LAPORAN ANALISIS DATA CARDIOVASCULAR (CVD) DENGAN REGRESI LOGISTIK

Nama	:	Brian Stefano
NIM	:	10818037
Kelompok	:	15
Topik	:	Regresi Logistik
Hari/Tanggal	:	Selasa, 18 Mei 2021

PROGRAM STUDI S1 AKTUARIA

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

INSTITUT TEKNOLOGI BANDUNG

2021

BAB I

PENDAHULUAN

A. LATAR BELAKANG

Serangan jantung merupakan penyakit yang sering menewaskan orang – orang secara tiba – tiba. Adapun penyebab dari penyakit jantung tersebut terkadang tidak mudah diketahui dan membutuhkan waktu untuk dipastikan. Tidak jarang bahwa orang yang terkena serangan jantung tidak terselamatkan dikarenakan ganasnya penyakit jantung tersebut. Gejala – gejala yang dialami juga terkadang tidak terlalu jelas dan dalam sekejap dapat menjadi sangat berbahaya bagi keselamatan orang yang mengalaminya. Karena itu, penting untuk kita mengetahui faktor – faktor yang dapat meningkatkan kemungkinan terkenanya penyakit serangan jantung.

Data yang dianalisis pada laporan ini merupakan data CVD yang didapat dari laman Kaggle. Data tersebut merupakan data pria Afrika Selatan yang memiliki 462 observasi dan 11 kolom yang masing – masing menyimpan informasi yang akan digunakan pada analisis ini. Lebih lengkapnya dapat dilihat pada tabel di bawah ini.

Nama Kolom	Keterangan
Ind (Indeks)	Indeks orang ke-i
Sbp (Systolic Blood Pressure)	Tekanan darah sistolik (mmHg)
Tobacco	Pemakaian tembakau (Kg)
Ldl (Low density lipoprotein)	Kolesterol buruk (mmol/L)
Adiposity	*Body Adiposity Index* dari orang tersebut
Famhist (Family History)	Terdiri dari 2 level: • "Present" → Terdapat rekan keluarga yang pernah mengalami CVD • "Absent" → Rekan keluarga tidak ada yang pernah mengalami CVD
Typea (Type A behaviour)	Nilai dari tes kepribadian Type-A Type-B, semakin tinggi nilai tersebut, orang tersebut semakin memiliki kepribadian tipe A
Obesity	*Body Mass Index* dari orang tersebut
Alcohol	Konsumsi alcohol (gram/minggu)
Age	Umur saat pencatatan data
Chd	Terdiri dari 2 level: • 0 → Tidak terkena penyakit CVD • 1 → Terkena penyakit CVD

Setelah dilakukan analisis pada data CVD ini, penulis berharap bahwa pembaca dapat lebih menjaga pola hidup mereka dan menghindari hal – hal yang dapat meningkatkan peluang seseorang terkena penyakit jantung. Adapun analisis dalam laporan ini dilakukan secara individu dengan diskusi bersama rekan – rekan penulis antara lain Muhammad Agam (10818033), Anthony (10818031), Vincent Valeriandy Kencana (10818049), Gabrielle Christy (10818016) Reuven Cannarivo Buntarco (10818003), dan Ian Sebastian (10117088).

B. RUMUSAN MASALAH

1. Variabel – variabel apa saja kah yang mempengaruhi peluang seseorang terkena CVD?
2. Model apa yang digunakan untuk memprediksi data CVD yang didapat?
3. Bagaimanakah hasil prediksi yang telah dilakukan?

C. TUJUAN PENELITIAN

1. Menentukan variabel – variabel yang mempengaruhi peluang seseorang terkena CVD
2. Menentukan model terbaik yang digunakan untuk memprediksi data CVD yang didapat
3. Menjelaskan hasil prediksi yang telah dilakukan

BAB II

METODOLOGI PENELITIAN

Sebelum dilakukan analisis, data CVD dibersihkan jika dirasa perlu. Dibersihkan pada hal ini berarti kita menghapus sel yang tidak memiliki nilai pada kolom – kolom data jika ada sehingga hasil akhirnya menjadi bersih (tidak ada sel yang tidak memiliki nilai). Data yang memiliki nilai pencilan tidak dihilangkan dikarenakan data tersebut akan dikategorikan. Alasan lain yang membuat nilai pencilan tidak dihilangkan adalah pencilan tersebut mungkin dapat mempengaruhi interpretasi penulis dari grafik yang ada.

Setiap variabel dianalisis dan ditentukan manakah variabel yang relevan dengan variabel responsnya. Pada hal ini, variabel "ind" dan "adiposity" tidak digunakan karena variabel "ind" hanya merupakan indeks yang menunjukkan data ke berapa, sementara variabel "adiposity" merupakan satuan pengukuran yang mirip seperti variabel "obesity". Karena variabel "obesity" dipandang lebih bagus dalam memodelkan data ini, maka variabel "adiposity" tidak digunakan.

Kemudian, data – data yang kontinu dikategorikan menjadi beberapa level sesuai dengan standar – standar yang telah berlaku. Contohnya: nilai "sbp" yang lebih rendah dari 120 menunjukkan bahwa tekanan darah orang tersebut normal, sementara nilai "sbp" yang berada diantara 120 dan 140 menunjukkan bahwa orang tersebut sudah memasuki tahap sebelum hipertensi, dan seterusnya. Adapun informasi pengkategorian variabel kontinu tersebut lebih lengkapnya dapat dilihat pada tabel di bawah ini:

Variabel	Level kategori
Sbp	Dikategorikan menjadi 4 level: • 0 → $sbp < 120$ (Normal) • 1 → $120 < sbp < 140$ (Tahap sebelum hipertensi) • 2 → $140 < sbp < 160$ (Hipertensi tahap 1) • 3 → $sbp > 160$ (Hipertensi tahap 2)
Tobacco	Dikategorikan menjadi 4 level: • 0 → $Tobacco = 0$ (Tidak memakai tembakau) • 1 → $Tobacco \leq 5$ (Pengonsumsi tembakau ringan) • 2 → $5 < Tobacco \leq 10$ (Pengonsumsi tembakau skala sedang) • 3 → $Tobacco > 10$ (Pengonsumsi tembakau skala berat)
Ldl	Dikategorikan menjadi 5 level: • 0 → $ldl \leq 2.6$ (Ideal untuk orang yang beresiko) • 1 → $2.6 < ldl \leq 3.3$ (Hampir ideal) • 2 → $3.3 < ldl \leq 4.1$ (Di ambang batas tinggi) • 3 → $4.1 < ldl \leq 4.9$ (Tinggi) • 4 → $ldl > 4.9$ (Sangat tinggi)
Typea	Dikategorikan menjadi 4 level: • 0 → $typea \leq 47$ (Tipe A) • 1 → $47 < typea \leq 53$ (Mendekati Tipe A) • 2 → $53 < typea \leq 60$ (Mendekati Tipe B) • 3 → $type > 60$ (Tipe B)
Obesity	Dikategorikan menjadi 6 level:

	• 0 → $obesity < 18.5$ (Kekurangan berat badan) • 1 → $18.5 < obesity \leq 24.9$ (Normal) • 2 → $24.9 < obesity \leq 29.9$ (Kelebihan berat badan) • 3 → $29.9 < obesity \leq 34.9$ (Obesitas 1) • 4 → $34.9 < obesity \leq 39.9$ (Obesitas 2) • 5 → $obesity > 39.9$ (Obesitas 3)
alcohol	Dikategorikan menjadi 5 level: • 0 → $alcohol = 0$ (Tidak mengonsumsi alkohol) • 1 → $0 < alcohol \leq 48$ (Mengonsumsi alkohol skala ringan 1) • 2 → $48 < alcohol \leq 96$ (Mengonsumsi alkohol skala ringan 2) • 3 → $96 < alcohol \leq 190$ (Mengonsumsi alkohol skala sedang) • 4 → $alcohol > 190$ (Mengonsumsi alkohol skala berat)
Age	Dikategorikan menjadi 6 level: • 0 → $age < 24$ • 1 → $24 \leq age < 34$ • 2 → $34 < age \leq 44$ • 3 → $44 < age \leq 54$ • 4 → $54 < age \leq 64$ • 5 → $age > 64$

Setelah variabel – variabel kontinu dikategorikan, data akan diacak dan dipisah menjadi data *training* dan data validasi. Pada analisis ini, diambil 80% dari banyak observasi data untuk data *training* sementara untuk validasi digunakan 20% dari banyak obervasi data. Selanjutnya dicari nilai – nilai yang muncul paling banyak dari semua variabel kategorikal dan variabel tersebut dijadikan base level.

