan menjumlahkan frekuensi dan kombinasi nilai dari dataset yang diberikan. Algoritma menggunakan teorema Bayes dan mengasumsikan semua atribut independen atau tidak saling ketergantungan yang diberikan olehnilai pada variabel kelas. Definisi lain mengatakan *Naive Bayes* merupakan pengklasifikasian dengan metode probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya.

Persamaan dari teorema Bayes adalah :

$$P(H|X) = \frac{P(X|H).P(H)}{P(X)} \qquad (1)$$

Dimana :

X : Data dengan *class* yang belum diketahui

H : Hipotesis data merupakan suatu *class* spesifik

P(H|X) : Probabilitas hipotesis *H* berdasar kondisi *X* (posteriori probabilitas)

P(H) : Probabilitas hipotesis *H* (prior probabilitas)

P(X|H) : Probabilitas *X* berdasarkan kondisi pada hipotesis *H*

P(X) : Probabilitas *X*

Untuk menjelaskan metode *Naive Bayes*, perlu diketahui bahwa proses klasifikasi memerlukan sejumlah petunjuk untuk menentukan kelas apa yang cocok bagi sampel yang dianalisis tersebut. Karena itu, metode *Naive Bayes* diatas disesuaikan sebagai berikut:

$$P(C|F1 \dots Fn) = \frac{P(C)P(F1 \dots Fn|C)}{P(F1 \dots Fn)} \qquad (2)$$

Dimana variabel *C* merepresentasikan kelas, sementara variabel *F1...Fn* merepresentasikan karakteristik petunjuk yang dibutuhkan untuk melakukan klasifikasi. Maka rumus tersebut menjelaskan bahwa peluang masuknya sampel karakteristik tertentu dalam kelas *C (Posterior)* adalah

peluang munculnya kelas *C* (sebelum masuk sampel tersebut, seringkali disebut *prior*), dikali dengan peluang kemunculan karakteristik-karakteristik sampel pada kelas *C* (disebut juga *likelihood*), dibagi dengan peluang kemunculan karakteristik-karakteristik sampel secara global (disebut juga *evidence*). Karena itu, rumus di atas dapat pula ditulis secara sederhana berikut:

$$Posterior = \frac{prior\ x\ likelihood}{evidence} \qquad (3)$$

Nilai *evidence* selalu tetap untuk setiap kelas pada satu sampel. Nilai dari *posterior* tersebut nantinya akan dibandingkan dengan nilai-nilai *posterior* kelas lainnya untuk menentukan ke kelas apa suatu sampel akan diklasifikasikan. Penjabaran lebih lanjut rumus *Bayes* tersebut dilakukan dengan menjabarkan (C|F1,...,Fn) menggunakan aturan perkalian sebagai berikut :

$P(C|F_1, \dots, F_n = P(C)P(F_1,...,F_n|C)$

$= P(C)P(F_1|C)P(F_2,...,F_n|C,F_1)$

$= P(C)P(F_1|C)P(F_2|C,F_1)P(F_3,...,F_n|C,F_1,F_2)$

$= P(C)P(F_1|C)P(F_2|C,F_1)P(F_3|C,F_1,F_2)P(F_4,...,F_n|C,F_1,F_2,F_3)$

$=P(C)P(F_1|C)P(F_2|C,F_1)P(F_3|C,F_1,F_2)$ ...$P(F_n|C,F_1,F_2,F_3,...,F_{n-1})$ (4)

Dapat dilihat bahwa hasil penjabaran tersebut menyebabkan semakin banyak dan semakin kompleksnya faktor-faktor syarat yang mempengaruhi nilai probabilitas, yang hampir mustahil untuk dianalisa satu persatu. Akibatnya, perhitungan tersebut menjadi sulit untuk dilakukan. Di sinilah digunakan asumsi independensi yang sangat tinggi (*naif*), bahwa masing-masing petunjuk (F1,F2...Fn) saling bebas (*independen*) satu sama lain. Dengan asumsi tersebut, maka berlaku suatu kesamaan sebagai berikut:

$$P(F_i|F_j) = \frac{P(F_i \cap F_j)}{P(F_j)} = \frac{P(F_i)P(F_j)}{P(F_j)} = P(F_i) \qquad (5)$$

Untuk i≠j, sehingga

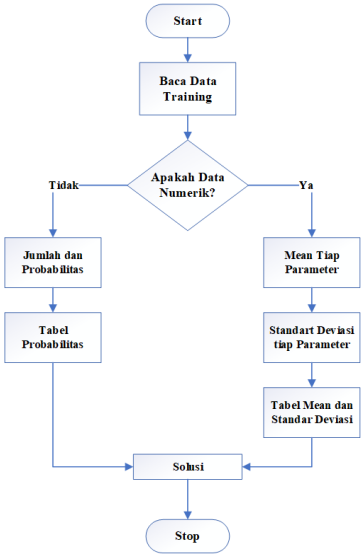
$$P(F_i|C,F_j) = P(F_i|C) \qquad (6)$$

Persamaan diatas merupakan model dari teorema *Naive Bayes* yang selanjutnya akan digunakan dalam proses klasifikasi. Untuk klasifiaksi dengan data kontinyu digunakan rumus *Densitas Gauss* :

$$P(X_i = x_i|Y = y_j) = \frac{1}{\sqrt{2\pi\sigma_{ij}}} e^{\frac{(x_i-\mu_{ij})^2}{2\sigma^2 ij}} \qquad (7)$$

