**Praktikum 7 – REGRESI LOGISTIK**

Berbeda dengan regresi lininer yang digunakan untuk memprediksi variabel Y yang bersifat kontinu, regresi logistic digunakan untuk memprediksi variabel Y yang bersifat kategorik. Kasus regresi logistic dengan Y yang terdiri dari hanya du akelas dinamakan *binary classification problems (****binomial logistic regression)****.* Prediktor dapat bersifat kontinu, kategorik maupun gabungan keduanya.

# Tugas 1. Titanic Dataset

Gunakan data berikut ini untuk membangun model prediktif. Berikan interpretasi atas setiap output yang dihasilkan. Mulailah analisis dengan membuat tabulasi silang setiap predictor yang bersifat kategorik dengan respon (Y).

### Data Dictionary

|  |  |  |
| --- | --- | --- |
| **Variable** | **Definition** | **Key** |
| survival | Survival | 0 = No, 1 = Yes |
| pclass | Ticket class | 1 = 1st, 2 = 2nd, 3 = 3rd |
| sex | Sex |  |
| Age | Age in years |  |
| sibsp | # of siblings / spouses aboard the Titanic |  |
| parch | # of parents / children aboard the Titanic |  |
| ticket | Ticket number |  |
| fare | Passenger fare |  |
| cabin | Cabin number |  |
| embarked | Port of Embarkation | C = Cherbourg, Q = Queenstown, S = Southampton |

**REGRESI LOGISTIK**

|  |
| --- |
| > data\_titanic\_steven=read.delim("clipboard")  > str(data\_titanic\_steven)  'data.frame': 714 obs. of 8 variables:  $ PassengerId: int 1 2 3 4 5 7 8 9 10 11 ...  $ Survived : int 0 1 1 1 0 0 0 1 1 1 ...  $ Pclass : int 3 1 3 1 3 1 3 3 2 3 ...  $ Sex : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 1 1 1 ...  $ Age : num 22 38 26 35 35 54 2 27 14 4 ...  $ SibSp : int 1 1 0 1 0 0 3 0 1 1 ...  $ Parch : int 0 0 0 0 0 0 1 2 0 1 ...  $ Fare : num 7.25 71.28 7.92 53.1 8.05 ...  > sampel1<-sample(1:nrow(data\_titanic\_steven),0.75\*nrow(data\_titanic\_steven))  > training\_logistik\_stev<-data.frame(data\_titanic\_steven)[sampel1,]  > testing\_logistik\_stev<-data.frame(data\_titanic\_steven)[-sampel1,]  > model\_logistik\_stev=glm(Survived~.,data=training\_logistik\_stev,family = binomial)  > summary(model\_logistik\_stev)  Call:  glm(formula = Survived ~ ., family = binomial, data = training\_logistik\_stev)  Deviance Residuals:  Min 1Q Median 3Q Max  -2.4867 -0.5714 -0.3473 0.5208 2.6420  Coefficients:  Estimate Std. Error z value Pr(>|z|)  (Intercept) 5.4141075 0.7536605 7.184 6.78e-13 \*\*\*  PassengerId 0.0008641 0.0004678 1.847 0.0647 .  Pclass -1.2573168 0.1958495 -6.420 1.36e-10 \*\*\*  Sexmale -3.1712667 0.2824389 -11.228 < 2e-16 \*\*\*  Age -0.0498725 0.0099898 -4.992 5.97e-07 \*\*\*  SibSp -0.3522237 0.1570812 -2.242 0.0249 \*  Parch -0.1720649 0.1445036 -1.191 0.2338  Fare 0.0026433 0.0027225 0.971 0.3316  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  (Dispersion parameter for binomial family taken to be 1)  Null deviance: 710.25 on 534 degrees of freedom  Residual deviance: 432.28 on 527 degrees of freedom  AIC: 448.28  Number of Fisher Scoring iterations: 5 |

**#MELAKUKAN PREDIKSI**

|  |
| --- |
| > library("caret")  > prediksilogistik=predict(model\_logistik\_stev,testing\_logistik\_stev)  > pred\_logreg<-as.numeric(prediksilogistik>.5)  > tabel\_logreg<-table(pred\_logreg,testing\_logistik\_stev$Survived)  > tabel\_logreg    pred\_logreg 0 1  0 77 33  1 15 54  > confusionMatrix(pred\_logreg,testing\_logistik\_stev$Survived,positive = "1")  Error: `data` and `reference` should be factors with the same levels. |

# Tugas 2. The iris data set (species virginica and versicolor only)

Gunakan data berikut ini untuk membangun model prediktif. Berikan interpretasi atas setiap output yang dihasilkan. Mulailah analisis dengan membuat tabulasi silang setiap predictor yang bersifat kategorik dengan respon (Y).

# make a reduced iris data set that only contains virginica and versicolor species

> library(dplyr)

Attaching package: ‘dplyr’

The following objects are masked from ‘package:arules’:

intersect, recode, setdiff, setequal, union

The following objects are masked from ‘package:stats’:

filter, lag

The following objects are masked from ‘package:base’:

intersect, setdiff, setequal, union

Warning message:

package ‘dplyr’ was built under R version 3.4.4

> iris.small <- filter(iris, Species %in% c("virginica", "versicolor"))

Warning message:

package ‘bindrcpp’ was built under R version 3.4.4

# logistic regression

> glm.out <- glm(Species ~ Sepal.Width + Sepal.Length + Petal.Width + Petal.Length,

+ data = iris.small,

+ family = binomial)

> summary(glm.out)

Call:

glm(formula = Species ~ Sepal.Width + Sepal.Length + Petal.Width +

Petal.Length, family = binomial, data = iris.small)

Deviance Residuals:

Min 1Q Median 3Q Max

-2.01105 -0.00541 -0.00001 0.00677 1.78065

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) -42.638 25.707 -1.659 0.0972 .