Kemudian dilakukan regresi logistik menggunakan R dan *stepwise regression* yang setelahnya didapat estimasi parameter – parameter yang diperlukan beserta tingkat signifikansi dari setiap parameter tersebut sehingga terdapat sebuah calon model yang memungkinkan kita untuk memprediksi hasil. Bentuk dari regresi logistik beserta link kanonikalnya adalah:

$$g(\mu) = \ln\left(\frac{\pi}{1-\pi}\right) = x'\beta \rightarrow \pi = \left(\frac{e^{x'\beta}}{1+e^{x'\beta}}\right)$$

Lalu kurva ROC (Receiver Operating Characteristic) diplot untuk menguji apakah model cukup baik untuk dipakai atau tidak. Kurva ROC memplot *sensitivity* dengan *specificity* untuk setiap *threshold*. Biasanya nilai dari $1 - specificity$ diplot di sumbu horizontal sementara *sensitivity* diplot di sumbu vertikal. Nilai yang berada di dekat 0 di sumbu x mengindikasikan *high specificity* dan *low sensitivity* sesuai dengan orientasi pada kedua sumbu. Area di bawah kurva ROC dinamakan AUC. Nilai dari AUC dihitung untuk menentukan kebaikan model. Nilai AUC yang semakin dekat dengan 1 mengindikasikan bahwa model bagus untuk digunakan.

Setelah model didapat, dilakukan prediksi menggunakan model tersebut dan dibuat *confusion matrix* dengan nilai – nilai dari responsnya. Kita menggunakan *confusion matrix* sebagai cara lain untuk menguji kebagusan dari model. Nilai peluang $\hat{\pi}_i$ dihitung dan setiap kasus ke-i

diprediksi menjadi “terjadi” atau “tidak terjadi” tergantung apakah $\hat{\pi}_i$ lebih besar atau lebih kecil dari *threshold* yang dipilih.

Isi dari *Confusion Matrix* adalah:

	Event	No Event
Event	A	B
No Event	C	D

Rumus – rumus yang akan digunakan dalam analisis adalah sebagai berikut:

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

$$Sensitivity = \frac{A}{A + C}$$

$$Specificity = \frac{D}{B + D}$$

$$Prevalence = \frac{A + C}{A + B + C + D}$$

$$PPV = \frac{Sensitivity \times Prevalence}{(Sensitivity \times Prevalence) + \big((1 - Specificity) \times (1 - Prevalence)\big)}$$

$$NPV = \frac{Specificity \times (1 - Prevalence)}{\big((1 - Sensivity) \times Prevalence\big) + \big((Specificity) \times (1 - Prevalence)\big)}$$

$$Detection\ Rate = \frac{A}{A + B + C + D}$$

$$Detection\ Prevalence = \frac{A + B}{A + B + C + D}$$

$$Balanced\ Accuracy = \frac{Sensitivity + Specificity}{2}$$

- *Accuracy* menunjukkan tingkat kecocokan data prediksi dengan data asli
- *Sensitivity* adalah proporsi terjadinya prediksi “0” dan sebenarnya “0” (*True Positive*) dengan jumlah kategori “0”
- *Specificity* adalah proporsi terjadinya prediksi “1” dan sebenarnya “1” (*True Negative*) dengan jumlah kategori “1”
- *Prevalence* adalah proporsi jumlah kategori “0” pada data
- *Positive Predicted Value* adalah presentasi dari *True Positive* dari seluruh hasil prediksi yang positif.
- *Negative Predicted Value* adalah presentasi dari *True Negative* dari seluruh hasil prediksi yang negatif

Kita menginginkan nilai *Accuracy*, *Sensitivity*, dan *Specificity* yang setinggi mungkin dengan nilai *threshold* yang sesuai untuk mengecek kebaikan model. Setelah dirasa cukup tinggi, maka model tersebut digunakan untuk memprediksi beberapa data yang kemudian dibandingkan dengan data validasi yang di awal telah dipisahkan dari data yang sudah bersih. Jika model yang digunakan bagus, maka prediksi yang dihasilkan akan memiliki ketepatan yang tinggi.

BAB III

ANALISIS DATA

Pertama, aktifkan library yang akan digunakan dalam permodelan

```
library(readxl)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(pROC)
```

```
## Warning: package 'pROC' was built under R version 4.0.5
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 4.0.5
```

```
## Loading required package: lattice
```

```
## Loading required package: ggplot2
```

Kemudian, input data ke dalam R dan bersihkan semua nilai yang kosong dan nilai pencilan

```
data = read_excel("D:/Brian Stefano/Semester 6/GLM/CVD.xlsx")
```

Selanjutnya, kita kategorikan variabel - variabel kontinu.

```
for (i in 1:nrow(data)){
  if(data$sbp[i] < 120){
    data$sbp[i] = 0
  }else if(data$sbp[i] < 140){
    data$sbp[i] = 1
  }else if(data$sbp[i] < 160){
    data$sbp[i] = 2
```

```
 }else{
   data$sbp[i] = 3}
}

for (i in 1:nrow(data)){
 if(data$tobacco[i] == 0){
   data$tobacco[i] = 0
 }else if(data$tobacco[i] <= 5){
   data$tobacco[i] = 1
 }else if(data$tobacco[i] <= 10){
   data$tobacco[i] = 2
 }else{
   data$tobacco[i] = 3
 }
}

for (i in 1:nrow(data)){
 if(data$ldl[i] <= 2.6){
   data$ldl[i] = 0
 }else if(data$ldl[i] <= 3.3){
   data$ldl[i] = 1
 }else if(data$ldl[i] <= 4.1){
   data$ldl[i] = 2
 }else if(data$ldl[i] <= 4.9){
   data$ldl[i] = 3
 }else{
   data$ldl[i] = 4
 }
}

for (i in 1:nrow(data)){
 if(data$famhist[i] == "Present"){
   data$famhist[i] = 0
 }else{
   data$famhist[i] = 1
 }
}

for (i in 1:nrow(data)){
 if(data$typea[i] <= 47){
   data$typea[i] = 0
 }else if(data$typea[i] <= 53){
   data$typea[i] = 1
 }else if(data$typea[i] <= 60){
   data$typea[i] = 2
 }else{
```

```
  data$typea[i] = 3
 }
}

for (i in 1:nrow(data)){
 if(data$obesity[i] < 18.5){
  data$obesity[i] = 0
 }else if(data$obesity[i] <= 24.9){
  data$obesity[i] = 1
 }else if(data$obesity[i] <= 29.9){
  data$obesity[i] = 2
 }else if(data$obesity[i] <= 34.9){
  data$obesity[i] = 3
 }else if(data$obesity[i] <= 39.9){
  data$obesity[i] = 4
 }else{
  data$obesity[i] = 5
 }
}

for (i in 1:nrow(data)){
 if(data$age[i] <= 14){
  data$age[i] = 0
 }else if(data$age[i] <= 24){
  data$age[i] = 1
 }else if(data$age[i] <= 34){
  data$age[i] = 2
 }else if(data$age[i] <= 44){
  data$age[i] = 3
 }else if(data$age[i] <= 54){
  data$age[i] = 4
 }else if(data$age[i] <= 64){
  data$age[i] = 5
 }else{
  data$age[i] = 6
 }
}

for (i in 1:nrow(data)){
 if(data$alcohol[i] == 0){
  data$alcohol[i] = 0
 }else if(data$alcohol[i] <= 48){
  data$alcohol[i] = 1
 }else if(data$alcohol[i] <= 96){
  data$alcohol[i] = 2
 }else if(data$alcohol[i] <= 190){
```

```
  data$alcohol[i] = 3
 }else{
  data$alcohol[i] = 4
 }
}
```

