JOISIE Journal Of Information System And Informatics Engineering
Vol. 5, No.2 Desember 2021, Hlm 109-117

p- ISSN: 2503-5304
e- ISSN: 2527-3116

ANALISIS KLASIFIKASI BENCANA BANJIR BERDASARKAN CURAH HUJAN MENGGUNAKAN ALGORITMA NAÏVE BAYES

Slamet Triyanto[1], Andi Sunyoto[2], M. Rudyanto Arief[3]
[1] Ilmu Komputer, Magister PJJ Teknik Informatika, Universitas Amikom Yogyakarta, Yogyakarta, Indonesia
[2,3] Ilmu Komputer, Magister Teknik Informatika, Universitas Amikom Yogyakarta, Yogyakarta, Indonesia
Email: [1]slamet@students.amikom.ac.id, [2]andi@amikom.ac.id, [3]rudy@amikom.ac.id

Abstract

Research on data mining in various aspects, including disasters, especially floods, continues to be carried out. With the amount of rainfall and with only 2 (two) classes in a flood disaster, namely Flood and No Flood, then Naïve Bayes is very possible to be used as an algorithm in performing probability calculations. With good accuracy, the Naïve Bayes algorithm in this research is very suitable to be compared with various other algorithms, especially for classifying with 2 (two) classes. In addition to selecting the right algorithm, using the right dataset for testing makes the data mining process able to provide the right information. The test uses a dataset that has been tested with other algorithms before, is used as a comparison so that it is known by the pattern and characteristics of the dataset to obtain the right data mining algorithm information. Thus, scientific improvement, especially in data mining, can continue to grow. Although it is undeniable that the available datasets need to be preprocessed. This data preprocessing stage is carried out to get a good dataset. Then the data passes through the splitting stage, then the data is processed using the nave Bayes algorithm. In testing using the library from Scikit-Learn, it turned out that Naïve was able to provide an accuracy of 79.16%. Unlike the next test, Naïve Bayes was applied to RapidMiner as a comparison. The output of RapidMiner is quite surprising because the accuracy reaches 98.31%.

Keywords*: flood, rain fall, nave bayes, datasets, preprocessing datasets, scikit-learn library, rapidminer, accuracy*

Abstrak

Penelitian data mining dalam berbagai aspek termasuk dalam bencana, khususnya bencana banjir terus dilakukan. Dengan adanya jumlah curah hujan dan dengan hanya terdapat 2 (dua) kelas dalam bencana banjir yakni Banjir dan Tidak Banjir, maka Naïve Bayes sangat memungkinkan untuk digunakan sebagai algoritma dalam melakukan perhitungan probability. Dengan akurasi yang baik, maka algoritma Naïve Bayes dalam peneltian ini sangat layak disandingkan dengan berbagai algoritma lainnya khususnya untuk melakukan klasifikasi dengan 2 (dua) kelas. Selain pemilihan algoritma yang tepat, penggunaan dataset yang tepat untuk dilakukan pengujian menjadikan proses mining data dapat memberikan informasi yang tepat. Pengujian menggunakan dataset yang telah diuji dengan algoritma lain sebelumnya, dijadikan pembanding sehingga diketahui dengan pola dan ciri khas dataset diperoleh informasi algoritma data mining yang tepat. Dengan demikian, peningkatan keilmuan khususnya dalam data mining dapat terus berkembang. Meskipun tidak dapat dipungkiri dataset yang tersedia perlu untuk dilakukan preprocessing data. Tahap preprocessing data ini dilakukan untuk mendapat dataset yang baik. Kemudian data melwati tahap pembagian/splitting, barulah data diolah menggunakan algoritma naïve bayes. Dalam pengujian menggunakan library dari Scikit-Learn ini ternyata Naïve mampu memberikan akurasi sebesar 79.16%. Berbeda dengan pengujian selanjutnya, Naïve Bayes diterapkan pada RapidMiner sebagai pembanding. Keluaran dari RapidMiner cukup mengejutkan karena akurasi mencapai 98,31%.

Kata Kunci: *banjir, curah hujan, naïve bayes, datasets, datasets preprocessing, library scikit-learn, rapidminer, akurasi*

1. PENDAHULUAN

Proses untuk mendapatkan informasi yang menarik dari data yang terhimpun disebut dengan *data mining*(Han & Kamber, 2001; Prihandoko et al., 2017). Informasi yang diperoleh dengan *data mining* setidaknya dapat dibagi menjadi 6 (enam) (Refonaa et al., 2015) yakni Deteksi Anomali, Aturan Asosiasi pembelajaran, Clustering, Klasifikasi dan Regresi (Prihandoko et al., 2017; Refonaa et al., 2015). Dengan kemampuan data mining ini, maka sangat memungkinkan untuk melakukan analisis

probabiliti bencana. Analisis probabiliti menjadi penting mengingat bencana merupakan kondisi yang tidak diinginkan yang mengakibatkan kerugian kehidupan manusia dan menyebabkan pengungsian (Arjun Singh & Saxena, 2016). Pada prinsipnya bencana merupakan suatu kejadian yang mengganggu kondisi normal dan menimbulkan penderitaan yang melebihi kemampuanya masyarakat yang terimbas (Panafrican Emergency Training Centre, 2002)

Salah satu bencana yang sangat berbahaya di Dunia adalah banjir (Naik et al., 2021). Salah satu bencana yang sangat berbahaya di Dunia adalah banjir (Naik et al., 2021). Selain kejadiannya yang sering terjadi dan mengakibakan kerugian yang cukup besar seperti di Indoensia dalam Laporan Index Risiko Bencana Indoensia (IRBI) tahun 2020 menjabarkan bahwa bencana alam yang paling banyak terjadi adalah Banjir yang mencapai angka 1.070 Kejadian (Sesa et al., 2020). Begitu juga dengan Negara lain seperti di India yang juga mengindikasikan bahwa jumlah kejadian bencana banjir cukup tinggi khususnya di Negara bagian Karala (Naik et al., 2021). Sebagai bencana alam, banjir sangat erat kaitannya dengan keadaan cuaca (Prihandoko et al., 2017). Salah satu yang menjadi penyebab banjir adalah jumlah curah hujan (Naik et al., 2021; Nugroho, 2002).

