Solutions to Assignment 4

Rongfei Jin

February 7, 2025

1 Conceptual 1

Solution.

$$
\begin{aligned}
\frac{p(X)}{1-p(X)} &= \frac{\frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}}{\frac{1+e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}} - \frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}} \\
&= \frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X} - e^{\beta_0+\beta_1 X}} \\
&= e^{\beta_0+\beta_1 X}
\end{aligned}
$$

■

2 Conceptual 2

Solution.

(a)

$$
\begin{aligned}
\Pr(Y=1|X_1=40, X_2=3.5) &= \frac{e^{\beta_0+\beta_1 X_1+\beta_2 X_2}}{1+e^{\beta_0+\beta_1 X_1+\beta_2 X_2}} \\
&= \frac{e^{-6+0.05\times 40+1\times 3.5}}{1+e^{-6+0.05\times 40+1\times 3.5}} \\
&\approx 0.38
\end{aligned}
$$

(b)

$$
\begin{aligned}
\frac{e^{\beta_0+\beta_1 X_1+\beta_2 X_2}}{1+e^{\beta_0+\beta_1 X_1+\beta_2 X_2}} &= 0.5 \\
e^{\beta_0+\beta_1 X_1+\beta_2 X_2} &= 0.5 + 0.5e^{\beta_0+\beta_1 X_1+\beta_2 X_2} \\
e^{\beta_0+\beta_1 X_1+\beta_2 X_2} &= 1 \\
\beta_0 + \beta_1 X_1 + \beta_2 X_2 &= 0 \\
-6 + 0.05 \times X_1 + 3.5 &= 0 \\
X_1 &= 50
\end{aligned}
$$

