Homework 2

Mari Sanders

2025-02-20

Problem 1

Linear Regression

- Uses continuous outcome variables.
- Assumes a linear relationship between the outcome and predictors.
- Coefficients represent the change in the dependent variable for a one-unit change in an independent variable.
- Errors are assumed to be normally distributed.
- Output is continuous and can take any real number.
- Uses RMSE or MSE to evaluate model fit

Logistic Regression

- Uses binary or categorical outcome variables.
- Models non-linear relationships using the logit function and probabilities.
- Coefficients are expressed as log odds, meaning a unit change in an independent variable affects the log odds of the outcome.
- Errors follow a binomial distribution.
- Output is a probability between 0 and 1.
- Uses deviance and score to evaluate model fit

Problem 2

$Odds = \frac{pi}{1-pi}$, which is the probability of an event occurring over the probability of the event not occurring. If you do e^{β} to the coefficients in logistic regression, you will get the odds ratio. This is interpretable because it is in terms of the original equation and also easy to understand.

$log(odds) = log(pi/1 - pi)$, which is the probability of odds ratio given that the values are in terms of log, which is less interpretable. The coefficients β in logistic regression are in terms of log odds, such that a change in log-odds for a one-unit increase in the outcome.

Problem 3

L1 Regularization: Also called a lasso regression, adds the absolute value of the sum of coefficients as a penalty term to the loss function. Lasso makes some of the coefficients go to zero

L2 Regularization: Also called a ridge regression, adds the squared sum of coefficients as the penalty term to the loss function.Ridge shrinks coefficients but does not make any coefficients go to zero.

Problem 4

Problem 5

Logit

```
dose <- c(0,1,2,3,4)
dying <- c(2, 8, 15, 23, 27)
data <- data.frame(dose, dying)

resp <- cbind(died = data$dying, alive = 30 - data$dying)
pred <- data$dose

fitlogit <- glm(resp~pred,family=binomial(link='logit'),data= data)
summary(fitlogit)
```

```
##
## Call:
## glm(formula = resp ~ pred, family = binomial(link = "logit"),
##     data = data)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -2.3238     0.4179  -5.561 2.69e-08 ***
## pred          1.1619     0.1814   6.405 1.51e-10 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 64.76327  on 4  degrees of freedom
## Residual deviance:  0.37875  on 3  degrees of freedom
## AIC: 20.854
##
## Number of Fisher Scoring iterations: 4
```

```
beta_1 <- exp(fitlogit$coefficient[2])
confint(fitlogit)
```

```
## Waiting for profiling to be done...
##                  2.5 %    97.5 %
## (Intercept) -3.2060617 -1.557314
## pred         0.8301789  1.546129
```

```r
confint_logit <- exp(confint(fitlogit))
```

```
## Waiting for profiling to be done...
```

```r
devfitlogit <- sum(residuals(fitlogit,type='deviance')^2)
predictionlogit <- predict(fitlogit, data.frame(dose=0.01), se.fit=TRUE,type='response')
```

```
## Warning: 'newdata' had 1 row but variables found have 5 rows
```

3.1959837

The CI for β_1 is (2.29372916, 4.6932687).

- Deviance: 0.3787483

\$p(dying|X = 0.01) = \$ c(1 = 0.089171765476152, 2 = 0.238323135714362, 3 = 0.5, 4 = 0.761676864285638, 5 = 0.910828234523848), c(1 = 0.0339409177453659, 2 = 0.0500085378173001, 3 = 0.0518311765437439, 4 = 0.0500085378166291, 5 = 0.0339409177449701), 1

Probit

```r
fitprobit <- glm(resp~pred,family=binomial(link='probit'),data= data)
summary(fitprobit)
```

```
##
## Call:
## glm(formula = resp ~ pred, family = binomial(link = "probit"),
##     data = data)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.37709    0.22781  -6.045 1.49e-09 ***
## pred         0.68638    0.09677   7.093 1.31e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 64.76327  on 4  degrees of freedom
## Residual deviance:  0.31367  on 3  degrees of freedom
## AIC: 20.789
##
## Number of Fisher Scoring iterations: 4
```

```
confint(fitprobit)
```

```
## Waiting for profiling to be done...

##                   2.5 %     97.5 %
## (Intercept) -1.8436290 -0.9442144
## pred         0.5033779  0.8840139
```

```
confint_probit <- exp(confint(fitprobit))
```

```
## Waiting for profiling to be done...
```

```
devprobit <- sum(residuals(fitprobit,type='deviance')^2)

predictionprobit <- predict(fitprobit, data.frame(dose = 0.01), se.fit = TRUE,type = 'response')
```

```
## Warning: 'newdata' had 1 row but variables found have 5 rows
```

1.9865113

The CI for β_1 is (1.6542999, 2.420596)

- Deviance: 0.3136684

$p(dying|X = 0.01) = $ c(1 = 0.0842418591382518, 2 = 0.244873352075345, 3 = 0.49827210025563, 4 = 0.752396116679648, 5 = 0.914411217164809), c(1 = 0.0352113067665975, 2 = 0.0484217481908314, 3 = 0.0477208305469469, 4 = 0.0485748290137956, 5 = 0.0355417297609161), 1

