

Laporan Tugas 4 Machine Learning

A. Analisis Masalah

Terdapat file TrainsetTugas4ML.xls yang berisi 298 objek data yang memiliki 2 atribut input yaitu X1 dan X2 dan 1 output kelas yang memiliki 2 label (1 atau 2). Terdapat pula file TestSetTugas4ML.xls yang berisi 75 objek data yang belum memiliki label kelas. Akan dibangun sebuah sistem klasifikasi menggunakan metode bagging yaitu salahsatu teknik ensemble learning dengan berbasis naive bayes untuk menentukan label dari data-data tersebut. Label-label tersebut akan disimpan pada file yang bernama TebakanTugas4ML.csv.

B. Strategi Penyelesaian Masalah

Strategi yang dilakukan untuk menentukan label dari 75 data tersebut adalah membuat program yang menerapkan metode bagging. Bahasa pemrograman yang digunakan adalah java. Langkah-langkahnya sebagai berikut :

- Membaca file data train dan test menggunakan function scanner

```
File filetrain= new File("TrainsetTugas4ML.xls");
File filetest= new File("TestsetTugas4ML.xls");
Scanner tes= new Scanner(filetest);
tes.nextLine();
Scanner tra = new Scanner(filetrain);
tra.nextLine();
```

- Membangun 7 model subhimpunan dari data train menggunakan metode bootstrap dengan distribusi seragam. Jadi setiap index pada array train memiliki peluang yang sama.

```
static String[][] subhimpunan(String[][] train){
    String[][] model=new String[298][3];
    for (int i = 0; i < model.length; i++) {
        Random rand = new Random();
        int r = rand.nextInt(298);
        model[i] = train[r];
    }
    return model;
}

model1 = subhimpunan(train);
model2 = subhimpunan(train);
model3 = subhimpunan(train);
model4 = subhimpunan(train);
model5 = subhimpunan(train);
model6 = subhimpunan(train);
model7 = subhimpunan(train);
```

- Setiap index pada data tes akan ditentukan label kelasnya untuk setiap model, untuk menentukan labelnya 1 atau 2 maka digunakan metode naive bayes.

```
while (tes.hasNext()) {
    String[] test = tes.nextLine().split(",");
    double label1 = getLabel(model1, test);
    double label2 = getLabel(model2, test);
    double label3 = getLabel(model3, test);
    double label4 = getLabel(model4, test);
    double label5 = getLabel(model5, test);
    double label6 = getLabel(model6, test);
    double label7 = getLabel(model7, test);
    i++;
}
```

Karena data bernilai kontinu maka digunakan rumus naive bayes :

$$P(x_k|C_i) = \frac{1}{\sigma_{ik}\sqrt{2\pi}} e^{-\frac{(x_k - \mu_{ik})^2}{2\sigma_{ik}^2}}$$

```
static double naivebayes(String[][] train, String b, String[] test){
    double a = prior2(train, b);
    double mean0 = mean(train, 0, b); double mean1 = mean(train, 1, b);
    double std0 = std(train, 0, b); double std1 = std(train, 1, b);
    return a*(Math.pow(Math.E, -(Math.pow((Double.parseDouble(test[0])-mean0)/std0, 2)/2)))/std0*Math.sqrt(2*Math.PI)*
        (Math.pow(Math.E, -(Math.pow((Double.parseDouble(test[1])-mean1)/std1, 2)/2)))/std1*Math.sqrt(2*Math.PI);
}
```

Mean adalah nilai rata-rata dari setiap atribut input yang memiliki kelas yang sama

```
static double mean(String[][] train, int a, String b){
    double c = 0;
    int d = 0;
    for (int i = 0; i < train.length; i++) {
        if (b.equals(train[i][2])){
            c = c + Double.parseDouble(train[i][a]);
            d++;
        }
    }
    return (double) c/d;
}
```

Standar deviasi adalah sebaran nilai-nilai dari data atribut input yang memiliki kelas yang sama.

```
static double std(String[][] train, int a, String b){
    double rata = mean(train,a,b);
    double c = 0;
    int d = 0;
    double f = 0;
    for (int i = 0; i < train.length; i++) {
        if (b.equals(train[i][2])){
            d++;
            c += Math.pow((Double.parseDouble(train[i][a])-rata), 2);
        }
    }
    return Math.sqrt(c/(d-1));
}
```

- Voting label terbanyak dari 7 label yang yang didapatkan dari 7 model. Label yang terpilih menjadi label dari index data tes tersebut dan disimpan ke file TebakanTugas4ML.csv.

```
atic String voting(double label1, double label2, double label3, double label4, double label5, double label6, double label7){
    double a = label1+label2+label3+label4+label5+label6+label7;
    if (a<11) return "1";
    else return "2";
}

try{
    fw.append(String.valueOf(voting(label1,label2,label3,label4,label5,label6,label7))+'\n');
}catch(Exception e){}
}
fw.flush();
fw.close();
}
```

Output :

1	2	20	1	39	1	58	1
2	2	21	2	40	1	59	1
3	2	22	1	41	1	60	1
4	2	23	2	42	2	61	1
5	2	24	1	43	1	62	1
6	2	25	1	44	1	63	1
7	2	26	1	45	1	64	1
8	2	27	1	46	1	65	1
9	2	28	2	47	1	66	1
10	2	29	1	48	1	67	1
11	2	30	2	49	1	68	1
12	1	31	2	50	1	69	1
13	2	32	1	51	1	70	1
14	1	33	2	52	1	71	1
15	1	34	1	53	1	72	1
16	1	35	1	54	1	73	1
17	1	36	1	55	1	74	1
18	1	37	1	56	1	75	1
19	1	38	1	57	1		