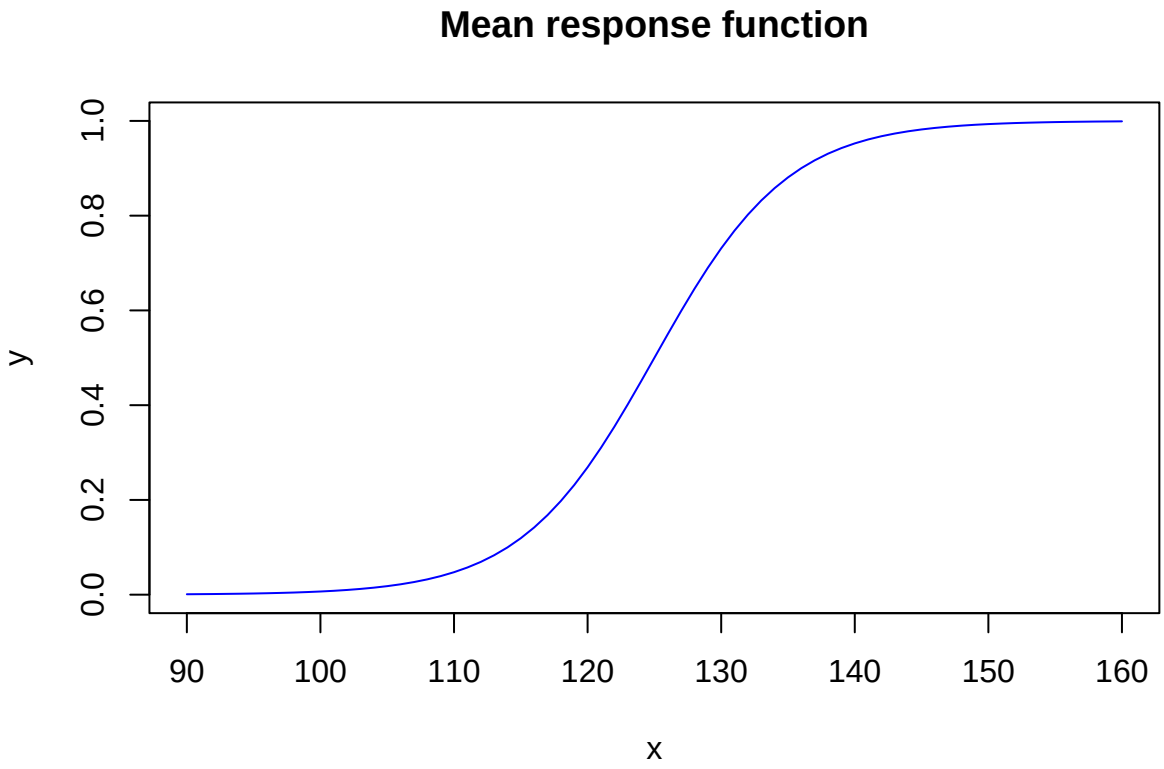
HW11 - Logistic Regression

Madhu Peduri

July 14, 2021

0.0.1 Problem 1(a)

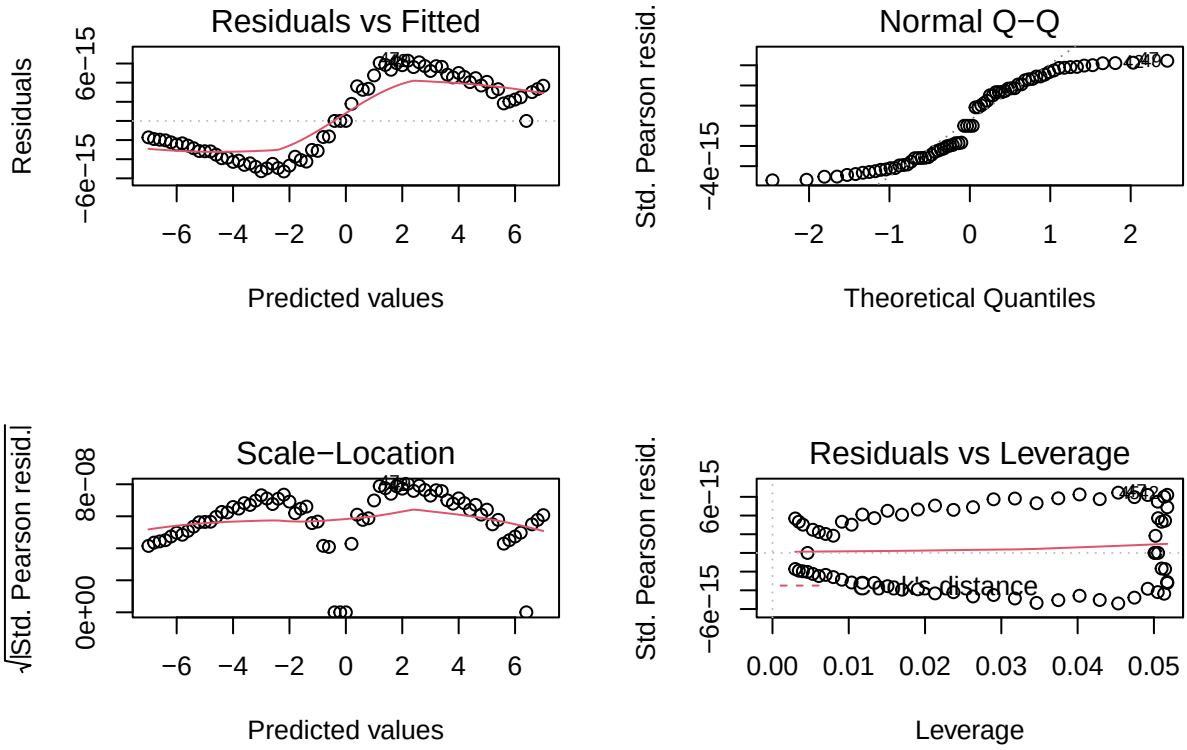
```
# Independent variable x
x <- seq(90, 160, 1)

# coefficients of linear predictor
intercept <- -25
coef1 <- 0.2

# link function, mean response function of a logistic
# regression model
mrf <- function(var) {
    val <- exp(-25 + 0.2 * var)/(1 + exp(-25 + 0.2 * var))
    val
}

# Response variable
y <- sapply(x, mrf)

# plot the model
plot(x, y, type = "l", col = "blue", main = "Mean response function")
```




```
lm1 <- glm(y ~ x, family = binomial)
```

```
## Warning in eval(family$initialize): non-integer #successes in a binomial glm!
```

```
par(mfrow = c(2, 2))
plot(lm1)
```



0.0.2 Problem 1(b)

```
# logit(x) = log(x/(1-x))
x_val <- (logit(0.5) - intercept)/coef1
sprintf("Value of x when probability Pr(Y=0.5) is: %.2f", x_val)
```

```
## [1] "Value of x when probability Pr(Y=0.5) is: 125.00"
```

```
# Compare with glm model
dose.p(lm1, p = 0.5)
```

```
##          Dose   SE
## p = 0.5:  125 2.24
```

0.0.3 Problem 1(c)

```
# Odds when x=150
o1 <- mrf(150)/(1 - mrf(150))
sprintf("Odds when x=150: %.3f", o1)
```

```
## [1] "Odds when x=150: 148.413"
```

```
# Odds when x=151
o2 <- mrf(151)/(1 - mrf(151))
sprintf("Odds when x=151: %.3f", o2)
```

```
## [1] "Odds when x=151: 181.272"
```

```r
# Ratio of odds
r <- o2/o1
sprintf("Ratio of odds 151 to 150: %.3f", r)
```

```
## [1] "Ratio of odds 151 to 150: 1.221"
```

```r
sprintf("exp of coefficient: %.3f", exp(coef1))
```

```
## [1] "exp of coefficient: 1.221"
```

0.0.3.1 Observation

- The ratio of odds for two values x=151 to x=150, a unit measure in x, is equal to the exp(coefficient) of the linear predictor.