Di mana :
P : Peluang
X_i : Atribut ke i
xi : Nilai atribut i
Y : Kelas yang dicari
yi : Sub kelas Y yang dicari
μ : mean, menyatakan rata-rata dari seluruh atribut
σ : Deviasi standar, menyatakan varian dari seluruh atribut.

Alur dari metode *Naive Bayes* dapat dilihat pada gambar 2.2 sebagai berikut:



Gambar 2.2. Alur Metode Naive Bayes(Sumber : Bustami, 2013)

Adapun keterangan dari gambar 2.2 di atas sebagai berikut:

1. Baca data training
2. Hitung jumlah probabilitas, namun apabila data numerik maka:
   a. Cari nilai mean dan standar deviasi dari masing-masing parameter yang merupakan data numerik.

   Adapun persamaan yang digunakan untuk menghitung nilai rata-rata hitung (mean) dapat dilihat sebagai berikut:

$$\mu = \frac{\sum_{i=1}^{n} x_i}{n} \qquad (8)$$

   Atau

$$\mu = \frac{x_1 + x_2 + x_3 + \cdots + x_n}{n} \qquad (9)$$

   Di mana :
   μ : rata-rata hitung (mean)
   x_i: nilai sample ke –i
   n : jumlah sampel

   dan persamaan untuk menghitung nilai simpangan baku (standar deviasi) dapat dilihat sebagai berikut:

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \mu)^2}{n-1}} \qquad (10)$$

   Di mana :
   σ : standar deviasi
   x_i: nilai x ke –i
   μ : rata-rata hitung
   n : jumlah sampel

   b. Cari nilai probabilistik dengan cara menghitung jumlah data yang sesuai dari kategori yang sama dibagi dengan jumlah data pada kategori tersebut.
3. Mendapatkan nilai dalam tabel mean, standard deviasi dan probabilitas.
4. Solusi kemudian dihasilkan.

Menurut Riska Amelia,(2013) yang dikutip dari(Sediaoetama, 2010). Status gizi adalah keadaan tubuh yang merupakan hasil dari keseimbangan antara zat gizi yang masuk ke dalam tubuh dan utilisasinya.

Gizi adalah suatu proses organisme menggunakan makanan yang dikonsumsi secara normal melalui proses pencernaan, absorbsi, transportasi, penyimpanan,

metabolisme, dan pengaruh zat-zat yang tidak digunakan untuk mempertahankan kehidupan, pertumbuhan dan fungsi nirmal dari organ-organ serta menghasilkan energi (Supariasa,dkk 2002).

Penilaian status gizi dapat digunakan sebagai pemantauan orang tua terhadap status gizi untuk pertumbuhan anak yang sangat dibutuhkan bagi perkembangan anak, juga dapat digunakan untuk memberikan penilaian status gizi terhadap seseorang yang berfungsi untuk rujukan dari masyarakat atau puskesmas. Banyak cara untuk menilai status gizi salah satunya adalah dengan cara pengukuran tubuh manusia yang dikenal dengan istilah *"Anthtropometri"*.

Dalam menentukan status gizi balita memiliki ukuran baku. Ukuran baku yang di gunakan di Indonesia sekarang adalah standar baku *World Health Organization-National Center for Health Statistics (WHO-NCHS)*. Penilaian status gizi balita dipisahkan antara laki-laki dan perempuan.

Menurut WHO, (2005) yang dikutip dari (Febrealty, 2011) penilaian status gizi berdasarkan Indeks BB/U (Berat Badan menurut Umur), TB/U (Tinggi Badan menurut Umur), BB/TB (Berat Badan menurut Tinggi Badan) dengan standar baku *Anthtropometri* WHO-NCHS dapat digolongkan menjadi:

Tabel 2.3 Penilaian Status Gizi (Febrealti, 2011)

No	Indeks yang dipakai	Batas Pengelompokan	Sebutan Status gizi
1.	BB/U	<-3 SD -2 s/d <-2 SD -2 s/d +2 SD > +2 SD	Gizi Buruk Gizi Kurang Gizi Baik Gizi Lebih
2.	TB/U	<-3 SD -2 s/d <-2 SD -2 s/d +2 SD > +2 SD	Sangat Pendek Pendek Normal Tinggi
3.	BB/TB	<-3 SD -2 s/d <-2 SD -2 s/d +2 SD > +2 SD	Sangat Kurus Kurus Normal Gemuk

