

# Laporan Tugas 1

## CSH3L3 Pembelajaran Mesin

Oleh: Raihan Hamid Suraperwata (1301160466)

### A. Analisis Masalah

Permasalahan yang dibahas kali ini adalah mengenai algoritma naive bayes yaitu sistem klasifikasi menggunakan metode **Naïve Bayes** untuk menentukan kelas/label data testing dalam Testset. Adapun permasalahan yang diberikan adalah sebagai berikut.

Diberikan sebuah Trainset berupa himpunan data berisi 160 objek data yang memiliki 7 atribut input (age, workclass, education, marital-status, occupation, relationship, hours-per-week) dan 1 output (label kelas income) yang memiliki 2 kelas/label (>50K, dan <=50K). Bangunlah sebuah sistem klasifikasi menggunakan metode Naïve Bayes untuk menentukan kelas/label data testing dalam Testset. Sistem membaca masukan file TrainsetTugas1ML.csv dan TestsetTugas1ML.csv dan mengeluarkan output berupa file TebakanTugas1ML.csv berupa satu kolom berisi 40 baris yang menyatakan kelas/label baris yang bersesuaian pada file TestsetTugas1ML.csv.

Pada permasalahan yang telah diberikan, kita mengetahui bahwa *output* yang ada pada TebakanTugas1ML.csv akan dihasilkan berdasarkan data yang diberikan pada testSet yang berada pada file DataTugas2.csv

### B. Metode Penyelesaian

Metode yang digunakan untuk menyelesaikan permasalahan ini adalah algoritma Naive Bayes. Naive Bayes classifier (NBC) merupakan salah satu metode pembelajaran mesin yang memanfaatkan perhitungan probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi probabilitas di masa depan berdasarkan pengalaman di masa sebelumnya.

Program Naive Bayes yang dibangun untuk menyelesaikan permasalahan ini menggunakan bahasa pemrograman *Javascript* dan platform Node.js serta Atom IDE sebagai *text editor*. Untuk melihat hasil program Naive Bayes yang telah dibangun dapat melakukan kompilasi program sesuai dengan petunjuk pada readme.md

### C. Penjelasan Fungsi dari Metode yang digunakan

Dengan menggunakan algoritma Naive Bayes, maka kita perlu mengimport trainset dan dataset yang telah diberikan.

Setelah melakukan import data trainset dan testset, langkah berikutnya adalah menjumlahkan seluruh data yang ada pada trainset untuk dihitung berapa peluang data yang memiliki income >50K dan juga <=50K, dalam hal ini peluang data tersebut disingkat menjadi  $P(c)$ .

```

8  function main(trainSet, testSet){
9      up = trainSet.filter(function(x) {return x.income == '>50K';}).length
10     don = trainSet.filter(function(x) {return x.income == '<=50K';}).length
11     upper = up / trainSet.length
12     under = don / trainSet.length
13     result = []
14

```

Setelah kita mengetahui peluang masing-masing income, hal yang perlu dilakukan selanjutnya adalah melakukan perulangan dari data testset untuk dprediksi hasil income yang dimiliki. Pada perulangan, program akan menghitung peluang yang dimiliki oleh masing-masing atribut input dan dikalikan dengan  $P(c)$ . Hasil perhitungan tersebut akan disimpan pada *array* result dan akan disimpan pada file TebakanTugas1ML.csv

```

15     for (var data in testSet) {
16         pAge1 = trainSet.filter(function(x) {return x.income == '>50K' && x.age == t
17         pAge2 = trainSet.filter(function(x) {return x.income == '<=50K' && x.age ==
18         pWorkclass1 = trainSet.filter(function(x) {return x.income == '>50K' && x.wo
19         pWorkclass2 = trainSet.filter(function(x) {return x.income == '<=50K' && x.w
20         pEducation1 = trainSet.filter(function(x) {return x.income == '>50K' && x.ed
21         pEducation2 = trainSet.filter(function(x) {return x.income == '<=50K' && x.e
22         pMarital1 = trainSet.filter(function(x) {return x.income == '>50K' && x['mar
23         pMarital2 = trainSet.filter(function(x) {return x.income == '<=50K' && x['ma
24         pOccupation1 = trainSet.filter(function(x) {return x.income == '>50K' && x.o
25         pOccupation2 = trainSet.filter(function(x) {return x.income == '<=50K' && x.
26         pRelation1 = trainSet.filter(function(x) {return x.income == '>50K' && x.rel
27         pRelation2 = trainSet.filter(function(x) {return x.income == '<=50K' && x.re
28         pTime1 = trainSet.filter(function(x) {return x.income == '>50K' && x['hours-
29         pTime2 = trainSet.filter(function(x) {return x.income == '<=50K' && x['hours
30
31         p1 = pAge1 * pWorkclass1 * pEducation1 * pMarital1 * pOccupation1 * pRelatio
32         p2 = pAge2 * pWorkclass2 * pEducation2 * pMarital2 * pOccupation2 * pRelatio
33         // console.log(p1 + " - " + p2)
34
35         if (p1 > p2) result.push(['>50K'])
36         else result.push(['<=50K'])
37     }
38

```

Hasil dari program tersebut dapat dilihat pada file TebakanTugas1ML.csv yang berisi 40 baris data prediksi income.