0.0.4 Problem 2(a)

```r
# read the data
q2 <- read.table("Q2.txt", quote = "\"", comment.char = "")
colnames(q2) <- c("y", "income", "age")

# print summary
str(q2)
```

```
## 'data.frame':    33 obs. of  3 variables:
##  $ y     : num  0 0 1 0 0 1 1 1 0 1 ...
##  $ income: num  32 45 60 53 25 68 82 38 67 92 ...
##  $ age   : num  3 2 2 1 4 1 2 5 2 2 ...
```

```r
mod2 <- glm(y ~ income + age, family = binomial(link = "logit"),
    data = q2)
summary(mod2)
```

```
##
## Call:
## glm(formula = y ~ income + age, family = binomial(link = "logit"),
##     data = q2)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.619  -0.895  -0.588   0.965   2.085
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -4.7393     2.1019   -2.25    0.024 *
## income        0.0677     0.0281    2.41    0.016 *
## age           0.5986     0.3901    1.53    0.125
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 44.987  on 32  degrees of freedom
## Residual deviance: 36.690  on 30  degrees of freedom
## AIC: 42.69
##
```

```
## Number of Fisher Scoring iterations: 4
```

```
str(summary(mod2))
```

```
## List of 17
##  $ call          : language glm(formula = y ~ income + age, family = binomial(link = "logit"), data =
##  $ terms         :Classes 'terms', 'formula'  language y ~ income + age
##   .. ..- attr(*, "variables")= language list(y, income, age)
##   .. ..- attr(*, "factors")= int [1:3, 1:2] 0 1 0 0 0 1
##   .. .. ..- attr(*, "dimnames")=List of 2
##   .. .. .. ..$ : chr [1:3] "y" "income" "age"
##   .. .. .. ..$ : chr [1:2] "income" "age"
##   .. ..- attr(*, "term.labels")= chr [1:2] "income" "age"
##   .. ..- attr(*, "order")= int [1:2] 1 1
##   .. ..- attr(*, "intercept")= int 1
##   .. ..- attr(*, "response")= int 1
##   .. ..- attr(*, ".Environment")=<environment: R_GlobalEnv>
##   .. ..- attr(*, "predvars")= language list(y, income, age)
##   .. ..- attr(*, "dataClasses")= Named chr [1:3] "numeric" "numeric" "numeric"
##   .. .. ..- attr(*, "names")= chr [1:3] "y" "income" "age"
##  $ family        :List of 12
##   ..$ family    : chr "binomial"
##   ..$ link      : chr "logit"
##   ..$ linkfun   :function (mu)
##   ..$ linkinv   :function (eta)
##   ..$ variance  :function (mu)
##   ..$ dev.resids:function (y, mu, wt)
##   ..$ aic       :function (y, n, mu, wt, dev)
##   ..$ mu.eta    :function (eta)
##   ..$ initialize: language {     if (NCOL(y) == 1) { ...
##   ..$ validmu   :function (mu)
##   ..$ valideta  :function (eta)
##   ..$ simulate  :function (object, nsim)
##   ..- attr(*, "class")= chr "family"
##  $ deviance      : num 36.7
##  $ aic           : num 42.7
##  $ contrasts     : NULL
##  $ df.residual   : int 30
##  $ null.deviance : num 45
##  $ df.null       : int 32
##  $ iter          : int 4
##  $ deviance.resid: Named num [1:33] -0.87 -0.976 0.965 -0.954 -0.916 ...
##   ..- attr(*, "names")= chr [1:33] "1" "2" "3" "4" ...
##  $ coefficients  : num [1:3, 1:4] -4.7393 0.0677 0.5986 2.1019 0.0281 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:3] "(Intercept)" "income" "age"
##   .. ..$ : chr [1:4] "Estimate" "Std. Error" "z value" "Pr(>|z|)"
##  $ aliased       : Named logi [1:3] FALSE FALSE FALSE
##   ..- attr(*, "names")= chr [1:3] "(Intercept)" "income" "age"
##  $ dispersion    : num 1
##  $ df            : int [1:3] 3 30 3
##  $ cov.unscaled  : num [1:3, 1:3] 4.418185 -0.0515 -0.737817 -0.0515 0.000787 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:3] "(Intercept)" "income" "age"
##   .. ..$ : chr [1:3] "(Intercept)" "income" "age"
```

```
##  $ cov.scaled    : num [1:3, 1:3] 4.418185 -0.0515 -0.737817 -0.0515 0.000787 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:3] "(Intercept)" "income" "age"
##   .. ..$ : chr [1:3] "(Intercept)" "income" "age"
##  - attr(*, "class")= chr "summary.glm"
```

```
# Coefficients
mod2$coefficients
```

```
## (Intercept)      income         age
##     -4.7393      0.0677      0.5986
```

```
sprintf("Beta0(Intercept):%.4f", mod2$coefficients[1])
```

```
## [1] "Beta0(Intercept):-4.7393"
```