Dimana SD adalah Skor Simpangan Baku (Standar Deviation = Z)

Cara menghitung ststus gizi dengan *Z-score* dapat ditentukan dengan menggunakan rumus:

$$Zscore = \frac{(Nilai\ Riel\ Perorangan - Nilai\ Mean\ Acuan)}{Nilai\ Simpang\ Baku\ Rujukan} \quad (11)$$

Dimana terdapat dua kategori dalam menghitung status gizi balita menggunakan *Z-score*, yaitu:

Bila "Nilai Riel Perorangan" hasil pengukuran ≥ "Nilai Median Acuan" BB/U, TB/U, BB/TB, maka rumusnya

$$Zscore = \frac{(Nilai\ Riel\ Perorangan - Nilai\ Mean\ Acuan)}{SD\ Upper}$$

$$Zscore = \frac{(Nilai\ Riel\ Perorangan - Nilai\ Mean\ Acuan)}{+1\ SD - Median}$$

Bila "Nilai Riel Perorangan" hasil pengukuran ≤ "Nilai Median Acuan" BB/U, TB/U, BB/TB, maka rumusnya

$$Zscore = \frac{(Nilai\ Riel\ Perorangan - Nilai\ Mean\ Acuan)}{SD\ Lower}$$

$$Zscore = \frac{(Nilai\ Riel\ Perorangan - Nilai\ Mean\ Acuan)}{Median - -1\ SD}$$

Keterangan: Nilai Riel itu berat badan sebenarnya (aktual)

Nilai Median itu diambil dari nilai tabel Baku Rujukan WHO-NCHS

Nilai (-1SD) itu juga dapat dilihat pada tabel WHO-NCHS

- Jika nilai riel lebih kecil dari pada nilai median berarti yang digunakan sebagai pembagi adalah nilai -1SD
- Jika nilai riel lebih besar dari pada nilai median berarti yang digunakan sebagai pembagi adalah nilai +1SD

Menurut Arisman, (2008) yang dikutip dari (Febrealty, 2011) interpretasi status gizi berdasarka tiga indeks *Anthtropometri* (BB/U, TB/U, BB/TB) adalah sebagai berikut:

Tabel 2.4 Interpretasi Status Gizi (Febrealty, 2011)

No	Indeks yang digunakan			Interpretasi
	BB/U	TB/U	BB/TB	
1.	Rendah Rendah Rendah	Rendah Tinggi Normal	Normal Rendah Rendah	Normal, dulu kurang gizi Sekarang kurang + + Sekarang kurang +
2.	Normal Normal Normal	Normal Tinggi Rendah	Normal Rendah Tinggi	Normal Sekarang kurang Sekarang lebih, dulu kurang
3.	Tinggi Tinggi Tinggi	Tinggi Rendah Normal	Normal Tinggi Tinggi	Tinggi, normal Obesitas Sekarang lebih, belum obesitas

Python adalah bahasa pemrograman interpretatif multiguna. Tidak seperti lain yang susah untuk dibaca dan dipahami, *python* lebih menekankan pada keterbacaan kode agar lebih mudah untuk memahami sintaks. Hal ini membuat *python* sangat mudah dipelajari baik untuk pemula maupun untuk yang sudah menguasai bahasa pemrograman lain.

Bahasa ini muncul pertama kali pada tahun 1991, dirancang oleh seorang bernama Guido van Rossum. Sampai saat ini *pyhton* masih dikembangkan oleh *Python Software Foundation*. Bahasa *python* mendukung hampir semua sistem operasi, bahkan untuk sistem operasi *Linux*, hampir semua distronya sudah menyertakan *python* di dalamnya.

Menurut Luts, (2010) yang dikutip dari (Harismawan, dkk. 2018) Python adalah bahasa pemrograman yag bersifat open source. Bahasa pemrograman ini dioptimasikan untuk *software quality*, *developer productivity*, *program portability*, *dan component integration*.

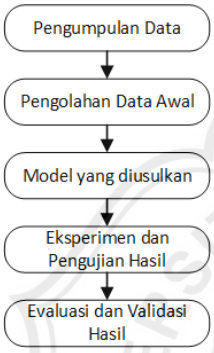
Menurut Luts, (2010) yang dikutip dari (Harismawan, dkk. 2018)Python telah digunakan untuk mengembangkan bahasa berbagai macam perangkat lunak, seperti *internet scripting*, *sytems programming*, *user interfaces*, *product customization*, *numeric programming* dll. Python saat ini telah menduduki posisi 4 atau 5 bahasa pemrograman paling sering digunakan diseluruh dunia.