Pastikan bahwa variabel kategorikal memiliki bentuk faktor

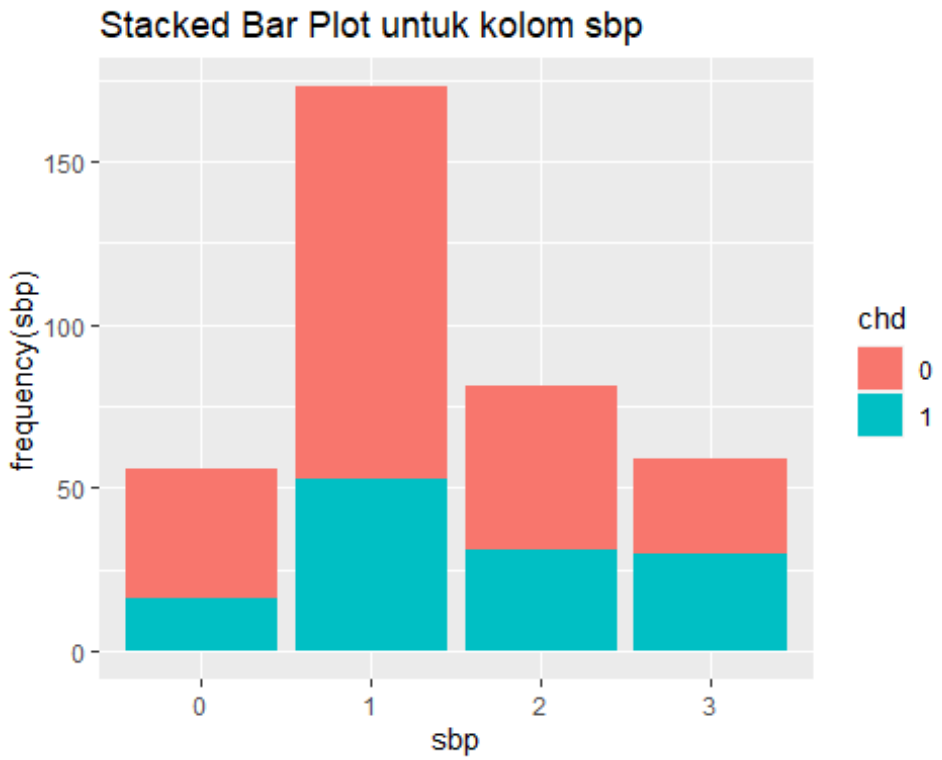
```
data$sbp = as.factor(data$sbp)
data$famhist = as.factor(data$famhist)
data$obesity = as.factor(data$obesity)
data$age = as.factor(data$age)
data$chd = as.factor(data$chd)
data$tobacco = as.factor(data$tobacco)
data$ldl = as.factor(data$ldl)
data$alcohol = as.factor(data$alcohol)
data$typea = as.factor(data$typea)
```

Kemudian, data diacak dan dibagi menjadi data train dan data validasi

```
set.seed(108)
data_shuffle = data[sample(nrow(data)),]
training_index = seq(1, nrow(data)*0.8, 1)
data_train = data_shuffle[training_index,]
validation = data_shuffle[-training_index,]
```

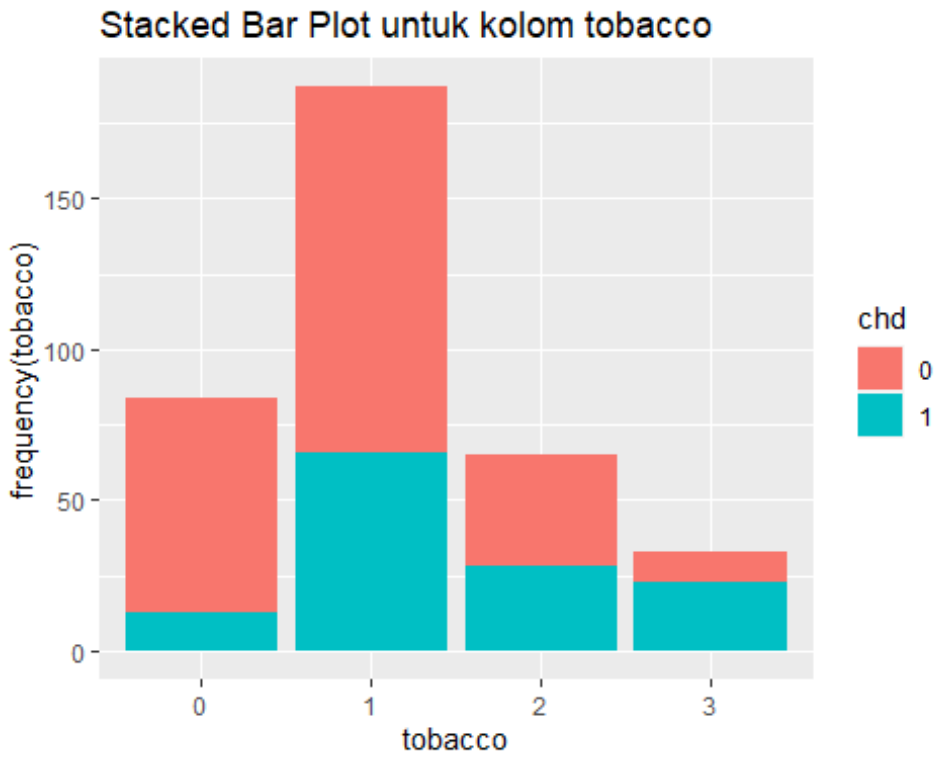
Setelahnya, dilakukan *Preliminary Analysis*

```
ggplot(data_train, aes(x = sbp, y = frequency(sbp), fill = chd)) + geom_bar(stat = "identity") + ggtitle("Stacked Bar Plot untuk kolom sbp")
```



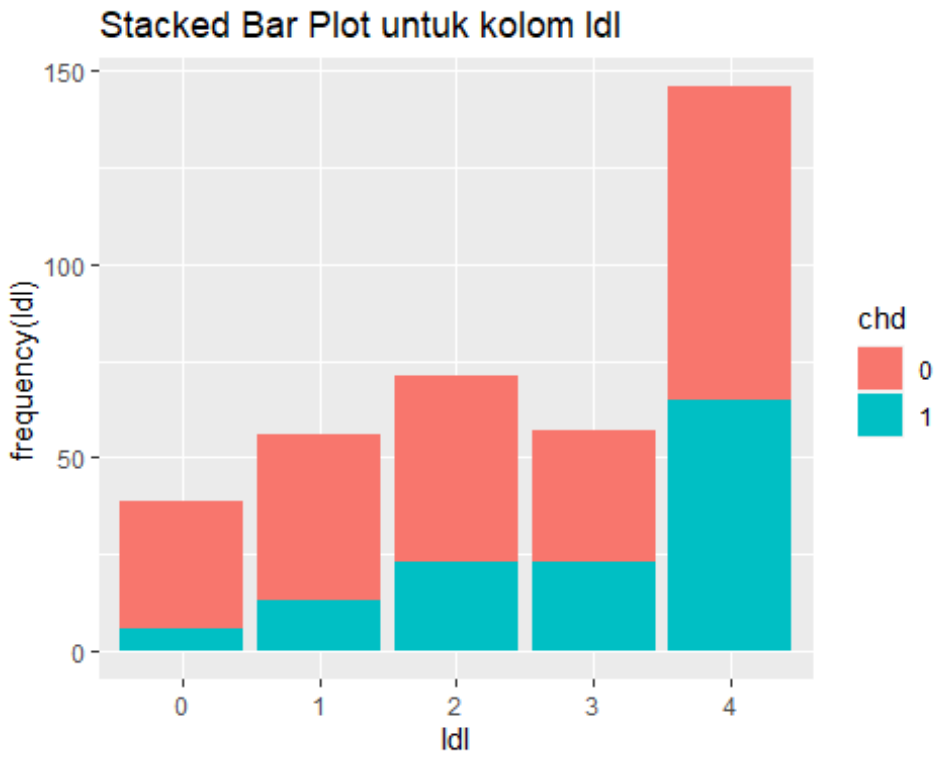
Pada stacked bar plot untuk kolom “sbp”, terlihat bahwa kategori 1 merupakan kategori dengan jumlah orang paling banyak, yaitu orang - orang yang berada pada tahap sebelum hipertensi. Akan tetapi, jika dilihat proporsi dari jumlah yang terkena CVD dengan yang tidak terkena CVD, kategori 3 yang terlihat paling tinggi.

```
ggplot(data_train, aes(x = tobacco, y = frequency(tobacco), fill = chd)) +
geom_bar(stat = "identity") + ggtitle("Stacked Bar Plot untuk kolom
tobacco")
```