Cloglog

```
fitcloglog <- glm(resp~pred, family = binomial(link = "cloglog"), data = data)
summary(fitcloglog)
```

```
##
## Call:
## glm(formula = resp ~ pred, family = binomial(link = "cloglog"),
##     data = data)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -1.9942     0.3126  -6.378 1.79e-10 ***
## pred          0.7468     0.1094   6.824 8.86e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 64.7633  on 4  degrees of freedom
## Residual deviance:  2.2305  on 3  degrees of freedom
## AIC: 22.706
##
## Number of Fisher Scoring iterations: 5
```

```
confint(fitcloglog)
```

```
## Waiting for profiling to be done...
```

```
##                  2.5 %    97.5 %
## (Intercept) -2.6445686 -1.425454
## pred         0.5459821  0.964783
```

```
confint_cloglog <- exp(confint(fitcloglog))
```

```
## Waiting for profiling to be done...
```

```
devcloglog <- sum(residuals(fitcloglog,type = 'deviance')^2)

predictcloglog <- predict(fitcloglog, data.frame(dose =0.01), se.fit=TRUE,type='response')
```

```
## Warning: 'newdata' had 1 row but variables found have 5 rows
```

2.1102364

The CI for β_1 is (1.72630302, 2.6242181)

- Deviance: 2.2304792

$p(dying|X = 0.01) = $ c(1 = 0.127269995587194, 2 = 0.249690899241929, 3 = 0.4545910450001, 4 = 0.721765494688089, 5 = 0.932771540962179), c(1 = 0.0371427130542408, 2 = 0.0469696252835337, 3 = 0.0481245582243864, 4 = 0.0488533250401917, 5 = 0.0365000247906957), 1

b)

```
beta0 <-  fitlogit$coefficients[1]
beta1 <- fitlogit$coefficients[2]
betacov <- vcov(fitlogit)
x0fit <- -beta0/beta1
exp(x0fit)
```

```
## (Intercept)
##    7.389056
```

```
varx0=betacov[1,1]/(beta1^2)+betacov[2,2]*(beta0^2)/(beta1^4)-2*betacov[1,2]*beta0/(beta1^3)
c(x0fit,sqrt(varx0)) # point est and se
```

```
## (Intercept)        pred
##   2.0000000   0.1784367
```

```
exp((x0fit+c(qnorm(0.05),-qnorm(0.05))*sqrt(varx0))) # 90% CI for LD50
```

```
## [1] 5.509631 9.909583
```

We are 90% confident that the LD50 for this bioassay study is between 5.509631 and 9.9095.

Problem 6

```
amount <- seq(from = 10, to = 90, by = 5)
offers <- c(4, 6, 10, 12, 39, 36, 22, 14, 10, 12, 8, 9, 3, 1, 5, 2, 1)
enrolls <- c(0, 2, 4, 2, 12, 14, 10, 7, 5, 5, 3, 5, 2, 0, 4, 2, 1)
declined <- offers - enrolls
data <- data.frame(amount, offers, enrolls, declined)

mphfit <- glm(cbind(enrolls, declined) ~ amount,
              family=binomial(link='logit'),data= data)
summary(mphfit)
```

```
##
## Call:
## glm(formula = cbind(enrolls, declined) ~ amount, family = binomial(link = "logit"),
##     data = data)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.64764    0.42144  -3.910 9.25e-05 ***
## amount       0.03095    0.00968   3.197  0.00139 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 21.617  on 16  degrees of freedom
## Residual deviance: 10.613  on 15  degrees of freedom
## AIC: 51.078
##
## Number of Fisher Scoring iterations: 4
```

a)

```
devmph <- sum(residuals(mphfit,type='deviance')^2)

sum(residuals(mphfit,type='pearson')^2)
```

```
## [1] 8.814299
```

```
pval=1-pchisq(devmph,17-2)


hl <- hoslem.test(mphfit$y, fitted(mphfit), g=10)  # fitted: returns \hat{pi}
hl
```

```
##
##  Hosmer and Lemeshow goodness of fit (GOF) test
##
## data:  mphfit$y, fitted(mphfit)
## X-squared = 1.6111, df = 8, p-value = 0.9907
```

b)

```
confint(mphfit)
```

```
## Waiting for profiling to be done...
```

```
##                   2.5 %      97.5 %
## (Intercept) -2.50117202 -0.84287573
## amount       0.01245819  0.05060401
```

```
exp(confint(mphfit))
```

```
## Waiting for profiling to be done...
```

```
##                  2.5 %    97.5 %
## (Intercept) 0.08198885 0.4304708
## amount      1.01253611 1.0519063
```

For a 1 unit increase in amount, the odds of enrollment increases by 1.031434 .

We are 95% confident that the the odds ratio is between 1.01253611 and 1.0519063 meaning that a unit increase in the amount of money increases the odds of enrollment by 2%-5%. Since the confidence interval does not include 1, we can say that the effect is statistically significant.

c)