III METODOLOGI PENELITIAN

Penelitian ini menggunakan pendekatan kuantitatif. Metode kuantitatif bertujuan untuk mencapai pemahaman tentang bagaimana suatu dikonstruksi, bagaimana dibangun, atau bagaimana cara kerjanya menurut Berndtsson, Hansson, Olsson, & Lundell, (2008) yang dikutip dari (Sabaruddin,

2017). Penelitian kuantitatif umumnya didorong oleh hipotesis, yang dirumuskan dan diuji secara ketat, dengan tujuan menunjukkan bahwa hipotesisnya salah. Oleh karena itu, salah satu uapaya adalah untuk menyalahkan hipotesis, dan jika hipotesis tahan uji, maka akan dianggap benar setelah terbukti. Aspek kuantitatif adalah untuk menekan bahwa pengukuran merupakan dasarnya karena memberikan hubungan antara observasi dan formalisasi model, teori, dan hipotesis.

Menurut Dawson, (2009) yang dikutip dari (Sabaruddin, 2017) Metode yang digunakan pada penelitian ini adalah eksperimen. Penelitian eksperimen mencakup investigasi hubungan sebab-akibat menggunakan pengujian yang dikontrol sendiri. Penelitian ini bertujuan untuk menerapkan pendekatan level data untuk mengurangi pengaruh ketidakseimbangan data. Karena penelitian yang diakui/diterima harus mengikuti aturan yang diakui, maka pada penelitian ini dilakukan dengan mengikuti tahapan seperti Gambar 3.1.

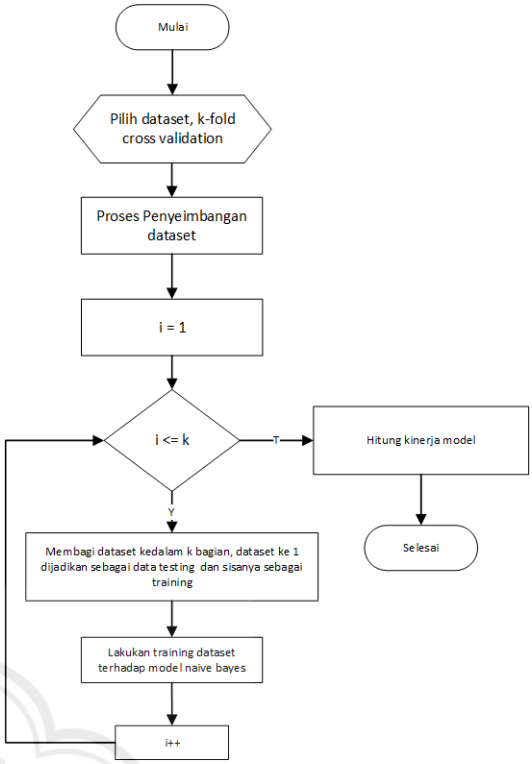


Gambar 3.1. Diagram Tahapan Penelitian

Pada penelitian ini digunakan data sekunder yaitu dari Puskesmas Tegaldlimo dengan proses pengambilan melalui Dinas terkait yaitu Dinas Kesehatan Kabupaten Banyuwangi terlebih dahulu, setelah mendapat persetujuan dari dinas kesehatan barulah dilanjutkan ke puskesmas Tegaldlimo. Adapun jumlah data yang diperoleh dari puskesmas tegaldlimo yaitu sebanyak 1186 dengan rincian balita gizi baik berjumlah 1101, balita gizi kurang berjumlah 39, balita gizi lebih berjumlah 39 dan balita gizi buruk berjumlah 7.

Pada pengolahan data awal ini akan dilakukan beberapa tahapan untuk memperoleh data yang diinginkan yaitu penggantian Status Gizi dengan gizi baik sama dengan 0, gizi kurang sama dengan 1, gizi lebih sama dengan 2, gizi buruk sama dengan 3.

Untuk mencari solusi masalah ketidakseimbangan kelas pada dataset, diusulkan model dengan pendekatan level data. Perancangan model yang diusulkan meliputi perancangan pendekatan level data ROS, dan algoritma pengklasifikasian berbasis *Naive Bayes*. Perancangan algoritma ditunjukkan melalui flowchart pada gambar 3.2.



Gambar 3.2 Flowchart Model yang Diusulkan.

Proses pertama yang dilakukan adalah memilih dataset, pendekatan level data, disini menggunakan pendekatan level data *Random Over Sampling* dan jumlah *k-fold* yang ingin diujikan, misal 10.

Proses kedua adalah proses penyeimbangan data, karena disini menggunakan *Random Over Sampling* jadi mencari selisih data antara data mayoritas dan data minoritas, setelah ketemu data minoritas ditambahkan secara *random* oleh sistem sehingga data minoritas dan data mayoritas seimbang.

Proses ketiga adalah membagi dataset kedalam *k* bagian, yang 1 dijadikan sebagai data *training* dan sisanya dijadikan data *testing*, lakukan training terhadap *Naive Bayes* dan ukur kinerjanya. Lakukan *looping* sesuai dengan *k* yang diinginkan.