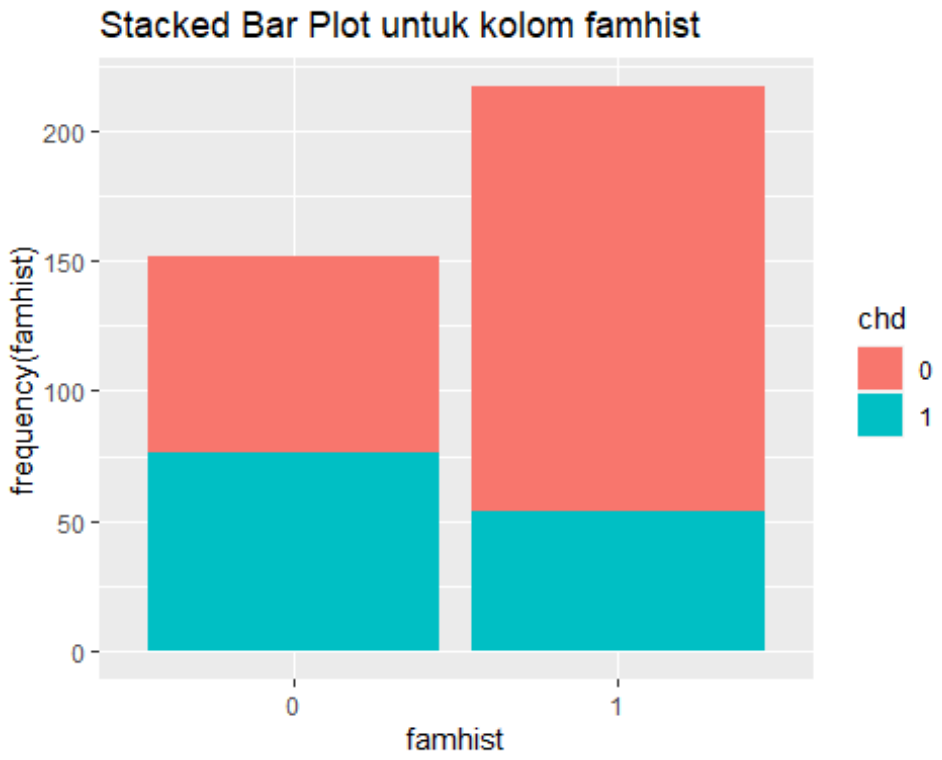
Pada stacked bar plot untuk kolom “tobacco”, sama dengan kolom “sbp”, kategori 1 merupakan kategori dengan jumlah orang terbanyak dan jumlah orang yang terkena CVD terbanyak, yaitu orang - orang yang mengonsumsi tembakau secara ringan. Pada kategori 3 terlihat bahwa proporsi orang yang terkena CVD dan yang tidak terkena CVD yang paling besar.

```
ggplot(data_train, aes(x = ldl, y = frequency(ldl), fill = chd)) + geom_bar(stat = "identity") + ggtitle("Stacked Bar Plot untuk kolom ldl")
```



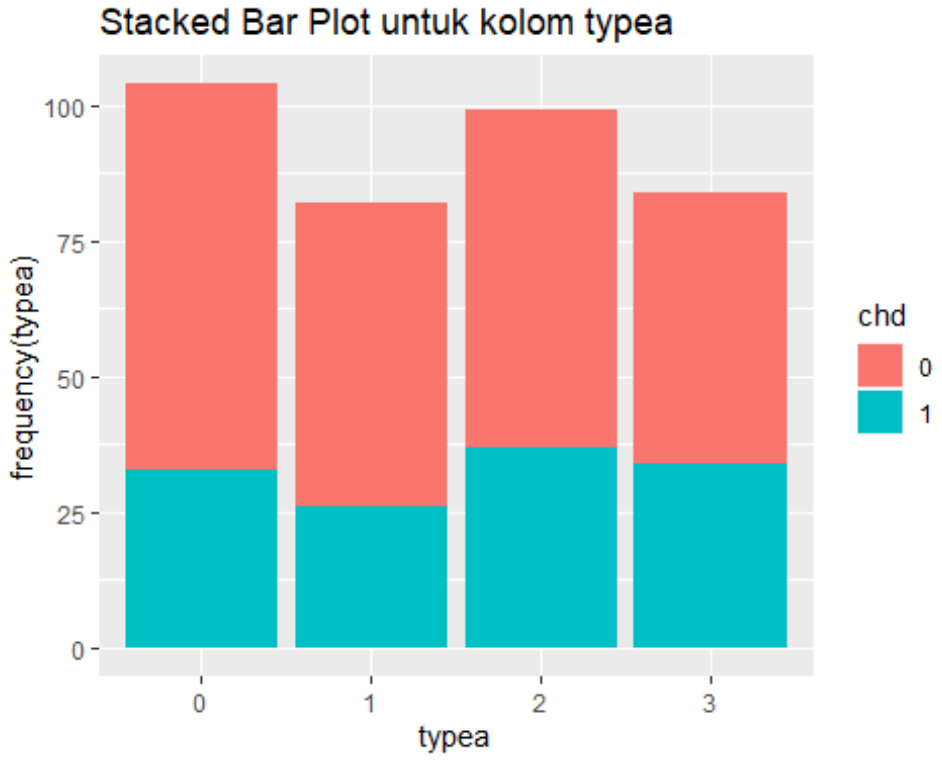
Pada Stacked Bar plot untuk kolom ldl, terlihat bahwa jumlah orang terbanyak dan orang yang mengalami CVD berada pada kategori ke 4, yaitu orang - orang yang memiliki kolesterol sangat tinggi. Adapun pada kategori lain, jumlah orang yang terkena CVD tidak sebanyak kategori 4.

```
ggplot(data_train, aes(x = famhist, y = frequency(famhist), fill = chd)) +
geom_bar(stat = "identity") + ggtitle("Stacked Bar Plot untuk kolom
famhist")
```



Pada Stacked Bar plot untuk kolom “famhist”, terlihat bahwa kategori 0 memiliki orang yang lebih banyak terkena CVD, yaitu kategori orang yang memiliki keluarga dengan riwayat penyakit CVD.

```
ggplot(data_train, aes(x = typea, y = frequency(typea), fill = chd)) +
geom_bar(stat = "identity") + ggtitle("Stacked Bar Plot untuk kolom
typea")
```



Pada Stack Bar plot untuk kolom “typea”, terlihat bahwa orang yang menderita CVD terbanyak jatuh pada kategori 3, yaitu orang yang memiliki kepribadian A, namun jika dibandingkan dengan kategori lain, jumlah orang yang mengalami CVD tidak banyak berbeda dari kategori 3. Pada tahap ini dapat diduga bahwa kolom “typea” tidak signifikan dari model.