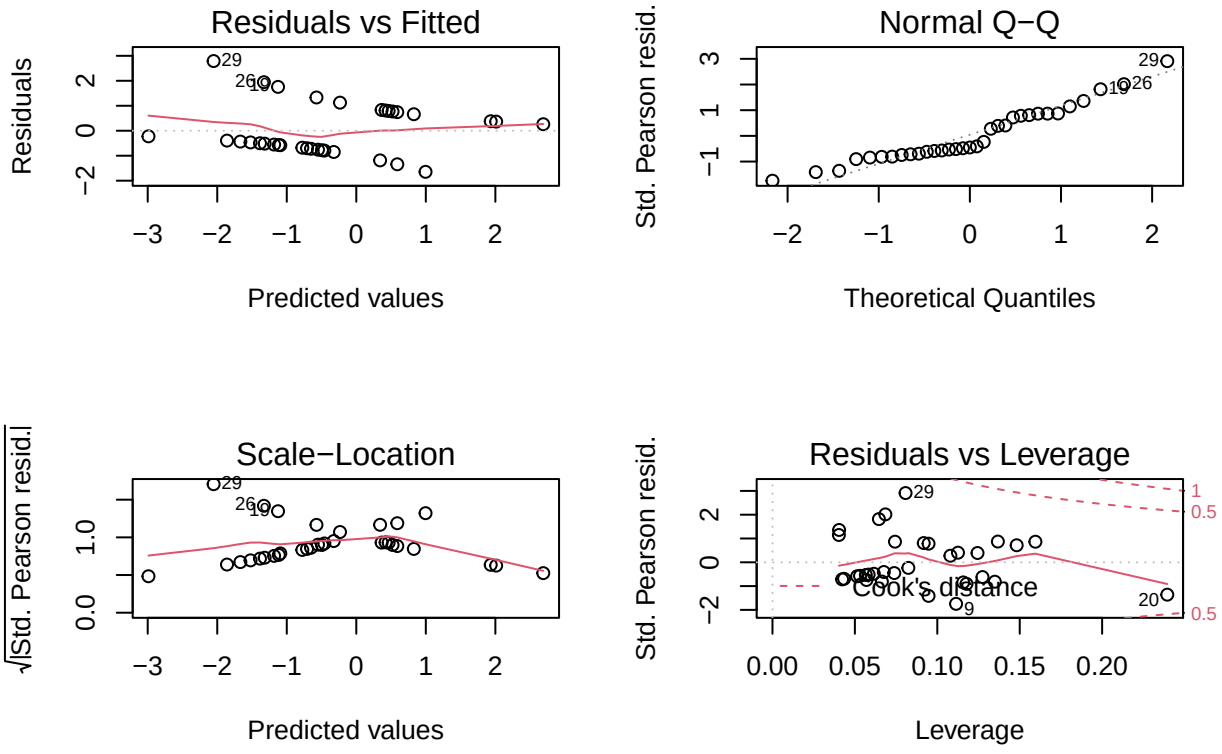
```
sprintf("Beta1(Coefficient1):%.4f", mod2$coefficients[2])
```

```
## [1] "Beta1(Coefficient1):0.0677"
```

```
sprintf("Beta2(Coefficient2):%.4f", mod2$coefficients[3])
```

```
## [1] "Beta2(Coefficient2):0.5986"
```

```
# plot the model
par(mfrow = c(2, 2))
plot(mod2)
```