Sebagai salah satu faktor penyebab banjir (Nugroho, 2002), data mengenai curah hujan ini tidak dapat serta merta menjadi label bahwa suatu wilayah terjadi banjir atau tidak(Nugroho, 2002). Sehingga perlu adanya pelabelan lebih lanjut untuk mendapat informasi kejadi sebenarnya. Dari dataset yang direlease secara public mengenai jumlah curah hujan dan kejadian banjir di Negara Bagian Kerala (Mukul, 2020), diperoleh data berupa jumlah curah hujan bulanan dari tahun 1901 hingga tahun 2018(Mukul, 2020) dan telah diberi label.

Saiesh Naik dkk melakukan analisis untuk memprediksi banjir dengan menggunakan algoritma *data mining* yakni Logistic Regression (Naik et al., 2021). Data yang digunakan untuk melakukan analisis adalah data dari Kerala State di India (Naik et al., 2021). Data tersebut berupa data jumlah curah hujan selama 115 Tahun yakni dari tahun 1901- 2015 (Naik et al., 2021), untuk saat ini data telah diupdate dari tahun 1901-2018 (Mukul, 2020) dan label berupa informasi YES untuk banjir dan NO untuk tidak banjir (Naik et al., 2021). Detail informasi dataset yang digunakan pada kolom pertama berupa SUB DIVISION yang berisi nama dari tempat yakni Kerala. Selanjutnya pada kolom kedua berisi tahun dengan *heading* YEARS dan tahun 1901 s/d 2015, sedangkan pada kolom ke tiga sampai kolom ke empat belas merupakan jumlah curah hujan perbulan dalam satu tahun dengan *heading* JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC. Sedangkan kolom ke lima belas merupakan kolom jumlah curah hujan pertahun dengan nama ANNUAL. Kolom terakhir yakni kolom enam belas merupakan kolom label dengan nama FLOODS yang berisi YES dan NO.

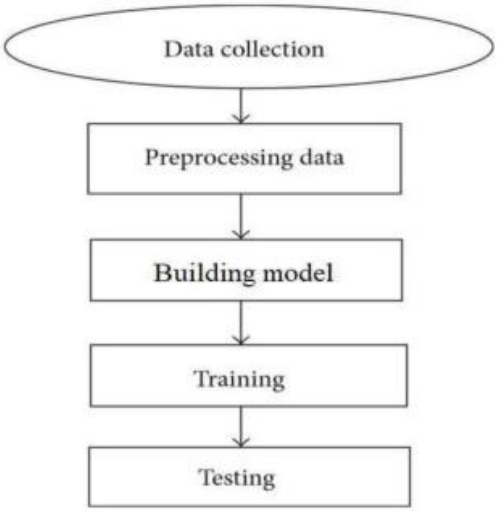
Masih didalam jurnal yang sama (Naik et al., 2021), para peneliti melakukan pengumpulan data. Dilanjutkan dengan melakukan *preprocessing data*, membangun model, melakukan training data dan melakukan test. Untuk saat ini, data yang telah dilakukan *preprocessing* dapat diunduh dari Kaggle dengan lisensi *public*(Mukul, 2020). Tahap *preprocessing data* terdapat beberapa hal yang dilakukan yakni dengan menghilangkan data *Sub Division* yang berupa data *text* nama Kerala dan merubah YES dan NO pada kolom FLOODS menjadi 1 dan 0 (Naik et al., 2021). Selanjutnya *data set* yang telah melewati tahap *preprocessing*, di bagi menjadi 2 (dua) sebagai data x dan y (Naik et al., 2021). Data x berisi data dari kolom 3 (tiga) sampai dengan kolom 15 dan y berisi data kolom 16 (Naik et al., 2021). Selajutnya data tersebut dengan menggunakan algoritma Logistic Regression dataset banjir dianalisis untuk mendapatkan prediksi banjir. Hasil dari analisis ini diperoleh informasi akurasi sebesar 75% dalam memprediksi bencana banjir(Naik et al., 2021).

Dengan adanya informasi akurasi sebesar 75%, maka dalam penelitian ini dilakukan analisis dengan menggunakan metode Naïve Bayes. Sedangkan data menggunakan dataset yang sama dengan yang digunakan untuk penelitian yang dilakuan oleh Saiesh Naik dkk sehingga didapat perbandingan akurasi antar kedua algoritma. Pemilihan penggunaan Naïve Bayes sebagai algoritma dalam penelitian merujuk pada artikel yang ditulis

oleh Aris J dkk (Ordoñez et al., 2018) yang melakukan penelitian untuk mengklasifikasikan SMS dengan menggunakan Naïve Bayes dan diperoleh informasi akurasi sebesar 89%. Hal menarik mengingat Naïve Bayes dari informasi *Microsoft Azure Machine Learning Algorithm Cheat Sheet* tidak menyertakan Naïve Bayes untuk melakukan klasifikasi terhadap 2 (dua) Kelas(Azure, 2021). Dari artikel tersebut diperoleh informasi bahwa untuk melakukan klasifikasi terhadap 2 (dua) kelas hanya terdapat 6 (enam) algoritma yakni Support Vector Machine, Average Perceptron, Decision Forest, Logistic Regression, Boosted Decision Tree, dan Neural Network. Hal ini menarik tentunya sangat menarik mengingat kemampuan akurasi yang dapat dilakukan oleh Naïve Bayes cukup tinggi.