■

3 Applied 13(a-d)

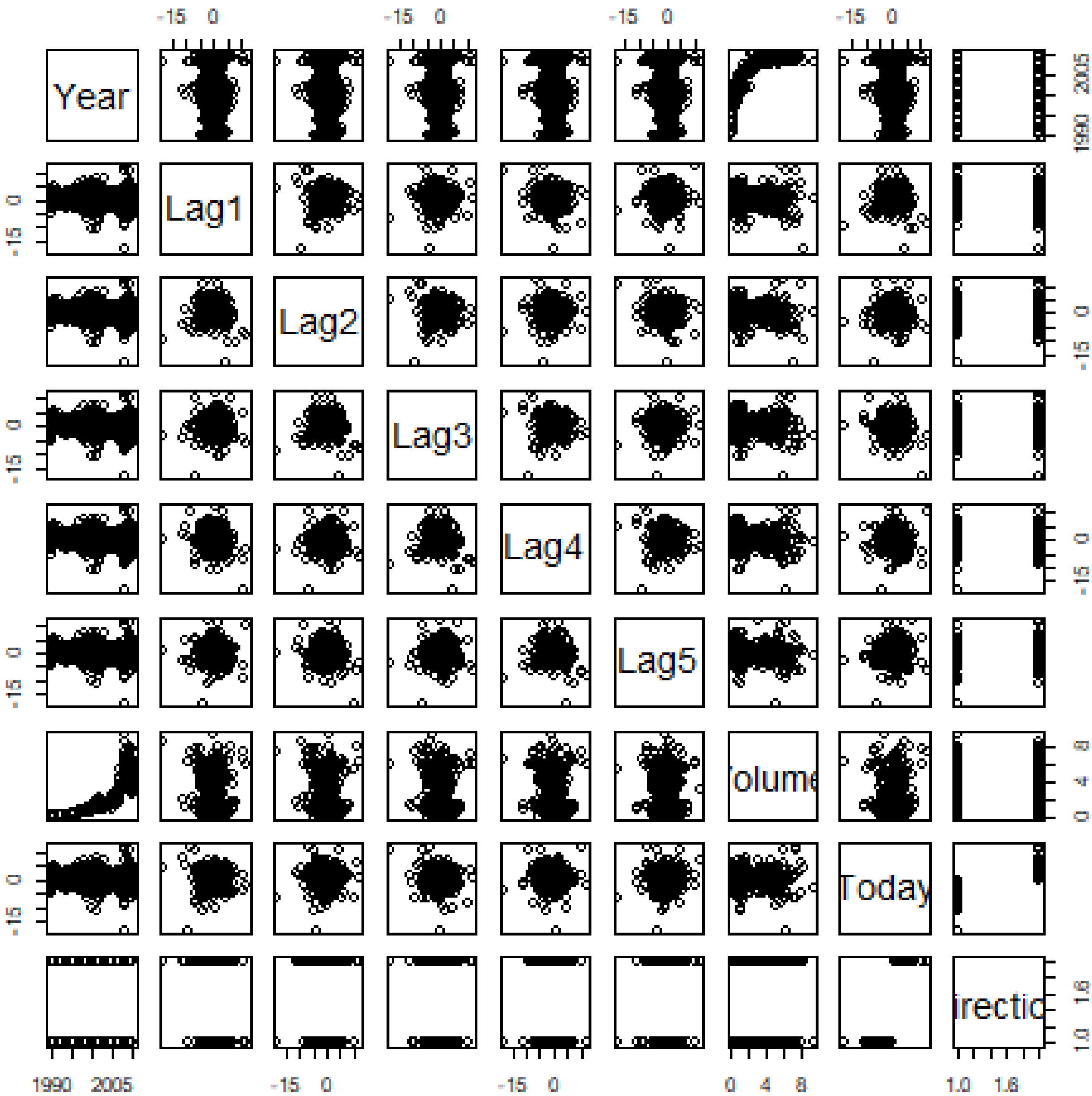

Figure 1: Pairs Plot

(a) The Year and Volumne are strongly positively correlated with a correlation coefficient of 0.84. The Lags have extremly small correlations

(b) Lag 2 is statistically significant

(c) Confusion matrix tells the prediction correctly identify 54 Down and incorrectly identify 48 Up as Down. It correctly identify 557 Up and incorrectly identify 430 Down as Up. The number of false positive error is 430 and the number of false negative error is 48.

(d) Accruacy 0.625

```r
# applied 13a-d

library(ISLR2)

summary(Weekly)

pairs(Weekly)

cor(subset(Weekly, select= -c(Direction)))

# logistic regression

glm.fit <- glm(Direction ~ Lag1 + Lag2 + Lag3 + Lag4 + Lag5 + Volume, data = Weekly, family = binomial)

summary(glm.fit)

# confusion matrix
glm.probs <- predict(glm.fit, type = "response")

glm.pred <- ifelse(glm.probs > 0.5, "Up", "Down")

# Ensure levels match
glm.pred <- factor(glm.pred, levels = levels(Weekly$Direction))

# Confusion matrix
conf_matrix <- table(Predicted = glm.pred, Actual = Weekly$Direction)
print(conf_matrix)

#           Actual
# Predicted Down  Up
#      Down   54  48
#      Up    430 557

# select year 2008 and earlier for training
train <- Weekly[Weekly$Year < 2009, ]
test <- Weekly[Weekly$Year >= 2009, ]

glm.fit <- glm(Direction ~ Lag2, data = train, family = binomial)
glm.probs <- predict(glm.fit, test, type = "response")
glm.pred <- ifelse(glm.probs > 0.5, "Up", "Down")
glm.pred <- factor(glm.pred, levels = levels(test$Direction))
conf_matrix <- table(Predicted = glm.pred, Actual = test$Direction)
print(conf_matrix)

#           Actual
# Predicted Down Up
#      Down    9  5
#      Up     34 56

# accuracy
accuracy <- sum(diag(conf_matrix)) / sum(conf_matrix)
print(accuracy)
# 0.625
```

4 Applied 14(a-c, f)

(a) Shown in code below

(b) Cylinder, Displacement, Horsepower and Weights are good predictors since there's less overlap between the two classes. The other predictors have more overlap between the two classes.