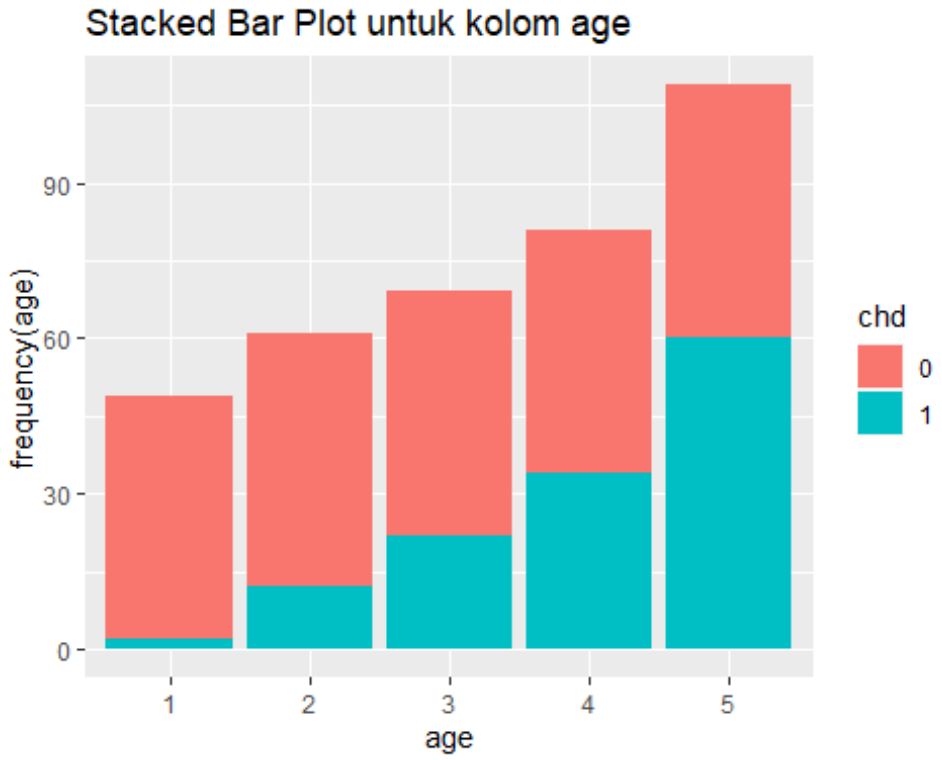
```
ggplot(data_train, aes(x = obesity, y = frequency(obesity), fill = chd)) +
geom_bar(stat = "identity") + ggtitle("Stacked Bar Plot untuk kolom
obesity")
```

Pada Stacked Bar plot untuk kolom “obesity”, jumlah orang terbanyak yang mengalami CVD adalah pada kategori 2, yaitu orang yang *Overweight*, diikuti dengan orang yang memiliki kategori 1 dan untuk kategori lainnya tidak terlalu banyak orang yang mengalami CVD.

```
ggplot(data_train, aes(x = alcohol, y = frequency(alcohol), fill = chd)) +
geom_bar(stat = "identity") + ggtitle("Stacked Bar Plot untuk kolom
alcohol")
```

Pada Stacked Bar plot untuk kolom “alcohol”, jumlah penderita CVD paling banyak berada pada kategori 1, yaitu kategori peminum ringan, sementara untuk kategori lain tidak terlalu banyak ditemukan orang yang mengalami CVD untuk kolom “alcohol” ini dikarenakan observasi yang jatuh pada kategori tersebut tidak banyak.

```
ggplot(data_train, aes(x = age, y = frequency(age), fill = chd)) + geom_bar(stat = "identity") + ggtitle("Stacked Bar Plot untuk kolom age")
```



Pada Stacked Bar plot untuk kolom “age”, terlihat bahwa semakin tua orang tersebut, semakin rentan orang tersebut menderita CVD.

Setelah dilakukan *Preliminary Analysis* untuk setiap kolom, akan dipilih base level untuk tiap - tiap variabel kategorikal. Pemilihan base level adalah kategori yang memiliki jumlah paling banyak.

```
summary(data_train)
```

```
##  sbp    tobacco ldl    famhist typea   obesity alcohol  age    chd
##  0: 56  0: 84   0: 39  0:152   0:104    0:  6     0: 92    1: 49  0:239
##  1:173   1:187  1: 56   1:217   1: 82   1:148     1:239    2: 61  1:130
##  2: 81   2: 65   2: 71           2: 99   2:158    2: 31     3: 69
##  3: 59   3: 33   3: 57           3: 84   3: 47    3:  7     4: 81
##                  4:146                   4:  8             5:109
##                                          5:  2
```

```
data_train = data_train %>% mutate(sbp = relevel(sbp, ref = "1"))
data_train = data_train %>% mutate(tobacco = relevel(tobacco, ref = "1"))
data_train = data_train %>% mutate(ldl = relevel(ldl, ref = "4"))
data_train = data_train %>% mutate(famhist = relevel(famhist, ref = "1"))
data_train = data_train %>% mutate(typea = relevel(typea, ref = "0"))
data_train = data_train %>% mutate(obesity = relevel(obesity, ref = "2"))
data_train = data_train %>% mutate(alcohol = relevel(alcohol, ref = "1"))
data_train = data_train %>% mutate(age = relevel(age, ref = "5"))
```

Selanjutnya, dimodelkan regresi logistik dan digunakan *Stepwise Regression*, *Forward Selection* dan *Backward Elimination* untuk mencari model yang memungkinkan.

```
model1 = step(glm(chd ~ sbp + tobacco + ldl + famhist + typea + obesity + alcohol + age,
          family = binomial(link = "logit"),
          data = data_train),
          direction = "both",
          trace = FALSE)
summary(model1)
```

```
##
## Call:
## glm(formula = chd ~ tobacco + famhist + typea + age, family = binomial(link = "logit"),
##     data = data_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1089  -0.8904  -0.4469   1.0080   2.9137
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.50273    0.33161  -1.516 0.129516
## tobacco0    -0.78392    0.37463  -2.093 0.036393 *
## tobacco2    -0.12571    0.32815  -0.383 0.701661
```

```
## tobacco3     0.93482    0.44800   2.087 0.036920 *
## famhist0     0.94031    0.25015   3.759 0.000171 ***
## typea1      -0.02503    0.35450  -0.071 0.943718
## typea2       0.46668    0.34491   1.353 0.176041
## typea3       0.73679    0.35661   2.066 0.038818 *
## age1        -2.91869    0.77902  -3.747 0.000179 ***
## age2        -1.43384    0.41584  -3.448 0.000565 ***
## age3        -0.94509    0.35755  -2.643 0.008212 **
## age4        -0.55892    0.31968  -1.748 0.080402 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 478.86  on 368  degrees of freedom
## Residual deviance: 390.93  on 357  degrees of freedom
## AIC: 414.93
##
## Number of Fisher Scoring iterations: 5
```

```
model2 = step(glm(chd ~ sbp + tobacco + ldl + famhist + typea + obesity + alcohol + age,
          family = binomial(link = "logit"),
          data = data_train),
          direction = "forward",
          trace = FALSE)
summary(model2)
```

```
##
## Call:
## glm(formula = chd ~ sbp + tobacco + ldl + famhist + typea + obesity +
##     alcohol + age, family = binomial(link = "logit"), data = data_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.9671  -0.8344  -0.4488   0.9556   2.7168
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.69820    0.45396  -1.538 0.124042
## sbp0         0.29664    0.40409   0.734 0.462888
## sbp2         0.21280    0.33551   0.634 0.525916
## sbp3         0.49080    0.36777   1.335 0.182032
## tobacco0    -0.73789    0.39399  -1.873 0.061092 .
## tobacco2    -0.18324    0.34497  -0.531 0.595298
## tobacco3     0.85529    0.45971   1.860 0.062817 .
## ldl0        -0.91313    0.55026  -1.659 0.097025 .
## ldl1        -0.28309    0.43323  -0.653 0.513473
## ldl2        -0.02733    0.36140  -0.076 0.939717
```

```
## ldl3       -0.00733    0.36625  -0.020 0.984033
## famhist0    1.01877    0.26325   3.870 0.000109 ***
## typea1     -0.01706    0.36740  -0.046 0.962962
## typea2      0.57310    0.36092   1.588 0.112312
## typea3      0.85270    0.38035   2.242 0.024970 *
## obesity0    1.45833    1.12577   1.295 0.195181
## obesity1    0.10321    0.29655   0.348 0.727824
## obesity3   -0.43531    0.39692  -1.097 0.272765
## obesity4   -0.32559    0.83793  -0.389 0.697601
## obesity5    2.53930    1.81046   1.403 0.160746
## alcohol0    0.07834    0.31660   0.247 0.804567
## alcohol2    0.16720    0.46114   0.363 0.716926
## alcohol3    0.07870    0.88154   0.089 0.928858
## age1       -3.05404    0.84585  -3.611 0.000305 ***
## age2       -1.34948    0.44007  -3.067 0.002166 **
## age3       -0.93719    0.38303  -2.447 0.014415 *
## age4       -0.56776    0.33822  -1.679 0.093218 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 478.86  on 368  degrees of freedom
## Residual deviance: 381.40  on 342  degrees of freedom
## AIC: 435.4
##
## Number of Fisher Scoring iterations: 6
```

```r
model3 = step(glm(chd ~ sbp + tobacco + ldl + famhist + typea + obesity + alcohol + age,
          family = binomial(link = "logit"),
          data = data_train),
          direction = "backward",
          trace = FALSE)
summary(model3)
```

```
##
## Call:
## glm(formula = chd ~ tobacco + famhist + typea + age, family = binomial(link = "logit"),
##     data = data_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1089  -0.8904  -0.4469   1.0080   2.9137
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.50273    0.33161  -1.516 0.129516
## tobacco0    -0.78392    0.37463  -2.093 0.036393 *
```

```
## tobacco2    -0.12571    0.32815  -0.383 0.701661
## tobacco3     0.93482    0.44800   2.087 0.036920 *
## famhist0     0.94031    0.25015   3.759 0.000171 ***
## typea1      -0.02503    0.35450  -0.071 0.943718
## typea2       0.46668    0.34491   1.353 0.176041
## typea3       0.73679    0.35661   2.066 0.038818 *
## age1        -2.91869    0.77902  -3.747 0.000179 ***
## age2        -1.43384    0.41584  -3.448 0.000565 ***
## age3        -0.94509    0.35755  -2.643 0.008212 **
## age4        -0.55892    0.31968  -1.748 0.080402 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 478.86  on 368  degrees of freedom
## Residual deviance: 390.93  on 357  degrees of freedom
## AIC: 414.93
##
## Number of Fisher Scoring iterations: 5
```