2. METODE PENELITIAN

Sebagai sebuah algoritma, Naïve Bayes tidak serta merta dapat berdiri sendiri. Untuk itu penelitian diawali dengan mendapatkan *dataset*. Mengingat penlitian berusaha menemukan akurasi pada penggunaan algoritma pada *datamining*, maka Untuk *dataset* sendiri sesuai dengan rencana awal menggunakan *dataset* yang digunakan dalam penelitian yang dilakukan oleh Saiesh Naik dkk. *Dataset* ini berupa csv dengan nama Karela.csv, dan telah diuji dengan beberapa metode klasifikasi yakni KNN Classifier, Logistic Regression, Decision Tree Classification, Random Forest Classification dan Enseble Learning (Mukul, 2020). Meskipun demikian, dataset tetap mengikuti tahap-demi tahap dalam penelitian merujuk pada journal (Naik et al., 2021). Langkah pertama yakni *dataset* yang telah diperolah dilakukan tahap *preprocessing*. Membagi data untuk dijadikan sebagai data training dan data test, kemduan dilanjutkan dengan melakukan proses analisis dengan menggunakan algorita Naïve Bayes, dan diperoleh informasi akurasi penggunaan Naïve Bayes. dalam tahap analisis dilakukan dengan menggunakan *library* yang disedikan oleh scikit-learn.org (https://scikit-learn.org/, n.d.) menggunakan python. Selain menggunakan library yang disediakan scikit-learn.org (https://scikit-learn.org/, n.d.), analisis juga menggunakan RapidMiner (Kotu & Deshpande, 2014). RapidMiner digunakan sebagai analisis pembading. Adapun secara keseluruhan tahapan dalam penelitian ini adalah sebagai berikut :



Gambar 1. Tahap Penelitian

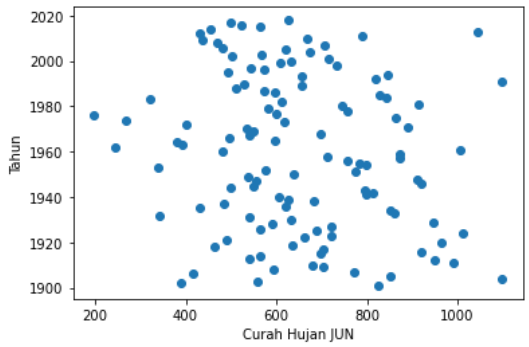
2.1 *DATASET DAN TAHAP PREPROCESSING DATA*

Meskipun *dataset* telah tersedia, namun untuk memastikan kebersihan data maka tahap *preprocessing* tetap dilaksanakan. Untuk *dataset* sendiri diunduh dari halaman https://www.kaggle.com/mukulthakur177/flood-prediction-model/data?select=kerala.csv berisi data banjir Negara Bagian Kerala. Berikut adalah 10 data teratas pada *dataset* :

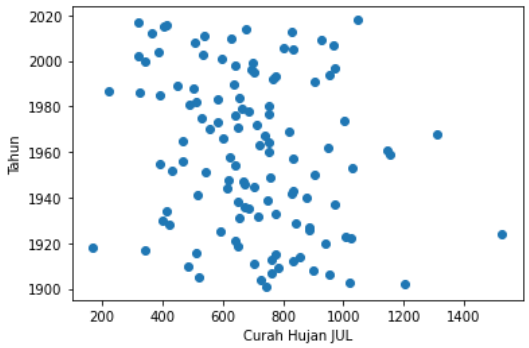
SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	ANNUAL_RAINFALL	FLOODS
KERALA	1901	28.7	44.7	51.6	160.0	174.7	824.6	743.0	357.5	197.7	266.9	350.8	48.4	3248.6	YES
KERALA	1902	6.7	2.6	57.3	83.9	134.5	390.9	1205.0	315.8	491.6	358.4	158.3	121.5	3326.6	YES
KERALA	1903	3.2	18.6	3.1	83.6	249.7	558.6	1022.5	420.2	341.8	354.1	157.0	59.0	3271.2	YES
KERALA	1904	23.7	3.0	32.2	71.5	235.7	1098.2	725.5	351.8	222.7	328.1	33.9	3.3	3129.7	YES
KERALA	1905	1.2	22.3	9.4	105.9	263.3	850.2	520.5	293.6	217.2	383.5	74.4	0.2	2741.6	NO
KERALA	1906	26.7	7.4	9.9	59.4	160.8	414.9	954.2	442.8	131.2	251.7	163.1	86.0	2708.0	NO
KERALA	1907	18.8	4.8	55.7	170.8	101.4	770.9	760.4	981.5	225.0	309.7	219.1	52.8	3671.1	YES
KERALA	1908	8.0	20.8	38.2	102.9	142.6	592.6	902.2	352.9	175.9	253.3	47.9	11.0	2648.3	NO
KERALA	1909	54.1	11.8	61.3	93.8	473.2	704.7	782.3	258.0	195.4	212.1	171.1	32.3	3050.2	YES
KERALA	1910	2.7	25.7	23.3	124.5	148.8	680.0	484.1	473.8	248.6	356.6	280.4	0.1	2848.6	NO

Gambar 2. Dataset Banjir

Jika melihat dataset, sebenarnya terdapat 4 (empat) bulan yang menjadi puncak terjadinya curah hujan tinggi yakni bulan JUN, JUL, AUG, dan SEP. Berikut adalah graphic jumlah curah hujan pada Bulan JUN(Juni) dan Bulan JUL(Juli) tersebut.