(c) Shown in code below

(f) error rate is 0.09

```
library(ISLR2)
cols <- colnames(subset(Auto, select = -c(name)))

median <- median(Auto$mpg)
Auto$mpg01 <- ifelse(Auto$mpg > median, TRUE, FALSE)
Auto$cylinders <- as.factor(Auto$cylinders)

str(Auto)

ncols <- length(cols)
w <- ceiling(sqrt(ncols))
par(mfrow = c(w, w))
for (col in cols) {
  boxplot(Auto[[col]] ~ Auto$mpg01, ylab = col, xlab = "mpg01")
  title(sprintf("Boxplot of %s", col))
}

rand_idx <- sample(1:nrow(Auto), nrow(Auto) * 0.8)
train <- Auto[rand_idx, ]
test <- Auto[-rand_idx, ]

# logistic regression

glm.fit <- glm(mpg01 ~ cylinders + displacement + horsepower + weight, , data = train, family = binomial

# compute test error
glm.probs <- predict(glm.fit, test, type = "response")
glm.pred <- ifelse(glm.probs > 0.5, TRUE, FALSE)
error_rate <- mean(glm.pred != test$mpg01)
error_rate
```

5 Applied 16

```r
library(ISLR2)
str(Boston)

Boston$y <- ifelse(Boston$crim > median(Boston$crim), TRUE, FALSE)
Boston <- subset(Boston, select = -c(crim))

rand_idx <- sample(1:nrow(Boston), nrow(Boston) * 0.8)
train <- Boston[rand_idx, ]
test <- Boston[-rand_idx, ]

glm.fit <- glm(y ~ ., data = train, family = binomial)

summary(glm.fit)
#              Estimate Std. Error z value Pr(>|z|)
# (Intercept) -39.579301   6.492978  -6.096 1.09e-09 ***
# zn           -0.065054   0.034724  -1.873  0.06101 .
# indus        -0.097390   0.050344  -1.935  0.05305 .
# chas          0.591790   0.752860   0.786  0.43183
# nox          47.361449   7.972041   5.941 2.83e-09 ***
# rm           -0.556126   0.764264  -0.728  0.46682
# age           0.013887   0.013074   1.062  0.28816
# dis           0.638130   0.221236   2.884  0.00392 **
# rad           0.611631   0.173401   3.527  0.00042 ***
# tax          -0.003977   0.003063  -1.298  0.19414
# ptratio       0.428036   0.137806   3.106  0.00190 **
# lstat         0.089197   0.054505   1.636  0.10174
# medv          0.196003   0.078097   2.510  0.01208 *

# confusion matrix
glm.probs <- predict(glm.fit, test, type = "response")
glm.pred <- ifelse(glm.probs > 0.5, TRUE, FALSE)
conf_matrix <- table(Predicted = glm.pred, Actual = test$y)
print(conf_matrix)
#          Actual
# Predicted FALSE TRUE
#     FALSE    44    9
#     TRUE      4   45

# accuracy
accuracy <- sum(diag(conf_matrix)) / sum(conf_matrix)
print(accuracy)
# [1] 0.872549
```

Based on the logistic regression model, the nox, dis, rad, ptratio and medv are statistically significant predictors of crime rate where zn, chas have almost significant p-value. The nox, dis, rad, ptratio are positively correlated with crime rate while zn and chas are negatively correlated with crime rate. The model has an accuracy of 0.87.

6 Additional 1

Solution.

(a)

$$\frac{0}{1+e^x} < \frac{e^x}{1+e^x} < \frac{e^x}{e^x} \implies 0 < \frac{e^x}{1+e^x} < 1$$

(b)

$$\begin{aligned}
d(\frac{e^x}{1+e^x}) &= \frac{e^x(1+e^x) - e^x e^x}{(1+e^x)^2} \\
&= \frac{e^x}{(1+e^x)^2} \\
&= \frac{1}{1+e^x}\frac{e^x}{1+e^x} \\
&= \frac{1+e^x-e^x}{1+e^x}\frac{e^x}{1+e^x} \\
&= (1-\frac{e^x}{1+e^x})\frac{e^x}{1+e^x} \\
&= (1-\phi(x))\phi(x)
\end{aligned}$$

■

7 Additional 2

Solution.

$$\begin{aligned}
\frac{d}{dp}(L(p|y)) &= -\frac{y}{p} + \frac{1-y}{1-p} \\
&= \frac{-y(1-p)+p-py}{p(1-p)} \\
&= \frac{-y+yp+p-py}{p(1-p)} \\
&= \frac{p-y}{p(1-p)}
\end{aligned}$$

■

8 Additional 3

Solution.

$$\begin{aligned}
\frac{d}{d\eta}l(\eta|y) &= -\frac{-y\phi'(\eta)}{\phi(\eta)} + \frac{(1-y)\phi'(\eta)}{1-\phi(\eta)} \\
&= -y[1-\phi(\eta)] + (1-y)\phi(\eta) \\
&= -y + \phi(\eta)
\end{aligned}$$

