A systems analyst studied the effect of computer programming experience on ability to complete a task within a specified time. Twenty-five persons selected for the study, with varying amounts of computer experience (in months). All programmers where given the same task and the results of their success registered in the file `task.csv`. Results are coded as: Y = 1 if task completed successfully; Y = 0, otherwise.

a) Fit a simple logistic regression to predict the success of a programming task based on the experience of the programmer.

b) Interpret estimated b_1

c) Find a 90% CIs for β_0 and β_1

d) Find a 95% CIs for the odds ratio

e) Predict probability of success of a programmer with 22 months experience

f) Plot the fitted logistic curve along with the scatterplot of the response and the predictor

g) Find the error rate on the entire data set

h) Use the validation set approach with 70% of the data set to estimate the *test error rate*.

```
d1 = read.csv("task.csv",header=T)
head(d1)
str(d1)
#'data.frame':   25 obs. of  2 variables:
# $ Experience: int  14 29 6 25 18 4 18 12 22 6 ...
# $ Success   : int  0 0 0 1 1 0 0 0 1 0 ...

names(d1) <- c("x", "y")
plot(y~x, d1,pch=19,xlab="experience (months)",ylab="success")
grid()

table(d1$y)
#  0  1
# 14 11

# Fit a simple logistic regression
#==============================================================

m1 <- glm(y ~ x, family = binomial(link = "logit"), d1)
# link = "logit" default

# same
yhat = m1$fitted
yhat = predict(m1,type = "response")

res = m1$residuals               # not the right residuals
res = resid(m1,type="response")

d2 = data.frame(d1,yhat,res)
head(d2)
#  x y    yhat        res2
# 14 0 0.31026237 -0.31026237
# 29 0 0.83526292 -0.83526292
#  6 0 0.10999616 -0.10999616
# 25 1 0.72660237  0.27339763
# 18 1 0.46183704  0.53816296
#  4 0 0.08213002 -0.08213002

# interpret b1
#=====================================================================
summary(m1)
# Coefficients:
#             Estimate Std. Error z value Pr(>|z|)
# (Intercept) -3.05970    1.25935  -2.430   0.0151 *
# x            0.16149    0.06498   2.485   0.0129 *

# (Dispersion parameter for binomial family taken to be 1)
#     Null deviance: 34.296  on 24  degrees of freedom
# Residual deviance: 25.425  on 23  degrees of freedom
# AIC: 29.425
```

```
# odds increase by exp(0.16149) = 1.175,
#                  17.5% with each additional month of experience
# (odds of successfully completing the task)


# CIs
#=====================================================================

# for betas
confint(m1,level=0.9)
#                     5 %       95 %
# (Intercept) -5.47908732 -1.2267399
# x            0.06628211  0.2855569

# for the odds ratio
exp(confint(m1,level=0.95))
#                   2.5 %    97.5 %
# (Intercept) 0.002388112 0.4001024
# x           1.051297434 1.3689441

# odds increase between 5% and 36.8%
# with each additional month of experince


# predict probability of success of programmer with 22 months experience
#========================================================================
newval = data.frame(x = 22)
predict(m1,newval,type="response")    # 0.6208116


# Plot Logistic Regression and Loess Fit
#==========================================================================

summary(d1$x)
#   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
#   4.00    9.00   18.00   16.88   24.00   32.00

xx = seq(4,32,len = 200)
newval = data.frame(x = xx)
yy = predict(m1,newval,type="response")   # y-coord of logistic curve

plot(y~x,d1,pch=19,xlab="months")
lines(xx,yy)
text(y~x,d1,labels=rownames(d1),pos=1,offset=0.25,cex=0.4)

# loess fit
m2 = loess(y ~ x, d1,span=1)
yl=predict(m2,newval)
lines(xx,yl,lty=2,col="red")
grid()
```

```
# error rate
#==========================================================================

# if predicted probabs > 0.5 then predict Y=1
n = length(yhat)
ypred=rep("0",n)
ypred[yhat>.5]="1"

table(ypred,d1$y)
# ypred  0  1
#     0 11  3
#     1  3  8

aux=prop.table(table(ypred,d1$y))
# ypred    0    1
#     0 0.44 0.12
#     1 0.12 0.32

# cols are observed vals
# rows are predictions

# off-diag values are % of incorrect predictions
1-sum(diag(aux))        #[1] 0.24

d3=data.frame(d1,ypred)  # ypred is factor now
#     x y ypred
# 1  14 0     0
# 2  29 0     1
# 3   6 0     0
# 4  25 1     1
# 5  18 1     0
# 6   4 0     0
# 7  18 0     0
# 8  12 0     0
# 9  22 1     1
# 10  6 0     0
# 11 30 1     1
# 12 11 0     0
# 13 30 1     1
# 14  5 0     0
# 15 20 1     1
# 16 13 0     0
# 17  9 0     0
# 18 32 1     1
# 19 24 0     1
# 20 13 1     0
# 21 19 0     1
# 22  4 0     0
# 23 28 1     1
# 24 22 1     1
# 25  8 1     0
```

```
# prediction errors
d4=d3[d3$y!=ypred,]
#     x y ypred
# 2  29 0     1
# 5  18 1     0
# 19 24 0     1
# 20 13 1     0
# 21 19 0     1
# 25  8 1     0

points(y~x,d4,col="red",cex=1.4)

# Validation approach - 70% training set
#==========================================================================

set.seed(1)
train <- sample(1:n,0.7*n)
dtrain = d1[train,]
dtest  = d1[-train,]

m3=glm(y~x,dtrain,family=binomial)
fitdtest=predict(m3,dtest,type="response")

n = nrow(dtest)
ypred=rep("0",n)
ypred[fitdtest>.5]="1"

table(ypred,dtest$y)
# ypred 0 1
#     0 5 1
#     1 0 2
# rows are predictions
# cols are observed values
# 1 observation misclassified

prop.table(table(ypred,dtest$y))
aux=prop.table(table(ypred,dtest$y))*100

# ypred    0    1
#     0 62.5 12.5
#     1  0.0 25.0

# test error rate
# off-diag values are % of incorrect predictions
100 - sum(diag(aux))      #[1] 12.50%
```