Gambar 3. Curah hujan Bulan JUN atau Juni



Gambar 4. Curah hujan Bulan JUL atau Juli

Data tersebut kemudian masuk ke tahap data preprocessing. Tahap preprocessing penting mengingat tujuan dari data preprocessing yakni untuk memperoleh kumpulan data yang dianggap benar untuk dapat dilakukan proses data mining (Salvador et al., 2016). Menurut Hua Yin & Cake Gai (2015) dalam melakukan clasifikasi dataset (Yin & Gai, 2015), ketika dataset tidak balance, maka akurasi dalam klasifikasi menjadi tidak baik(Yin & Gai, 2015). Dalam penelitian ini tahap preprocessing yakni memastikan bahwa tidak terdapat data yang null. Setelah dipastikan tidak terdapat null, selanjutnya yakni pelebelan(Netscher & Eder, 2018) dataset. Dataset tersebut telah diberi label (Netscher & Eder, 2018). Sehingga tidak diperlukan pelabelan lebih lanjut. Namun label untuk class masih berupa huruf yakni YES dan NO(Naik et al., 2021). Sehingga perlu untuk dilakukan perubahan sehingga class berupa huruf dirubah menjadi angka dalam hal ini adalah 0 dan 1(Naik et al., 2021).

Setelah tahap preprocessing, tahap selanjutnya adalah membangun model (*building model*). Tahap ini bertujuan untuk menentukan alternatif algoritma lain dari penelitian sebelumnya yang menggunakan Logistic Regression. Sehingga dalam tahap ini menggunakan Naïve Bayes sebagai algoritma. Pemilihan Naïve Bayes sebagai algoritma merujuk pada beberapa penelitian terdahulu yang menyatakan bahwa Naïve Bayes efektif dalam menyelesaikan permasalahan menggunakan *data mining* dengan akurasi mencapai 89% (Ordoñez et al., 2018). Atau penelitian lain diluar kebencanaan yang dilakukan oleh Rumini & Norhikmah (2019) yang menarik kesimpulan bahwa Naïve Bayes mampu memberikan prediksi kelulusan sebesar 77,97% (Rumini & Norhikmah, 2019).

2.2 KLASIFIKASI DENGAN NAÏVE BAYES

Algoritma Naïve Bayes merupakan metode untuk menyelesaikan permasalahan klasifikasi (Ordoñez et al., 2018). Dalam menyelesaikan klasifikasi, algoritma ini dianggap efisien (Ordoñez et al., 2018). Untuk menyelesaikan permasalahan, formula yang digunakan oleh Naïve Bayes adalah sebagai berikut (Zhang, 2004):

$$P(y|x_1,\dots,x_n) = \frac{P(y)P(x_1,\dots,x_n|y)}{P(x_1,\dots,x_n)} \quad (1)$$

Keterangan :

x : data dengan class yang belum diketahui

c : hipotesis data x merupakan suatu class spesifik

P(c|x) : probabilitas hipotesis c berdasar kondisi x (posteriori probability)

P(c) : probabilitas hipotesis c (prior probability)

P(x|c) : probabilitas x berdsar kondisi pada hipotesis c

P(x) : probabilitas dari x

Konsep yang digunakan oleh naïve bayes yakni menghitung sekumpulan probabilitas dengan menjumlahkan frekuensi dan kombinasi nilai dari dataset yang diberikan dalam melakukan pengklasfikasian (Rumini & Norhikmah, 2019). Konsep Naïve Bayes sendiri pertama dikemukakan oleh Thomas Bayes (Bustami, 2013). Konsep ini yakni proses pengklasifikasian dengan metode probabilitas dan statistik(Bustami, 2013).

Salah satu algoritma dalam Naïve Bayes adalah Gausian Naïve Bayes (https://scikit-learn.org/, n.d.). Dalam

algoritma Gausian Naïve Bayes, formulai yang digunakan adalah sebagai berikut (https://scikit-learn.org/, n.d.):

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} exp\left(-\frac{(x_i-\mu_y)^2}{2\sigma_y^2}\right) \quad (2)$$

2.3 *LIBRARY NAÏVE BAYES* DAN *RAPIDMINER*

Untuk melakukan analisis dengan Naïve bayes terdapat berbagai cara. Dalam penelitian ini akan menggunakan 2 (dua) cara yakni Menggunakan *library* Naïve Bayes dari scikit-learn.org (https://scikit-learn.org/, n.d.), dan menggunakan RapidMiner(Kotu & Deshpande, 2014). Dalam proses pengklasifikasian, kedua cara tersebut sebenarnya sepenuhnya hampir semua tahap dalam melakukan tidak diketahui secara pasti apa yang terjadi terhadap dataset yang kita masukkan. Namun demikian, harapannya dari informasi yang diberikan dapat diketahui apkaah algoritma Naïve Bayes cukup efektif untuk diterapkan dengan *dataset* yang dinakan dalam penelitian.