Eksperimen dan hasil pengujian penelitian ini menggunakan resampling untuk menangani ketidakseimbangan kelas berbasis *Naive Bayes*. Gambar 3.3 menggambarkan algoritma model yang diusulkan pada penelitian ini.

Dataset yang digunakan dalam penelitian ini berasal dari data balita Puskesmas. Data ini diambil langsung dari Puskesmas dengan beberapa proses pengambilan yang sudah dijelaskan diawal.

Langkah pertama pada pengujian ini adalah menghitung terlebih dahulu selisih antara data gizi baik dan gizi kurang untuk menentukan jarak masing-masing data pada kelas mayoritas dan kelas minoritas. Algoritma yang diusulkan adalah *over-sampling*, yaitu menambah populasi kelas minoritas sedemikian hingga seimbang dalam jumlah kelas mayoritas.

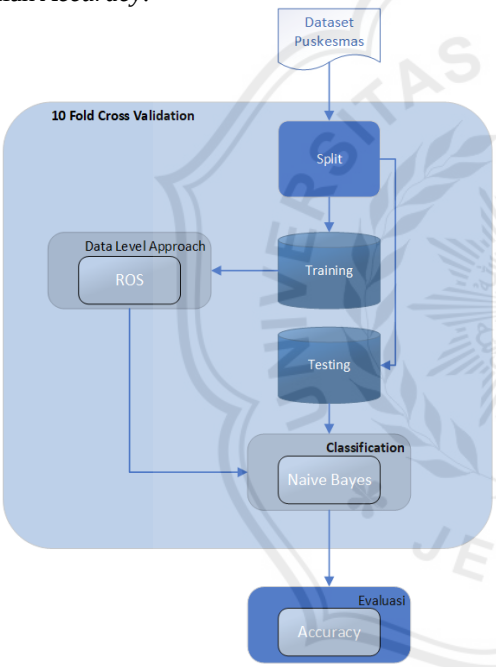
Langkah kedua pada pengujian ini adalah menggunakan hasil *resampling* yang didapat pada langkah sebelumnya sebagai data masukan untuk model prediksi berbasis *Naive Bayes*. Kinerja dari model prediksi kemudian dievaluasi menggunakan *2, 4, 5, 10-fold cross validation*.

Hasil validasi digunakan untuk mengukur kinerja masing-masing model, dan dilakukan perbandingan untuk mencari model yang memiliki kinerja terbaik.

Metode resampling yang digunakan adalah *random over sampling* (ROS). Perangkat lunak yang digunakan adalah Anaconda 3 dengan bahasa pemrograman Phyton dan Microsoft Excel.

Menurut Han, Kamber, & Pei, (2011, p.365) yang dikutip dari (Utomo Pujianto, 2016) Untuk mengukur kinerja model digunakan *confusion matrix*, karena *confusion matrix* merupakan alat yang berguna untuk menganalisa seberapa baik pengklasifikasidapat mengenali tupel/ fitur dari kelas yang berbeda menurut yang dikutip dari. *Confusion matrix* dapat membantu menunjukkan rincian kinerja pengklasifikasi dengan memberikan informasi jumlah fitur suatu kelasyang diklasifikasikan dengan cepat dan tidak tepat menurut Bramer, (2007, p. 89) yang dikutip dari (Utomo Pujianto, 2016). *Confusion matrix* merupakan matrik 2 dimensi yang menggambarkan perbandingan antara hasil prediksi dengan kenyataan.

Untuk data tidak seimbang, akurasi lebih didominasi oleh ketepatan pada data kelas minoritas, maka metrik yang tepat adalah *Accuracy*.



Gambar 3.3 Kerangka kinerja yang diusulkan

Pengukuran kinerja model dilakukan dengan menggunakan confusion matrix. Kinerja yang diukur termasuk akurasi secara umum, akurasi dalam memprediksi kelas minoritas. Confusion matrix diperoleh dari validasi menggunakan *2, 4, 5, 10-fold cross validation*, sehingga model yang terbentuk dapat langsung diuji dengan melakukan 2, 4, 5, 10 kali pengujian.

Kinerja model yang diperoleh digunakan untuk membandingkan antara model dasar dengan algoritma *Naive Bayes* dengan model yang dibentuk menggunakan algoritma *Naive bayes* dengan pendekatan level data ROS. Untuk melihat model yang terbentuk. dimana nilai Accuracy digunakan untuk menentukan klasifikasi keakuratan pengujian diagnostik.

Sebuah pedoman umum untuk mengklasifikasikan keakuratan pengujian diagnostik menggunakan Accuracy dapat dilihat pada sistem tradisional (Gorunescu, 2011) disajikan pada tabel 3.3.

Nilai	Klasifikasi
0.9 – 1	Excellent classification
0.8 – 0.9	Good classification
0.7 – 0.8	Fair classification
0.6 – 0.7	Poor classification
< 0.6	Failure

IV HASIL DAN PEMBAHASAN

Sumber data yang dipakai dalam penelitian ini berasal dari *dataset* Puskesmas Tegaldlimo. Sedangkan aplikasi pendukung yang digunakan adalah *Anaconda 3*. Model yang diuji adalah model pengklasifikasian *Naive Bayes* dan *Random Over Sampling*.