Dicoba juga membuat model dengan menghilangkan variabel “typea” sesuai dengan dugaan tadi.

```
model4 = step(glm(chd ~ sbp + tobacco + ldl + famhist + obesity + alcohol + age,
          family = binomial(link = "logit"),
          data = data_train),
          direction = "both",
          trace = FALSE)
summary(model4)
```

```
##
## Call:
## glm(formula = chd ~ tobacco + famhist + age, family = binomial(link = "logit"),
##     data = data_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8998  -0.8675  -0.4869   0.9722   2.7682
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.3016     0.2742  -1.100 0.271339
## tobacco0     -0.7963     0.3719  -2.141 0.032257 *
## tobacco2     -0.1349     0.3232  -0.418 0.676250
## tobacco3      0.9859     0.4413   2.234 0.025487 *
## famhist0      0.9406     0.2473   3.804 0.000143 ***
## age1         -2.7115     0.7698  -3.523 0.000427 ***
## age2         -1.2615     0.4044  -3.120 0.001811 **
```

```
## age3        -0.7951     0.3425  -2.322 0.020245 *
## age4        -0.4817     0.3126  -1.541 0.123306
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 478.86  on 368  degrees of freedom
## Residual deviance: 397.22  on 360  degrees of freedom
## AIC: 415.22
##
## Number of Fisher Scoring iterations: 5
```

```
model5 = step(glm(chd ~ sbp + tobacco + ldl + famhist + obesity + alcohol + age,
          family = binomial(link = "logit"),
          data = data_train),
          direction = "forward",
          trace = FALSE)
summary(model5)
```

```
##
## Call:
## glm(formula = chd ~ sbp + tobacco + ldl + famhist + obesity +
##     alcohol + age, family = binomial(link = "logit"), data = data_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7731  -0.8566  -0.4602   0.9521   2.7625
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.41220    0.39574  -1.042 0.297608
## sbp0         0.25296    0.39658   0.638 0.523578
## sbp2         0.17916    0.33079   0.542 0.588085
## sbp3         0.34565    0.35750   0.967 0.333612
## tobacco0    -0.75715    0.38735  -1.955 0.050622 .
## tobacco2    -0.19426    0.33979  -0.572 0.567516
## tobacco3     0.92405    0.45353   2.037 0.041606 *
## ldl0        -0.92144    0.54788  -1.682 0.092606 .
## ldl1        -0.37312    0.42349  -0.881 0.378287
## ldl2        -0.07136    0.35621  -0.200 0.841229
## ldl3        -0.03831    0.35883  -0.107 0.914976
## famhist0     0.98337    0.25870   3.801 0.000144 ***
## obesity0     1.45936    1.06198   1.374 0.169383
## obesity1     0.09931    0.29382   0.338 0.735371
## obesity3    -0.26385    0.38560  -0.684 0.493808
## obesity4    -0.20352    0.82343  -0.247 0.804785
## obesity5     2.44017    1.94025   1.258 0.208517
```

```
## alcohol0    0.11201    0.31140   0.360 0.719070
## alcohol2    0.12287    0.45290   0.271 0.786160
## alcohol3    0.46636    0.88156   0.529 0.596797
## age1       -2.81209    0.84480  -3.329 0.000873 ***
## age2       -1.15394    0.42787  -2.697 0.006997 **
## age3       -0.76169    0.36815  -2.069 0.038551 *
## age4       -0.46344    0.33085  -1.401 0.161292
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 478.86  on 368  degrees of freedom
## Residual deviance: 389.02  on 345  degrees of freedom
## AIC: 437.02
##
## Number of Fisher Scoring iterations: 6
```

```
model6 =  step(glm(chd ~  sbp +  tobacco +  ldl +  famhist +  obesity +  alcohol +  age,
          family = binomial(link = "logit"),
          data = data_train),
          direction = "backward",
          trace = FALSE)
summary(model6)
```

```
##
## Call:
## glm(formula = chd ~ tobacco + famhist + age, family = binomial(link = "logit"),
##     data = data_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8998  -0.8675  -0.4869   0.9722   2.7682
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.3016     0.2742  -1.100 0.271339
## tobacco0     -0.7963     0.3719  -2.141 0.032257 *
## tobacco2     -0.1349     0.3232  -0.418 0.676250
## tobacco3      0.9859     0.4413   2.234 0.025487 *
## famhist0      0.9406     0.2473   3.804 0.000143 ***
## age1         -2.7115     0.7698  -3.523 0.000427 ***
## age2         -1.2615     0.4044  -3.120 0.001811 **
## age3         -0.7951     0.3425  -2.322 0.020245 *
## age4         -0.4817     0.3126  -1.541 0.123306
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 478.86  on 368  degrees of freedom
## Residual deviance: 397.22  on 360  degrees of freedom
## AIC: 415.22
##
## Number of Fisher Scoring iterations: 5
```

Dapat dilihat bahwa terdapat 4 model yang didapat dari hasil pembuatan model (Model dari *Stepwise Regression* dan model dari *Backward Elimination* sama). Setelah didapat model yang memungkinkan, dibuat plot ROC dan dihitung nilai AUC untuk mengukur kebaikan dari masing - masing model.

```
pred1 = predict(model1, type = "response")
rcurve1 = roc(data_train$chd ~ pred1)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
pred2 = predict(model2, type = "response")
rcurve2 = roc(data_train$chd ~ pred2)
```

```
## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
```