Penggunaan *library* dari scikit-learn.org (https://scikit-learn.org/, n.d.), perlu untuk menggunakan bahasa python. Hal ini dapat kita lihat dari struktur dan code yang terdapat pada library tersebut dan cara penggunaanya. Berikut adalah kode program untuk menggunakan *library* dari scikit-learn.org (https://scikit-learn.org/, n.d.):

```
from sklearn.naive_bayes import
GaussianNB
gnb = GaussianNB()
y_pred = gnb.fit(X_train,
y_train).predict(X_test)
```

Kode diatas akan secara otomatis melakukan klasifikas dan menyimpan hasil klasifikasi ke dalam variabel y_pred. Kode tersebut juga menggunakan Gaussian Naïve Bayes dengan formula yang telah dijabarkan sebelumnya(https://scikit-learn.org/, n.d.).

Selain menggunakan *library* dari scikit-learn.org, Untuk mendapatkan ketepatan informasi Naïve Bayes. Maka *dataset* yang sama diuji juga dengan menggunakan perangkat lunak RapidMiner. RapidMiner sendiri merupakan perangkat lunak yang digunakan untuk melakukan data mining dan analisis prediksi (Kotu & Deshpande, 2014). Adapun proses yang terjadi pada RapirMiner juga sama tidak diketahui informasi detail tahap demi tahap proses yang terjadi pada *dataset*. Dalam proses melakukan klasifikasi dengan Naïve Bayes, sebelumnya data di *split* menjadi 2 (dua) bagian yang nantinya dijadikan sebagai *data training* dan *data test*. Untuk melakukan proses *splitting* sendiri menggunakan perbandingan 0.5 : 0.5 yang berarti 50% sebagai *data training* dan 50% sebagai *data test*. Sedangkan tahap *splitting* pada RapidMiner menggunakan metode stratified sampling. Pemilihan penggunaan metode ini mengingat ini merupakan metode probability yang sering digunakan untuk proses sampling (Parsons, 2014).

Dengan demikian, proses *training* menggunakan 2 (dua) *tools* yakni menggunakan *library* Naïve Bayes dari scikit-learn.org dan menggunakan RapidMiner. Sedangkan data yang dilakukan training adalah data dari tahun 1901 sampai dengan tahun 2018.

3. HASIL DAN PEMBAHASAN

3.1 *DATA PREPROCESSING DAN PENGGUNAAN LIBRARY GAUSIAN NAÏVE BAYES*

Sesuai dengan tahapan dalam penelitian, maka tahap pertama adalah melakukan *preprocessing* data. Tahap ini menggunakan python untuk melakukan pengecekan apakah terdapat data yang null, dan hasil dari pengecekan adalah tidak terdapat data yang null.

```
#Now we will cheak if any colomns is left empty
data.apply(lambda x:sum(x.isnull()), axis=0)
```

```
SUBDIVISION          0
YEAR                 0
JAN                  0
FEB                  0
MAR                  0
APR                  0
MAY                  0
JUN                  0
JUL                  0
AUG                  0
SEP                  0
OCT                  0
NOV                  0
DEC                  0
 ANNUAL RAINFALL     0
FLOODS               0
dtype: int64
```

Gambar 5. Cek Dataset tidak terdapat null

Setelah diketahui tidak terdapat data yang null, maka tahap selanjutnya adalah melakukan perubahan pada kolom FLOODS dengan merubah kata YES menjadi 1 dan NO menjadi 0. Proses perubahan ini terlihat pada gambar berikut :

```
#We want the data in numbers, therefore we will replace the yes/no in floods colounm by 1/0
data['FLOODS'].replace(['YES','NO'],[1,0],inplace=True)

#Let's see how are data looks like now
data.head()
```

	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	ANNUAL RAINFALL	FLOODS
0	KERALA	1901	28.7	44.7	51.6	160.0	174.7	824.6	743.0	357.5	197.7	266.9	350.8	48.4	3248.6	1
1	KERALA	1902	6.7	2.6	57.3	83.9	134.5	390.9	1205.0	315.8	491.6	358.4	158.3	121.5	3326.6	1
2	KERALA	1903	3.2	18.6	3.1	83.6	249.7	558.6	1022.5	420.2	341.8	354.1	157.0	59.0	3271.2	1
3	KERALA	1904	23.7	3.0	32.2	71.5	235.7	1098.2	725.5	351.8	222.7	328.1	33.9	3.3	3129.7	1
4	KERALA	1905	1.2	22.3	9.4	105.9	263.3	850.2	520.5	293.6	217.2	383.5	74.4	0.2	2741.6	0

Gambar 6. Tahap perubahan label YES dan NO menjadi 1 dan 0

Setelah data diberi label, sebenarnya tahap ini telah selesai. Namun berhubung akan dilakukan pengujian dengan RapidMiner, maka langkah selanjutnya *eksport dataset* dalam bentuk csv yang nantinya akan digunakan pada pengujian menggunakan RapidMiner. Sebelum langkah tersebut, terlebih dahulu dilakukan penghapusan kolom SUBDIVISION yang merupakan data berupa string. Detail penghapusan dan export data dapat dilihat pada gambar berikut :

```
#Let's see how are data looks like now
#data.head()
#ini difungsikan untuk convert data ke csv setelh direplace floodsnya
df = pd.DataFrame(data)
df = df.drop('SUBDIVISION', 1)
#print(df)
df.to_csv('data_cek.csv', sep='\t')
```