Jumlah data yang diperoleh dari *dataset* Puskesmas adalah sebanyak 1186 data, dengan kategori gizi baik sebanyak 1101 balita, gizi kurang sebanyak 39 balita gizi lebih sebanyak 39 balita dan gizi buruk sebanyak 7 balita. Karena hasil dari gizi baik, gizi kurang, gizi lebih dan gizi buruk bertipe *String* sehingga tidak bisa digunakan dan harus dirubah dalam bentuk *integer*, dengan ketentuan gizi baik adalah 0, gizi kurang adalah 1, gizi lebih adalah 2 dan gizi buruk adalah 3.

Tabel 4.1 Dataset Puskesmas

	Sex	Umur	BB	TB	Status_Gizi
0	2	8	7,3	65	0
1	1	7	8	65	0
2	2	7	8,3	68	0
3	1	42	14,6	95	0
4	2	41	14	93	0
...	...	...	...	...	...
1182	2	34	10	86	3
1183	2	33	10	86	3
1184	1	44	12	88,5	3
1185	2	44	11	100	3
1186	1	38	10,8	79	3

Dataset diatas disimpan menggunakan ekstensi *xlsx*. Format *xlsx* merupakan salah satu format ekstensi dari *Microsoft Office* yaitu *Microsoft Office Excel*. Berikut adalah *source code* yang digunakan dalam penggunaan file ekstensi xlsx di Phyton.

import pandas as pd
df = pd.read_excel('C:\databalita.xlsx')

Import pandas as pd merupakan *library* untuk penggunaan file ekstensi *xlsx*, *df* bisa di ibaratkan dengan *class* yang menampung semua kolom yang ada di *file* tersebut. *pd.read_excel* merupakan perintah untuk membaca format *xlsx* atau *excel*. *C:\databalita.xlsx* merupakan lokasi penyimpanan *dataset* yang akan digunakan.

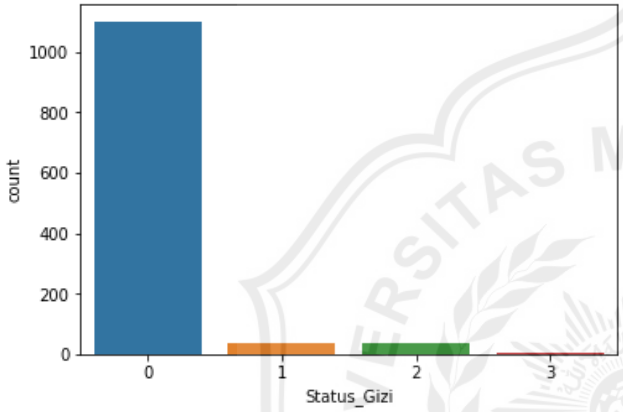
Teknik validasi yang digunakan adalah 2, 4, 5, 10-fold cross validation, sehingga dataset dibagi menjadi 2, 4, 5, 10 bagian, kemudian diambil 2 bagian secara random oleh sistem untuk dijadikan sebagai data uji (*test*) dan yang lainnya dijadikan data latih(*training*).

Untuk tahap selanjutnya yaitu penanganan kelas data yang tidak seimbang dengan menggunakan *OverSampling*:

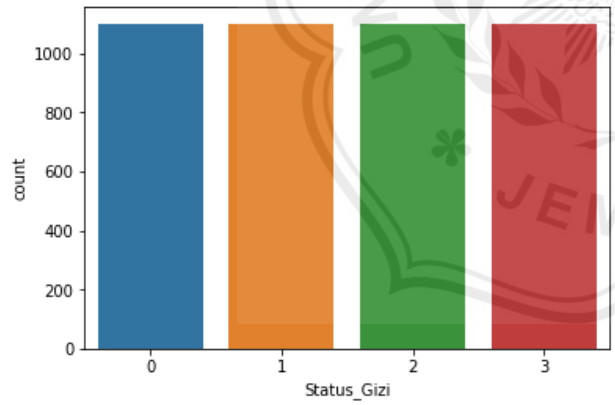
i. Menentukan nilai Fold dengan k =2, 4, 5, 10 dengan 70% dari data dijadikan data training dan 30% data dijadikan data testing.

ii. Menghitung jumlah data mayoritas dan data minoritas, yang selanjutnya dicari selisih antara ke 2 data tersebut.
iii. Melakukan perhitungan untuk mencari data minoritas untuk ditambahkan secara random.
iv. Menerapkan metode *Naive Bayes* untuk mengklasifikasi data uji.
v. Membandingkan kinerja klasifikasi tanpa dan dengan diterapkannya ROS *(Random Over Sampling)*.