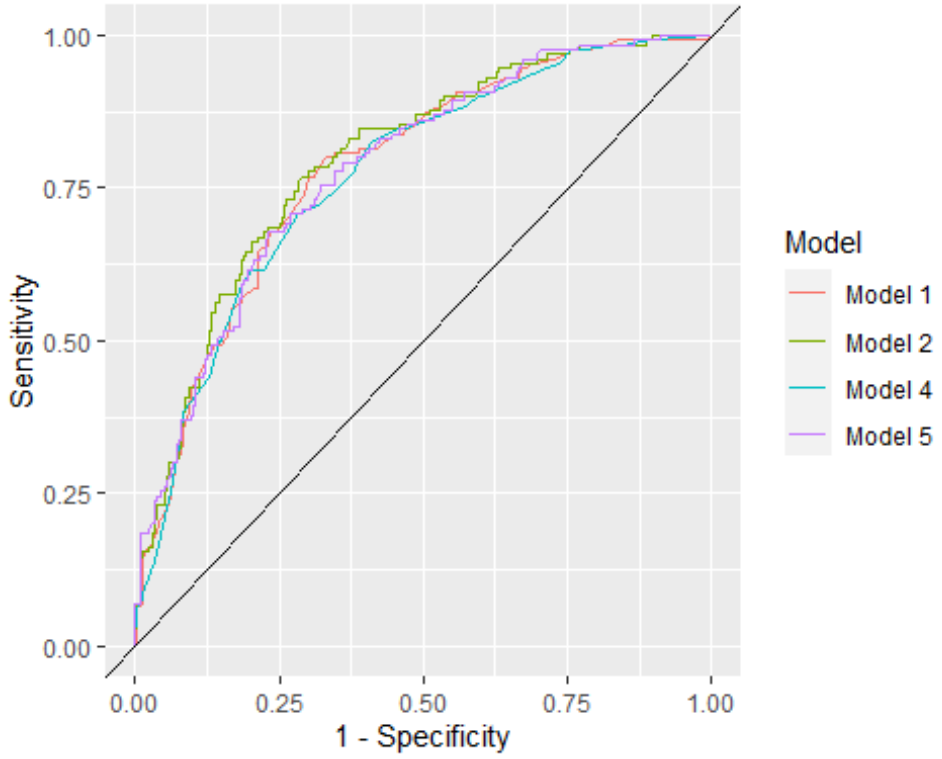
```
pred4 = predict(model4, type = "response")
rcurve4 = roc(data_train$chd ~ pred4)
```

```
## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
```

```
pred5 = predict(model5, type = "response")
rcurve5 = roc(data_train$chd ~ pred5)
```

```
## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
```

```
roclist = list("Model 1" = rcurve1, "Model 2" = rcurve2, "Model 4" = rcurve4, "Model
5" = rcurve5)
ggroc(roclist, aes = "colour", legacy.axes = TRUE) +
  geom_abline(intercept = 0, slope = 1) +
  labs(x = "1 - Specificity",
   y = "Sensitivity",
   colour = "Model")
```




```
auc_value =  c(auc(rcurve1), auc(rcurve2), auc(rcurve4), auc(rcurve5))
auc_value
```

```
## [1] 0.7827486 0.7956389 0.7716286 0.7829417
```

Pada grafik terlihat bahwa model 2 merupakan model yang paling cepat menuju 1 dan juga model 2 memiliki nilai auc terbesar, namun perhatikan juga bahwa nilai dari auc dan grafik masing - masing model tidak jauh berbeda satu sama lain.

Kemudian, dicari nilai threshold yang memaksimalkan nilai akurasi, sensitivitas dan spesifisitas.

```
a =  coords(rcurve1, "best", ret= "threshold")
b =  coords(rcurve2, "best", ret= "threshold")
c =  coords(rcurve4, "best", ret= "threshold")
d =  coords(rcurve5, "best", ret= "threshold")
c(a, b, c, d)
```

```
## $threshold
## [1] 0.3473233
##
## $threshold
## [1] 0.3452894
##
## $threshold
## [1] 0.408856
##
## $threshold
## [1] 0.4235245
```

Gunakan nilai threshold terbaik untuk mencari prediksi dari masing - masing model

```
predict1 = factor(ifelse(model1$fitted.values < 0.3473233, 0, 1))
mat1 = confusionMatrix(predict1, data_train$chd)
mat1
```

```
## Confusion Matrix and Statistics
##
##          Reference
## Prediction   0   1
##         0 160  26
##         1  79 104
##
##               Accuracy : 0.7154
##                 95% CI : (0.6665, 0.7609)
##    No Information Rate : 0.6477
##    P-Value [Acc > NIR] : 0.003381
##
##                  Kappa : 0.4295
##
##  Mcnemar's Test P-Value : 3.881e-07
##
##             Sensitivity : 0.6695
##             Specificity : 0.8000
##          Pos Pred Value : 0.8602
##          Neg Pred Value : 0.5683
##              Prevalence : 0.6477
##          Detection Rate : 0.4336
##    Detection Prevalence : 0.5041
##       Balanced Accuracy : 0.7347
##
##        'Positive' Class : 0
##
```

```
prob1 = predict(model1, validation, type = "response")
predtest1 = factor(ifelse(prob1 < 0.3473233, 0, 1))
data_valid1 = data.frame(FittedValue = predtest1, validation)

count1 = 0
for (i in 1:nrow(data_valid1)){
 if(data_valid1$FittedValue[i] == data_valid1$chd[i]){
  count1 = count1 + 1
 }
}
count1
```

```
## [1] 64
```

```r
predict2 = factor(ifelse(model2$fitted.values < 0.3452894, 0, 1))
mat2 = confusionMatrix(predict2, data_train$chd)
mat2
```

```
## Confusion Matrix and Statistics
##
##          Reference
## Prediction   0   1
##         0 170  30
##         1  69 100
##
##               Accuracy : 0.7317
##                 95% CI : (0.6834, 0.7763)
##    No Information Rate : 0.6477
##    P-Value [Acc > NIR] : 0.0003506
##
##                  Kappa : 0.4498
##
##  Mcnemar's Test P-Value : 0.0001339
##
##             Sensitivity : 0.7113
##             Specificity : 0.7692
##          Pos Pred Value : 0.8500
##          Neg Pred Value : 0.5917
##              Prevalence : 0.6477
##          Detection Rate : 0.4607
##    Detection Prevalence : 0.5420
##       Balanced Accuracy : 0.7403
##
##       'Positive' Class : 0
##
```

```r
prob2 = predict(model2, validation, type = "response")
predtest2 = factor(ifelse(prob2 < 0.3452894, 0, 1))
data_valid2 = data.frame(FittedValue = predtest2, validation)

count2 = 0
for (i in 1:nrow(data_valid2)){
 if(data_valid2$FittedValue[i] == data_valid2$chd[i]){
  count2 = count2 + 1
 }
}
count2
```

```
## [1] 67
```

```r
predict4 = factor(ifelse(model4$fitted.values < 0.408856, 0, 1))
mat4 = confusionMatrix(predict4, data_train$chd)
mat4
```

```
## Confusion Matrix and Statistics
##
##          Reference
## Prediction   0   1
##         0 171  38
##         1  68  92
##
##               Accuracy : 0.7127
##                 95% CI : (0.6636, 0.7584)
##    No Information Rate : 0.6477
##    P-Value [Acc > NIR] : 0.004722
##
##                  Kappa : 0.402
##
##  Mcnemar's Test P-Value : 0.004852
##
##             Sensitivity : 0.7155
##             Specificity : 0.7077
##          Pos Pred Value : 0.8182
##          Neg Pred Value : 0.5750
##              Prevalence : 0.6477
##          Detection Rate : 0.4634
##    Detection Prevalence : 0.5664
##       Balanced Accuracy : 0.7116
##
##        'Positive' Class : 0
##
```

```
prob4 = predict(model4, validation, type = "response")
predtest4 = factor(ifelse(prob4 < 0.408856, 0, 1))
data_valid4 = data.frame(FittedValue = predtest4, validation)

count4 = 0
for (i in 1:nrow(data_valid4)){
 if(data_valid4$FittedValue[i] == data_valid4$chd[i]){
  count4 = count4 + 1
 }
}
count4
```

```
## [1] 62
```

```
predict5 = factor(ifelse(model5$fitted.values < 0.408856, 0, 1))
mat5 = confusionMatrix(predict5, data_train$chd)
mat5
```

```
## Confusion Matrix and Statistics
##
##          Reference
```

```
## Prediction   0   1
##         0 178  42
##         1  61  88
##
##               Accuracy : 0.7209
##                 95% CI : (0.6721, 0.7661)
##    No Information Rate : 0.6477
##    P-Value [Acc > NIR] : 0.001671
##
##                  Kappa : 0.4081
##
##  Mcnemar's Test P-Value : 0.076131
##
##             Sensitivity : 0.7448
##             Specificity : 0.6769
##          Pos Pred Value : 0.8091
##          Neg Pred Value : 0.5906
##              Prevalence : 0.6477
##          Detection Rate : 0.4824
##    Detection Prevalence : 0.5962
##       Balanced Accuracy : 0.7108
##
##       'Positive' Class : 0
##
```

```
prob5 = predict(model5, validation, type = "response")
predtest5 = factor(ifelse(prob5 < 0.408856, 0, 1))
data_valid5 = data.frame(FittedValue = predtest5, validation)

count5 = 0
for (i in 1:nrow(data_valid5)){
  if(data_valid5$FittedValue[i] ==  data_valid5$chd[i]){
    count5 = count5 + 1
  }
}
count5
```

```
## [1] 69
```

Perhatikanlah bahwa nilai dari P-value[Acc >NIR] < 0.05 (alpha yang diambil). Karena itu, memakai model yang telah dibuat lebih baik daripada memprediksi dengan proporsi terbanyak tanpa menggunakan model.