Gambar 7. Hapus kolom SUBDIVISION dan convert datake csv

Sesuai dengan informasi sebelumnya setelah data di beri label, dilanjutkan dengan membagi 2 (dua) data (Naik et al., 2021) yakni dari kolom ke 2 (dua) hingga kolom 15(Naik et al., 2021) dan disimpan pada variabel x.

```
#Now let's seperate the data which we are gonna use for prediction
x = data.iloc[:,1:14]
x.head()
```

	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
0	1901	28.7	44.7	51.6	160.0	174.7	824.6	743.0	357.5	197.7	266.9	350.8	48.4
1	1902	6.7	2.6	57.3	83.9	134.5	390.9	1205.0	315.8	491.6	358.4	158.3	121.5
2	1903	3.2	18.6	3.1	83.6	249.7	558.6	1022.5	420.2	341.8	354.1	157.0	59.0
3	1904	23.7	3.0	32.2	71.5	235.7	1098.2	725.5	351.8	222.7	328.1	33.9	3.3
4	1905	1.2	22.3	9.4	105.9	263.3	850.2	520.5	293.6	217.2	383.5	74.4	0.2

Gambar 8. Pengambilan data dari kolom 2 (dua) hingga kolom 15

Sedangkan kolom ke 16 yang merupakan kolom label disimpan pada variabel yang berbeda yakni variabel **y**.

```
#Now seperate the flood label from the dataset
y = data.iloc[:, -1]
y.head()
```

```
0    1
1    1
2    1
3    1
4    0
Name: FLOODS, dtype: int64
```

Gambar 9. Data pada kolom 16 disimpan ke variabel y

Dengan telah diselesaikannya tahap *preprocessing*, maka data yang telah terkumpul siap diterapkan ke *machine learning* (Dwiasnati & Devianto, 2021).

Menggunakan library Gausian Naïve Bayes dari scikit-learn.org diperoleh akurasi sebesar 79,16%. Detail keluaran adalah sebagai berikut :

```
from sklearn.naive_bayes import GaussianNB
gnb = GaussianNB()
y_pred = gnb.fit(x_train, y_train).predict(x_test)
print("Number of mislabeled points out of a total %d points : %d" % (x_test.shape[0], (y_test != y_pred).sum()))
```

```
Number of mislabeled points out of a total 24 points : 5
```

```
from sklearn.metrics import accuracy_score,recall_score,roc_auc_score,confusion_matrix
print("\naccuracy score:%f"%(accuracy_score(y_test,y_pred)*100))
```

```
accuracy score:79.166667
```

Gambar 10. Gaussian Naïve Bayes

Untuk melihat perbandingan antara hasil pelabelan yang diberikan oleh naïve bayes dengan label sebelumnya, berikut adalah perbedaannya :

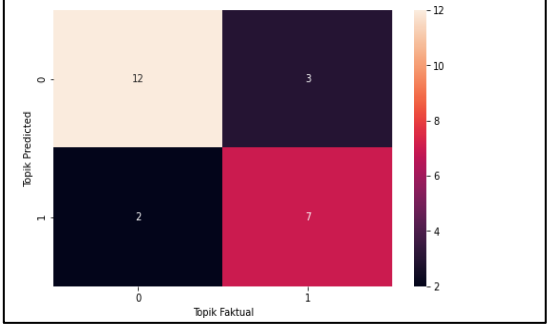
```
print("Label Hasil Prediksi Naive Bayes\n")
print(y_pred)
print("\n Lable Sebelumnya\n")
print(y_test.values)
```

```
Label Hasil Prediksi Naive Bayes

[1 0 1 0 1 0 1 1 1 1 1 1 0 1 1 1 1 0 1 1 0 1 1 1 0]

 Lable Sebelumnya

[0 1 1 0 1 0 1 1 0 1 0 1 0 1 1 1 1 0 0 0 1 0 1 0]
```

Gambar 11. Hasil pelabelan setelah dan sebelum Naïve Bayes

Dengan informasi hasil prediksi, selanjutnya diperlihatkan perbandingan antara hasil prediksi dan kebenaran informasi :

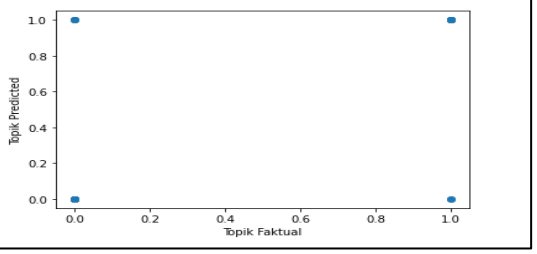
```
import seaborn as sns
from sklearn.metrics import confusion_matrix
import matplotlib.pyplot as plt
f, ax = plt.subplots(figsize=(8,5))
sns.heatmap(confusion_matrix(y_test, y_pred),
annot=True, fmt=".0f", ax=ax)
plt.xlabel("Topik Faktual")
plt.ylabel("Topik Predicted")
plt.show()
```



Gambar 12. Matrix perbandingan hasil prediksi dan keadaan sebenarnya

Untuk plotting data hasil prediksi dan keadaan sebenarnya terdapat terlihat pada gambar berikut :

```
#Scatter Plot
plt.scatter(y_test, y_pred)
plt.xlabel("Topik Faktual")
plt.ylabel("Topik Predicted")
plt.show()
```