Penerapan ROS untuk meningkatkan data minoritas yang jumlahnya jauh dibandingkan dengan data mayoritas pada dataset Puskesmas, dari data awal yang berjumlah 1186 data dengan Gizi baik 1101 data, gizi lebih 39 data, gizi kurang 39 data dan gizi buruk 7 data, sehingga total data keseluruhan setelah dilakukan *Random Over Sampling* menjadi 4404 yang terdiri dari 1101 Gizi Baik (label 0) , 1101 Gizi Kurang (label 1) 1101 Gizi Kurang(label 2), 1101 Gizi Buruk (label 3).



Gambar 4.1. *Grafik ketidak seimbangan awal*



Gambar 4.2. *Grafik setelah diterapkanny*a *Random Over Sampling*

Berdasarkan gambar diatas dapat dipahami bahwa penerapan *Random Over Sampling* untuk data Gizi kurang (1) yaitu sebesar 1062 data, Gizi lebih (2) yaitu sebesar 1062 data dan Gizi buruk (3) yaitu sebesar 1094 data.

Pada tabel berikut ditampilkan performa klasifikasi dengan *Naive Bayes* dan dengan *Naive Bayes + Random Over Sampling*. Pada setiap percobaan menerapkan nilai *k* yang bervariasi yaitu 2, 4, 5, 10.

Tabel 4.2. Hasil dari NB + ROS

Fold	Akurasi Ros cv 2	Akurasi Ros cv 4	Akurasi Ros cv 5	Akurasi Ros cv 10
1	0,486405	0,510511	0,503759	0,544776
2	0,512121	0,504532	0,548872	0,537313
3		0,507599	0,483019	0,541353
4		0,516717	0,555133	0,462121
5			0,492366	0,5
6				0,484848
7				0,530303
8				0,606061
9				0,549618
10				0,530769
Rata-Rata	**0,499263**	**0,50983975**	**0,5166298**	**0,5287162**

Tabel 4.3. Hasil dari NB

Fold	Akurasi NB cv2	Akurasi NB cv4	Akurasi NB cv5	Akurasi NB cv10
1	0,928934	0,929293	0,930556	0,918919
2	0,948718	0,938776	0,985915	0,972222
3		0,938776	0,971831	0,972222
4		0,948454	0,957746	1
5			0,929577	0,972222
6				0,971429
7				0,942857
8				0,942857
9				0,914286
10				0,971429
Rata-Rata	**0,938826**	**0,93882475**	**0,955125**	**0,9578443**

Pada bagian ini akan membahas hasil pengukuran kinerja model, yaitu Bagaimana tingkat akurasi algoritma *Random Over Sampling* untuk mengatasi terjadinya ketidakseimbangan kelas? Untuk menjawab ditunjukkan perbandingan kinerja model *Naive Bayes* dan model *Random Over Sampling + Naive Bayes* pada tabel 4.4 dan tabel 4.5 menunjukkan tingkat *accuracy* pada *dataset* Puskesmas.

Tabel 4.4. Perbandingan *Accuracy* model dengan *Naive Bayes*

Fold	NB	Keterangan
Fold 1	0,918919	Excellent
Fold 2	0,972222	Excellent
Fold 3	0,972222	Excellent
Fold 4	1	Excellent
Fold 5	0,972222	Excellent
Fold 6	0,971429	Excellent
Fold 7	0,942857	Excellent
Fold 8	0,942857	Excellent
Fold 9	0,914286	Excellent
Fold 10	0,971429	Excellent
Rata-Rata	**0,957844**	**Excellent**

Tabel 4.5. Perbandingan *Accuracy* model dengan *Naive Bayes* + ROS

Fold	ROS	Keterangan
Fold 1	0,544776	Failure
Fold 2	0,537313	Failure
Fold 3	0,541353	Failure
Fold 4	0,462121	Failure
Fold 5	0,5	Failure
Fold 6	0,484848	Failure
Fold 7	0,530303	Failure
Fold 8	0,606061	Poor
Fold 9	0,549618	Failure
Fold 10	0,530769	Failure
Rata-Rata	**0,528716**	**Failure**

Tabel perbandingan akurasi pada tabel 4.4 dan tabel 4.5 dari hasil beberapa fold yang digunakan adalah 2, 4, 5, 10 dan dari hasil yang terbaik adalah 10 fold, maka 10 fold yang diambil, menunjukkan bahwa model *Naive Bayes* menghasilkan akurasi *Excellent* (Sangat Baik) dengan rata – rata akurasi yang dihasilkan adalah 0,957844. Sedangkan akurasi dengan model ROS + NB cenderung lebih rendah atau dibisa dikatakan belum maksimal untuk meningkatkan nilai akurasi dengan memiliki rata – rata yaitu 0,528716 *Fail* (Gagal) dalam mengklasifikasi *dataset* Puskesmas, hal ini dikarenakan jumlah data yang sangat signifikan.