Jika dibandingkan nilai AUC, *Accuracy*, *Sensitivity*, *Specificity*, dan *count* (jumlah prediksi yang benar), maka hasilnya dapat dilihat pada tabel di bawah ini.

	AUC	*Accuracy*	*Sensitivity*	*Specificity*	*Count*	%
Model1	0.7827486	0.7154	0.6695	0.8000	64	68.81%
Model2	0.7956389	0.7317	0.7113	0.7692	67	72.04%

Model4	0.7716286	0.7127	0.7155	0.7077	62	66.66%
Model5	0.7829417	0.7209	0.7448	0.6769	69	74.19%

Dari hasil di atas, penulis memilih Model2 sebagai model yang terbaik untuk memodelkan data CVD dikarenakan beberapa hal berikut:

- Memiliki nilai AUC dan *Accuracy* yang paling tinggi
- Walaupun tidak memiliki nilai *Sensitivity* dan *Specificity* yang tertinggi, namun nilai tersebut pada Model2 merupakan nilai yang paling seimbang
- *Count* bermakna seberapa banyak prediksi yang benar saat validasi model. Hasil yang tidak berbeda jauh dengan nilai yang tertinggi mengindikasikan bahwa Model2 cukup baik dalam memprediksi hasil.

Dengan demikian, model akhir yang dipilih adalah:

$$\ln\left(\frac{\pi}{1-\pi}\right) = -0.6982 + 0.29664x_1 + 0.2128x_3 + 0.4908x_4 - 0.73789x_5 - 0.18324x_7 + 0.85529x_8 - 0.91313x_9 - 0.28309x_{10} - 0.02733x_{11} - 0.00733x_{12} + 1.01877x_{14} - 0.01706x_{17} + 0.57310x_{18} + 0.8527x_{19} + 1.45833x_{20} + 0.10321x_{21} - 0.43531x_{23} - 0.32559x_{24} + 2.5393x_{25} + 0.07834x_{26} + 0.1672x_{28} + 0.0787x_{29} - 3.05404x_{30} - 1.34948x_{31} - 0.93719x_{32} - 0.56776x_{34}$$

x_1, x_3, x_4 = sbp → x_2 = Base level

x_5, x_7, x_8 = tobacco → x_6 = Base level

$x_9, x_{10}, x_{11}, x_{12}$ = ldl → x_{13} = Base level

x_{14} = famhist → x_{15} = Base level

x_{17}, x_{18}, x_{19} = typea → x_{16} = Base level

$x_{20}, x_{21}, x_{23}, x_{24}, x_{25}$ = obesity → x_{22} = Base level

x_{26}, x_{28}, x_{29} = alcohol → x_{27} = Base level

$x_{30}, x_{31}, x_{32}, x_{34}$ = age → x_{33} = Base level

BAB IV

KESIMPULAN

Setelah dilakukan analisis, dapat ditentukan bahwa Model2 adalah model yang terbaik dalam memodelkan data. Variabel – variabel yang berpengaruh pada respons adalah “sbp”, “tobacco”, “ldl”, “famhist”, “typea”, “obesity”, “alcohol”, dan “age” dengan hanya “famhist”, “typea” kategori 3, dan “age” yang berpengaruh secara signifikan. Bentuk model tersebut adalah:

$$\begin{aligned}\ln\left(\frac{\pi}{1-\pi}\right) = {} & -0.6982 + 0.29664x_1 + 0.2128x_3 + 0.4908x_4 - 0.73789x_5 - 0.18324x_7 \\ & + 0.85529x_8 - 0.91313x_9 - 0.28309x_{10} - 0.02733x_{11} - 0.00733x_{12} \\ & + 1.01877x_{14} - 0.01706x_{17} + 0.57310x_{18} + 0.8527x_{19} + 1.45833x_{20} \\ & + 0.10321x_{21} - 0.43531x_{23} - 0.32559x_{24} + 2.5393x_{25} + 0.07834x_{26} \\ & + 0.1672x_{28} + 0.0787x_{29} - 3.05404x_{30} - 1.34948x_{31} - 0.93719x_{32} \\ & - 0.56776x_{34}\end{aligned}$$

x_1, x_3, x_4 = sbp → x_2 = Base level

x_5, x_7, x_8 = tobacco → x_6 = Base level

$x_9, x_{10}, x_{11}, x_{12}$ = ldl → x_{13} = Base level

x_{14} = famhist → x_{15} = Base level

x_{17}, x_{18}, x_{19} = typea → x_{16} = Base level

$x_{20}, x_{21}, x_{23}, x_{24}, x_{25}$ = obesity → x_{22} = Base level

x_{26}, x_{28}, x_{29} = alcohol → x_{27} = Base level

$x_{30}, x_{31}, x_{32}, x_{34}$ = age → x_{33} = Base level

Kemudian, dengan memvalidasi data dengan data validasi yang telah dipisah saat awal analisis, Model2 berhasil memprediksi sebanyak 67 dari 93 total data (72.04%).

REFERENSI

https://www.kaggle.com/yassinehamdaoui1/cardiovascular-disease

https://www.google.com/imgres?imgurl=https%3A%2F%2Fwww.researchgate.net%2Fprofile%2FReza-Malekzadeh-3%2Fpublication%2F242075175%2Ffigure%2Ftbl1%2FAS%3A669083280371734%4015365332935 09%2FClassification-of-obesity-in-adults.png&imgrefurl=https%3A%2F%2Fwww.researchgate.net%2Ffigure%2FClassification-of-obesity-in-adults_tbl1_242075175&tbnid=0z0hA35dYdDssM&vet=12ahUKEwiMsJ7t4cnwAhXglUsFHS2LBgcQMygBegUIARDFAQ..i&docid=utPV_5W4A974SM&w=610&h=311&q=obesity%20classification&ved=2ahUKEwiMsJ7t4cnwAhXglUsFHS2LBgcQMygBegUIARDFAQ

https://www.google.com/url?sa=i&url=https%3A%2F%2Fbcmj.org%2Farticles%2Fgeriatric-drinkers-evaluation-and-treatment-alcohol-overuse&psig=AOvVaw2upm4yGiXjIYab5rxJKo6U&ust=1621051115829000&source=images&cd=vfe&ved=0CAIQjRxqFwoTCIifm6SkyPACFQAAAAAdAAAAABAD

https://www.google.com/imgres?imgurl=https%3A%2F%2Fwww.yourbariatricsurgeryguide.com%2Fwp-content%2Fuploads%2F2018%2F05%2Fcholesterol-ldl1-410x158.gif&imgrefurl=https%3A%2F%2Fwww.yourbariatricsurgeryguide.com%2Fcholesterol%2F&tbnid=cGeybZ1ho-v9gM&vet=10CI8BEDMorAFqFwoTCLDG3si9yfACFQAAAAAdAAAAABAC..i&docid=w_Z0o77i09fbYM&w=410&h=158&q=low%20density%20lipoprotein%20cholesterol%20levels&hl=en&ved=0CI8BEDMorAFqFwoTCLDG3si9yfACFQAAAAAdAAAAABAC

https://www.simplypsychology.org/personality-a.html