Gambar 13. Plotting data

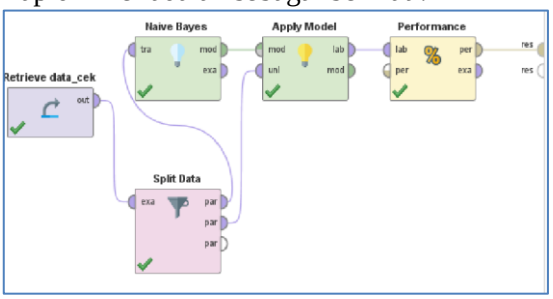
3.2 Penggunaan RapidMiner algoritma Naïve Bayes

Data csv yang sebelumnya telah di peroleh merupakan data dari kolom YEARS hingga kolom FLOODS. Adapun 10 data teratas adalah sebagai berikut :

YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	ANNUAL_RAINFALL	FLOODS
1901	28.7	44.7	51.6	160.0	174.7	824.6	743.0	357.5	197.7	266.9	350.8	48.4	3248.6	1
1902	6.7	2.6	57.3	83.9	134.5	390.9	1205.0	315.8	491.6	358.4	158.3	121.5	3326.6	1
1903	3.2	18.6	3.1	83.6	249.7	558.6	1022.5	420.2	341.8	354.1	157.0	59.0	3271.2	1
1904	23.7	3.0	32.2	71.5	235.7	1098.2	725.5	351.8	222.7	328.1	33.9	3.3	3129.7	1
1905	1.2	22.3	9.4	105.9	263.3	850.2	520.5	293.6	217.2	383.5	74.4	0.2	2741.6	0
1906	26.7	7.4	9.9	59.4	160.8	414.9	954.2	442.8	131.2	251.7	163.1	86.0	2708.0	0
1907	18.8	4.8	55.7	170.8	101.4	770.9	760.4	981.5	225.0	309.7	219.1	52.8	3671.1	1
1908	8.0	20.8	38.2	102.9	142.6	592.6	902.2	352.9	175.9	253.3	47.9	11.0	2648.3	0
1909	54.1	11.8	61.3	93.8	473.2	704.7	782.3	258.0	195.4	212.1	171.1	32.3	3050.2	1
1910	2.7	25.7	23.3	124.5	148.8	680.0	484.1	473.8	248.6	356.6	280.4	0.1	2848.6	0

Gambar 14. Dataset Setelah *Preprocessing*

Menggunakan data csv yang telah tersedia selanjutnya dilakukan proses mining data dengan RapidMiner. Secara keseluruhan proses yang dilakukan dengan menggunakan RapidMiner adalah sebagai berikut :



Gambar 15. Proses RapidMiner

accuracy: 98.31%

	true KERALA	true KERALA	class precision
pred. KERALA	58	1	98.31%
pred. KERALA	0	0	0.00%
class recall	100.00%	0.00%	

Gambar 16. Akurasi Naïve Bayes RapidMiner

Tahap *splitting* data sesuai dengan penjelasan sebelumnya menggunakan stratified sampling. Sedang hasil yang diperoleh dengan menggunakan RapidMiner, diperoleh informasi bahwa akurasi algoritma Naïve Bayes sebesar 98,31%.

4. SIMPULAN

Adanya perbedaan yang signifikan antara akurasi yang diperoleh dengan menggunakan library Gausian Naïve Bayes dari scikit-learn.org dan RapidMiner menjadikan sulit untuk disimpulkan apakah Naïve Bayes efektif untuk diterapkan pada dataset banjir. Namun secara keseluruhan, jika dibandingkan dengan penelitian sebelumnya yakni "Flood Prediction using Logistic Regression for Kerala State"(Naik et al., 2021) dengan akurasi mencapai 75%(Naik et al., 2021) maka prediksi banjir dengan dataset yang sama menggunakan algoritma Naïve Bayes dan library Gausian Naïve Bayes masih belum begitu signifikan hanya 79,16%. Keluaran berbeda terjadi ketika menggunakan RapidMiner sebagai analisis algoritma Naïve Bayes, akurasi yang diberikan mencapai 98,31%. Hal ini yang menjadi alasan kesulitan untuk menarik kesimpulan yang tepat apakah Naïve Bayes lebih akurat jika dibandingkan dengan Logistic Regression(Naik et al., 2021) yang telah melakukan penelitian terdahulu. Dengan adanya hasil penelitian ini, maka perlu untuk dilakukan kajian lebih lanjut sehingga tahap demi tahap ketika menggunakan algoritma Naïve Bayes pada dataset banjir, dapat memberikan informasi yang tepat letak permasalahan yang terjadi. Harapannya selain membandingkan akurasi antar algoritma pada data mining, juga dapat menambah kekayaan keilmuan dalam bidang data mining.

Dari sisi lain tidak dapat dipungkiri, proses splitting dataset yang berbeda tidak menutup kemungkinan mengakibatkan

perbedaan keluaran. Splitting yang dilakukan dengan menggunakan python secara jelas memperlihatkan kolom - kolom yang dijadikan sebagai data test maupun data training. Adanya proses splitting dataset untuk dijadikan sebagai data test dari kolom ke 2 (dua) sampai kolom ke 15 menggunakan python dan kemudian menjadikan data training untuk kolom yang berada di kolom ke 16. Tidak menutup kemungkinan menjadikan library Gausian Naïve Bayes memberikan keluaran akurasi yang berbeda dengan ketika menggunakan RapidMiner. Hal ini terlihat ketika menggunakan RapidMiner, sangat berbeda dalam melakukan proses splitting. Proses splitting yang dilakukan oleh RapidMiner tidak diketahui kolom ataupun baris yang di jadikan sebagai data test maupun data training. Adanya informasi ini, dapat menjadi rujukan untuk perlu dilakukan splitting diawal untuk mendapatkan data test maupun data training sebelum dilakukan proses analisis menggunakan Naïve Bayes sehingga harapannya akan memberikan keluaran yang tepat.