■

9 Additional 4

Solution.

$$
\begin{aligned}
\frac{\partial}{\partial \beta_0} l(\beta_0, \beta_1 | y, x) &= \frac{-y\phi'(\beta_0 + \beta_1 x) \cdot 1}{\phi(\beta_0 + \beta_1 x)} + \frac{(1-y)\phi'(\beta_0 + \beta_1 x) \cdot 1}{1 - \phi(\beta_0 + \beta_1 x)} \\
&= \phi(\beta_0 + \beta_1 x) - y
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial}{\partial \beta_1} l(\beta_0, \beta_1 | y, x) &= \frac{-y\phi'(\beta_0 + \beta_1 x) \cdot x}{\phi(\beta_0 + \beta_1 x)} + \frac{(1-y)\phi'(\beta_0 + \beta_1 x) \cdot x}{1 - \phi(\beta_0 + \beta_1 x)} \\
&= -y[x - \phi(\beta_0 + \beta_1 x) \cdot x] + (1-y)\phi(\beta_0 + \beta_1 x) \cdot x \\
&= (\phi(\beta_0 + \beta_1 x) - y)x
\end{aligned}
$$

■

10 Additional 5

Solution.

$$
\begin{aligned}
\frac{\partial^2}{\partial \beta_0^2} l(\beta_0, \beta_1 | y, x) &= \frac{\partial}{\partial \beta_0} \frac{\partial}{\partial \beta_0} l(\beta_0, \beta_1 | y, x) \\
&= \frac{\partial}{\partial \beta_0} (\phi(\beta_0 + \beta_1 x) - y) \\
&= \phi'(\beta_0 + \beta_1 x) \\
&= \phi(\beta_0 + \beta_1 x)(1 - \phi(\beta_0 + \beta_1 x))
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial}{\partial \beta_0 \partial \beta_1} l(\beta_0, \beta_1 | y, x) &= \frac{\partial}{\partial \beta_0} \frac{\partial}{\partial \beta_1} l(\beta_0, \beta_1 | y, x) \\
&= \frac{\partial}{\partial \beta_0} (\phi(\beta_0 + \beta_1 x) - y)x \\
&= \phi'(\beta_0 + \beta_1 x)x \\
&= x\phi(\beta_0 + \beta_1 x)(1 - \phi(\beta_0 + \beta_1 x))
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial^2}{\partial \beta_1^2} l(\beta_0, \beta_1 | y, x) &= \frac{\partial}{\partial \beta_1} \frac{\partial}{\partial \beta_1} l(\beta_0, \beta_1 | y, x) \\
&= \frac{\partial}{\partial \beta_1} (\phi(\beta_0 + \beta_1 x) - y)x \\
&= (\phi'(\beta_0 + \beta_1 x)x - 0) \\
&= x^2\phi(\beta_0 + \beta_1 x)(1 - \phi(\beta_0 + \beta_1 x))
\end{aligned}
$$

■

11 Additional 6

Solution.

$$
\begin{aligned}
\nabla \mathcal{L}(\beta|y,X) &= \begin{bmatrix} \frac{\partial}{\partial \beta_0}\mathcal{L}(\beta|y,X) \\ \frac{\partial}{\partial \beta_1}\mathcal{L}(\beta|y,X) \end{bmatrix} \\
&= \begin{bmatrix} \sum_{i=1}^n \frac{\partial}{\partial \beta_0} l(\beta|y_i,x_i) \\ \sum_{i=1}^n \frac{\partial}{\partial \beta_1} l(\beta|y_i,x_i) \end{bmatrix} \\
&= \sum_{i=1}^n \begin{bmatrix} \phi(\beta_0+\beta_1 x_i) - y_i \\ (\phi(\beta_0+\beta_1 x_i) - y_i)x_i \end{bmatrix}
\end{aligned}
$$