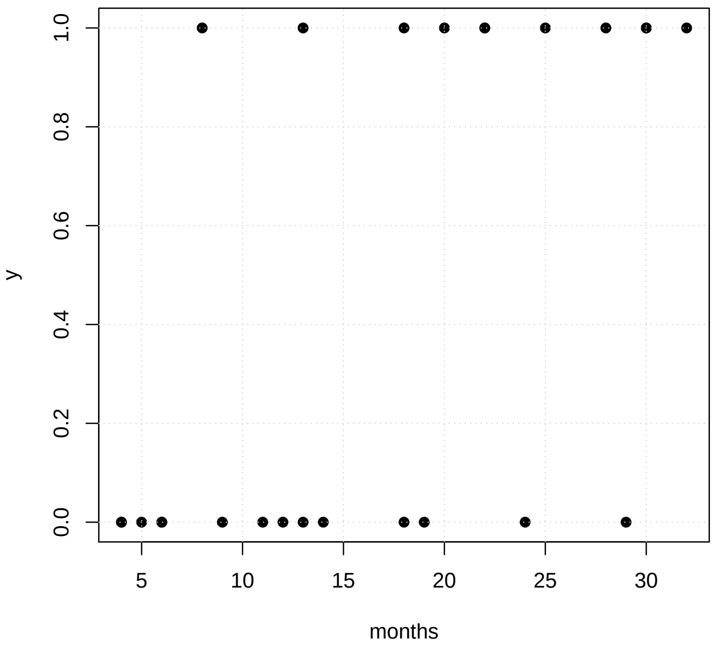

Figure 1: scatterplot of response vs. months of experience

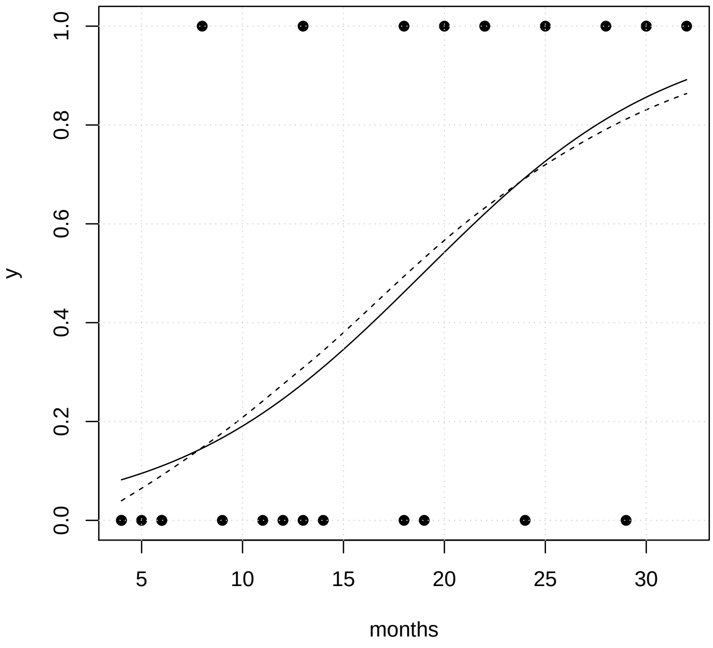

Figure 2: fitted logistic regression