5. DAFTAR PUSTAKA

Arjun Singh, & Saxena, A. (2016). A Hybrid Data Model for Prediction of Disaster using Data Mining Approaches. *International Journal of Engineering Trends and Technology (IJETT)*, 384–392.

Azure, M. (2021, November 4). Machine Learning Algorithm Cheat Sheet for Azure Machine Learning designer. https://docs.microsoft.com/enus/azure/machine-learning/algorithm-cheat-sheet

Bustami. (2013). Penerapan Algoritma Naive Bayes Untuk Mengklasifikasi Data Nasabah Asuransi. *TECHSI : Jurnal Penelitian Teknik Informatika*, *3*, 127–146.

Dwiasnati, D., & Devianto, Y. (2021). Optimasi Prediksi Bencana Banjir menggunakan Algoritma SVM untuk penentuan Daerah Rawan Bencana Banjir. *Prosiding Seminar Nasional Sistem Informasi Dan Teknologi (SISFOTEK)*, *5*, 202–207.

Han, j, & Kamber, M. (2001). Data Mining: Concepts and Techniques. Morgan Kaufmann Publishers.

https://scikit-learn.org/. (n.d.). *Naive Bayes*. Retrieved November 20, 2021, from https://scikit-learn.org/stable/modules/naive_bayes.html

Kotu, V., & Deshpande, B. (2014). Predictive Analytics and Data Mining Concepts and Practice with RapidMiner. Morgan Kaufmann is an imprint of Elsevier.

Mukul. (2020). *Flood Prediction Model*. https://www.kaggle.com/mukulthakur177/flood-prediction-model/

Naik, S., Verma, A., Ashok Patil, S., & Hingmire, A. (2021). Flood Prediction using Logistic Regression for Kerala State. *International Journal of Engineering Research & Technology*, 36–38.

Netscher, S., & Eder, C. (2018). Variable Definition and Composition of the Dataset. In: Data Processing and Documentation: Generating High Quality Research Data in Quantitative Social Science Research. *GESIS Papers*, 15–24.

Nugroho, S. P. (2002). Evaluasi Dan Analisis Curah Hujan Sebagai Faktor Penyebab Bencana Banjir Jakarta. *Urnal Sains & Teknologi Modifikasi Cuaca*, *3*, 91–97.

Ordoñez, A. J., Paje, R. E. J., & Naz, R. N. (2018). SMS Classification Method for Disaster Response using Naïve Bayes Algorithm. *2018 International Symposium on Computer, Consumer and Control (IS3C)*, 233–236. https://doi.org/10.1109/IS3C.2018.00066

Panafrican Emergency Training Centre. (2002). *Disasters & Emergencies Definitions*.https://apps.who.int/disasters/repo/7656.pdf

Parsons, VanL. (2014). Stratified Sampling. *WileyStatsRef:StatisticsReferenceOnline*,111.https://doi.org/10.1002/9781118445112

Prihandoko, Bertalya, & Ramadhan, M. I. (2017). An Analysis of Natural Disaster Data by Using K-Means and K- Medoids Algorithm of Data Mining Techniques. *2017 15th International Conference on Quality in Research (QiR) : International Symposium on Electrical and Computer Engineering*, 221-225.https://doi.org/10.1109/QIR.2017.8

168485

Refonaa, j, Lakshmi, M., & Vivek, v. (2015). Analysis and Prediction Of natural Disaster Using Spatial Data Mining Technique. *2015 International Conference on Circuit, Power and Computing Technologies [ICCPCT]*, 16.https://doi.org/10.1109/ICCPCT.2015.7159379

Rumini, & Norhikmah. (2019). Prediksi Kegagalan Siswa Dalam Data Mining Dengan Menggunakan Metode Naïve Bayes. *Jurnal Mantik Penusa*, *3*, 42–46.

Salvador, G. ıa, Julián, L., & Francisco, H. (2016). Tutorial on Practical Tips of the Most Influential Data Preprocessing Algorithms in Data Mining. *Knowledge-Based Systems*, *98*, 1–29.

Sesa, W., shalih, O., Syauqi, W.Adi, A., Z.Shabrina, F., Rizqi, A., Widiastomo, Y., S.Putra, A., Karimah, R., Eveline, F., Alfian, A., Hafiz, A., BAgaskoro, Y., Nomita Dewi, A., & Rahmawati, I. (2020). *Index Risiko Bencana Indonesia Tahun 2020*. Badan Nasional PenanggulanganBencana.https://inarisk.bnpb.go.id/pdf/BUKU%20IRBI%202020%20KP.pdf

Yin, H., & Gai, K. (2015). An Empirical Study on Preprocessing High-dimensional Class-imbalanced Data for Classification. *IEEE 17th International Conference on High Performance Computing and Communications (HPCC)*,1419.https://doi.org/10.1109/HPCC-CSS-ICESS.2015.205

Zhang, H. (2004). The Optimality of Naive Bayes. *Proceedings of 17th International Florida Artificial Intelligence Research Society Conference*, 562–567.