Sepal.Width -6.681 4.480 -1.491 0.1359

Sepal.Length -2.465 2.394 -1.030 0.3032

Petal.Width 18.286 9.743 1.877 0.0605 .

Petal.Length 9.429 4.737 1.991 0.0465 \*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 138.629 on 99 degrees of freedom

Residual deviance: 11.899 on 95 degrees of freedom

AIC: 21.899

Number of Fisher Scoring iterations: 10

> exp(coef(glm.out)) # to get odds ratio

(Intercept) Sepal.Width Sepal.Length Petal.Width Petal.Length

3.038345e-19 1.254665e-03 8.499013e-02 8.741145e+07 1.244887e+04

# *Remove Sepal.Length as it has the highest P value:*

> glm.out <- glm(Species ~ Sepal.Width + Petal.Width + Petal.Length,

+ data = iris.small,

+ family = binomial)

Warning message:

glm.fit: fitted probabilities numerically 0 or 1 occurred

> exp(coef(glm.out))

(Intercept) Sepal.Width Petal.Width Petal.Length

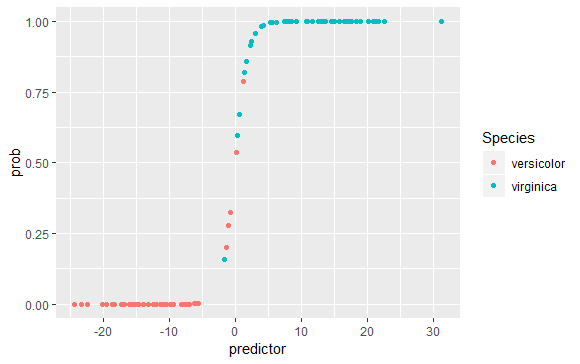
1.138872e-22 2.303132e-04 2.026532e+09 2.629495e+03

# to get odds ratio

Make a plot of the fitted probability as a function of the linear predictor, colored by species identity. Hint: you will have to make a new data frame combining data from the fitted model with data from the *iris.small* data frame.

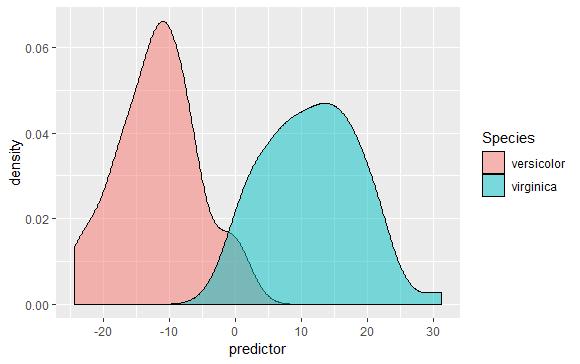
> lr\_data <- data.frame(predictor=glm.out$linear.predictors, prob=glm.out$fitted.values, Species=iris.small$Species)

> ggplot(lr\_data, aes(x=predictor, y=prob, color=Species)) + geom\_point()



Make a density plot that shows how the two species are separated by the linear predictor.

> ggplot(lr\_data, aes(x=predictor, fill=Species)) + geom\_density(alpha=.5)



Assume you have obtained samples from three plants, with measurements as listed below. Predict the likelihood that each of these plants belongs to the species virginica.

> plant1 <- data.frame(Sepal.Length=6.4, Sepal.Width=2.8, Petal.Length=4.6, Petal.Width=1.8)

> plant2 <- data.frame(Sepal.Length=6.3, Sepal.Width=2.5, Petal.Length=4.1, Petal.Width=1.7)

> plant3 <- data.frame(Sepal.Length=6.7, Sepal.Width=3.3, Petal.Length=5.2, Petal.Width=2.3)

> predict(glm.out, plant1, type="response")

1

0.6934611

> predict(glm.out, plant2, type="response")

1

0.06002675

> predict(glm.out, plant3, type="response")

1

0.9999943

Pick a cutoff predictor value at which you would decide that a specimen belongs to virginica rather than versicolor. Calculate how many virginicas you call correctly and how many incorrectly given that choice.

> cutoff <- 0

> virg\_true <- sum(lr\_data$predictor > cutoff & lr\_data$Species=="virginica")

> virg\_false <- sum(lr\_data$predictor <= cutoff & lr\_data$Species=="virginica")

> virg\_true

[1] 49

> virg\_false

[1] 1

Now do the same calculation for versicolor.

|  |
| --- |
| > vers\_true <- sum(lr\_data$predictor <= cutoff & lr\_data$Species=="versicolor")  > vers\_false <- sum(lr\_data$predictor > cutoff & lr\_data$Species=="versicolor")  > vers\_true  [1] 48  > vers\_false  [1] 2 |
|  |
| |  | | --- | |  | |

If we define a call of virginica as a positive and a call of versicolor as a negative, what are the true positive rate (sensitivity, true positives divided by all possible positives) and the true negative rate (specificity, true negatives divided by all possible negatives) in your analysis?

> tp <- virg\_true/(virg\_true + virg\_false)

> tn <- vers\_true/(vers\_true + vers\_false)

> tp

[1] 0.98

> tn

[1] 0.96

Sumber : [**http://wilkelab.org/classes/SDS348/2015\_spring\_worksheets/class11\_solutions.html**](http://wilkelab.org/classes/SDS348/2015_spring_worksheets/class11_solutions.html)