$$
\begin{aligned}
\nabla^2 \mathcal{L}(\beta|y,X) &= \begin{bmatrix} \frac{\partial^2}{\partial \beta_0^2}\mathcal{L}(\beta|y,X) & \frac{\partial}{\partial \beta_0 \partial \beta_1}\mathcal{L}(\beta|y,X) \\ \frac{\partial}{\partial \beta_0 \partial \beta_1}\mathcal{L}(\beta|y,X) & \frac{\partial^2}{\partial \beta_1^2}\mathcal{L}(\beta|y,X) \end{bmatrix} \\
&= \begin{bmatrix} \sum_{i=1}^n \frac{\partial^2}{\partial \beta_0^2} l(\beta|y_i,x_i) & \sum_{i=1}^n \frac{\partial}{\partial \beta_0 \partial \beta_1} l(\beta|y_i,x_i) \\ \sum_{i=1}^n \frac{\partial}{\partial \beta_0 \partial \beta_1} l(\beta|y_i,x_i) & \sum_{i=1}^n \frac{\partial^2}{\partial \beta_1^2} l(\beta|y_i,x_i) \end{bmatrix} \\
&= \sum_{i=1}^n \begin{bmatrix} \phi(\beta_0+\beta_1 x_i)(1-\phi(\beta_0+\beta_1 x_i)) & x_i\phi(\beta_0+\beta_1 x_i)(1-\phi(\beta_0+\beta_1 x_i)) \\ x_i\phi(\beta_0+\beta_1 x_i)(1-\phi(\beta_0+\beta_1 x_i)) & x_i^2\phi(\beta_0+\beta_1 x_i)(1-\phi(\beta_0+\beta_1 x_i)) \end{bmatrix} \\
&= \sum_{i=1}^n \phi(\beta_0+\beta_1 x_i)(1-\phi(\beta_0+\beta_1 x_i)) \begin{bmatrix} 1 & x_i \\ x_i & x_i^2 \end{bmatrix}
\end{aligned}
$$

■

12 Additional 7

The estimated beta values from newton's method is exactly the same as the beta values from glm function.

```
### Clear workspace

rm(list=ls())

### Logistic function

phi <- function(z) 1/(1+exp(-z))

###
### We simulate the data
###

### Set the size of the problem

n <- 25
p <- 1

### Simulate the x-values

sigx <- 1 # standard deviation of the x-values
X <- cbind(1,matrix(rnorm(n*p,0,sigx),n,p))
X

### Simulate the true beta values
```

```r
sigb <- 1 # standard deviation of the true beta-values
beta <- rnorm(p+1,0,sigb)

### Simulate the y-values

eta <- as.numeric(X%*%beta)
y <- rep(0,n)
for(i in 1:n){
        pr <- phi(eta[i])
        y[i] <- rbinom(1,1,pr)
}

###
### We estimate the parameters from the data.
###

grad <- function(beta,y,X){
        eta <- as.numeric(X%*%beta)
        colSums(sweep(X,1,phi(eta)-y,"*"))
}

hess <- function(beta,y,X){
        peta <- phi(as.numeric(X%*%beta))
        t(X)%*%sweep(X,1,peta,"*")
}

logreg <- function(y,X,eps=1e-6){
        pp1 <- ncol(X) # pp1 = p+1
        beta1 <- rnorm(pp1)
        beta0 <- beta1+1
        ct <- 0
        while(max(abs(beta1-beta0))>eps){
                ct <- ct+1
                beta0 <- beta1
                beta1 <- beta0-as.numeric(solve(hess(beta0,y,X))%*%grad(beta0,y,X))
                print(max(abs(beta1-beta0)))
        }
        beta1
}

betahat <- logreg(y,X)

betahat

m <- glm(y~.,data=data.frame(X[,-1]),family=binomial)

data.frame(cbind(betahat,m$coef))

### Plot what we have
### Along with the true logistic function

# the truth
par(mfrow=c(1,1))
plot(eta,y,pch=19,
        main="binary response versus linear predictor",
```

```
        xlab="linear predictor (eta)",
        ylab="y")
ix <- sort(eta,index.return=TRUE)$ix
lines(eta[ix],phi(eta[ix]),lwd=3)
abline(h=c(0,1/2,1))
abline(v=0)

# the estimated model

etahat <- as.numeric(X%*%betahat)
ix <- sort(etahat,index.return=TRUE)$ix
lines(etahat[ix],phi(etahat[ix]),lwd=2,lty=2,col='red')

# misclassification rate

yhat <- 1*(phi(etahat)>0.5)
mean(y!=yhat)
```