V KESIMPULAN

Data Puskesmas merupakan sebuah dataset yang memiliki ketidak seimbangan kelas. Penelitian ini menerapkan *Random Over Sampling* untuk menangani ketidak seimbangan kelas pada dataset Puskesmas, serta metode *Naive Bayes* (NB) untuk melaksanakan fungsi klasifikasi. Hasil eksperimen dengan menerapkan nilai k yang bervariasi yaitu 2, 4, 5, 10 dan mengambil dari nilai k yang terbaik yaitu 10 fold diperoleh rata – rata nilai performa *Accuracy* sebesar 53% untuk skema ROS + NB dan 96% untuk skema NB, hal ini menunjukkan bahwa ROS + NB kurang maksimal .

Berdasarkan hasil eksperimen di atas diperoleh kesimpulan bahwa penerapan *Naive Bayes + Random Over Sampling* kurang maksimal untuk mengatasi masalah *imbalance* data pada klasifikasi gizi balita dikarenakan jumlah data pada gizi baik yang signifikan lebih banyak dari pada status gizi yang lainnya, sehingga akan mempengaruhi hasil dari proses *random over sampling* pada klasifikasi balita.

DAFTAR PUSTAKA

Amelia, Rizka. (2013). *Penyapihan Dini dengan Status Balita Usia 0-24 Bulan Di Posyandu Dusun Kedungbendo Desa Gemekan Sooko Mojokerto*. Hospital Majapahit. Vol 5 No.1 Pebruari 2013.

Bustami. (2013). *Penerapan Algoritma Naive Bayes untuk Mengklasifikasi Data Nasabah Asuransi*. Jurnal Informatika. Universitas Malikussaleh. Vol.8, No.1 Januari 2014.

Eka, Rahmanurul Febrealti. (2011). *Sistem Penentuan Status Gizi Balita Menggunakan Metode K-NN (K-Nearest Neighbor)* [Skripsi]. Riau (ID). Universitas Islam Negeri Sultan Syarif Kasim.

Febrealty, Eka. *Klasifikasi Status Gizi balita menggunakan metode k-Nearest Neighbor*[Skripsi].2011.

Harismawan, F.A., Kharisma, P.A., Afirianto, T. (2018). *Analisis Perbandingan Performa Web Service Menggunakan Bahasa Pemrograman Python, PHP, dan Perl pada Client Berbasis Android.* Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer. Universitas Brawijaya. Vol. 2, No.1, Januari 2018.

Prasetyo, Eko. (2012). *Data Mining-Konsep dan Aplikasi Menggunakan Matlab*. Yogyakarta :Andi Offset.

Proverawati, Asfuah S., 2009. *Buku Ajar Gizi untuk Kebidanan*. Yogyakarta: Nuha Medika.

Pujianto, Utomo. (2016). *Strategi Resampling berbasis Centroid untuk Menangani Ketidakseimbangan Kelas pada Prediksi cacat Perangkat Lunak*. Malang (ID). Universitas Negeri Malang. Tekno, Vol 25 Maret 2016.

Riswanto Ricky, dkk (2007). *Metode Sampling dalam Menyelesaikan Data text Imbalance untuk Klasifikasi Multi-Label*. Bandung [ID]. Universitas Telkom.

Sabaruddin, Raja. (2017). *Prediksi Cacat Software Menggunakan Resampling Berbasis C5.0 untuk Menangani Ketidakseimbangan Kelas* [tesis]. Jakarta (ID). STMIK Nusa Mandiri.

Saifudin, Aries. (2014). *Pendekatan Level Data dan Algoritma untuk Penanganan Ketidakseimbangan pada Prediksi Cacat Software Berbasis Naive Bayes* [tesis]. Jakarta (ID). Sekolah Tinggi Manajemen Informatika dan Komputer Eresha.

Saifudin, Aries, Wahono. (2015). *Pendekatan Level Data Untuk Menangani Ketidakseimbangan Kelas pada Prediksi Cacat Software*. Jurnal Of Software Engineering, Vol.1:2.

Supariasa, D, dkk. (2002). *Penilaian Status Gizi*. Jakarta: EGC.

Suyanto. 2009. *Ilmu Pengantar Status Gizi*. Penerbir Parana Ilmu. Bandung.

Venny Lovina Gumiri, Diyah Puspitaningrum, Ernawati. (2015). *Sistem Pakar Klasifikasi Status Perkembangan Anak Usia Dini dengan Metode Naive Bayes Classifer Berbasis DDST Rules*. Jurnal Rekursif, Vol. 3:108-111.

ZK. Abdurahman Baizal, Moch. Arif Bijaksana, Angelina Sagita Sastrawan. (2009). *Analisis Pengaruh Metode Over Sampling Dalam Churn Prediction untuk Perusahaan Telekomunikasi*. Seminar Nasional Aplikasi Teknologi Informasi 2009 (SNATI 2009). G61