0.0.5 Problem 2(b)

```
# Compute linear predictor
eq <- function(i, a) {
    mod2$coefficients[1] + mod2$coefficients[2] * i + mod2$coefficients[3] *
        a
}

# compute probabilities keeping age constant
p1 <- ilogit(eq(32, 3))
p2 <- ilogit(eq(33, 3))

# Compute the odds
o1 <- p1/(1 - p1)
o2 <- p2/(1 - p2)

# Compare Ratio of odds and exp(intercept)
o2/o1
```

```
## (Intercept)
##        1.07
```

```
exp(mod2$coefficients[2])
```

```
## income
##   1.07
```

```
# compute probabilities keeping age constant
p1 <- ilogit(eq(32, 3))
p2 <- ilogit(eq(32, 4))

# Compute the odds
o1 <- p1/(1 - p1)
o2 <- p2/(1 - p2)

# Compare Ratio of odds and exp(intercept)
o2/o1
```

```
## (Intercept)
##        1.82
```

```
exp(mod2$coefficients[3])
```

```
##  age
## 1.82
```

0.0.5.1 Observation

- exp(beta1): This is the coefficient for income. We can see a unit change in the income predictor, by keeping age as constant) makes the ratio of odds(income) equal to the exp(beta1).
- exp(beta2): This is the coefficient for age We can see a unit change in the age predictor, by keeping income as constant) makes the ratio of odds(age) equal to the exp(beta2).

0.0.6 Problem 2(c)

```
# prob for income=50, age=3 using coefficients
ilogit(eq(50, 3))
```

```
## (Intercept)
##       0.609
```

```
# prob for income=50, age=3 using predict method
predict(mod2, newdata = data.frame(income = 50, age = 3), type = "response")
```

```
##     1
## 0.609
```

```
pr1 <- predict(mod2, newdata = data.frame(income = 50, age = 3),
    se = T)

ilogit(c(pr1$fit - 1.96 * pr1$se.fit, pr1$fit + 1.96 * pr1$se.fit))
```

```
##     1     1
## 0.365 0.808
```

0.0.7 Problem 2(d)

```
# using likelihood approach
confint(mod2)
```

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

```
##               2.5 % 97.5 %
## (Intercept) -9.4452 -1.049
## income       0.0195  0.132
## age         -0.1222  1.447
```

```
# 95% CI for using normal approximations
c(0.0677 - 1.96 * 0.0281, 0.0677 + 1.96 * 0.0281)
```

```
## [1] 0.0126 0.1228
```

```
c(0.5986 - 1.96 * 0.3901, 0.5986 + 1.96 * 0.3901)
```

```
## [1] -0.166  1.363
```

0.0.7.1 Observations

- Income: For every unit increase in this variable, the odds of buying the car increases by approximately 1.07 with 95% certainity that coefficient is with in the values (0.0195 0.132)
- Age: For every unit increase in this variable, the odds of buying the car increases by approximately 1.82 with 95% certainity that coefficient is with in the values (-0.1222 1.447)

0.0.8 Problem 2(e)

```
deviance <- mod2$null.deviance - mod2$deviance
sprintf("Deviance of the logistic model: %.3f", deviance)
```

```
## [1] "Deviance of the logistic model: 8.298"
```

```
anova(mod2, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: y
```

```
##
## Terms added sequentially (first to last)
##
##
##        Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                     32       45.0
## income  1     5.68        31       39.3    0.017 *
## age     1     2.61        30       36.7    0.106
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

0.0.8.1 Observation

- We can see the model deviance is not very far from the null (saturated model) deviance. This suggest a good fit for our model.
- We can see less p-values for the predictors in our anova model, suggesting a good fit.

0.1 Document Information.

All of the statistical analyses in this document will be performed using R version 4.1.0 (2021-05-18). R packages used will be maintained using the packrat dependency management system.

```
sessionInfo()
```

```
## R version 4.1.0 (2021-05-18)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19041)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_United States.1252
## [2] LC_CTYPE=English_United States.1252
## [3] LC_MONETARY=English_United States.1252
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.1252
##
## attached base packages:
## [1] grid      stats     graphics  grDevices utils     datasets  methods
## [8] base
##
## other attached packages:
##  [1] Matrix_1.3-4       psych_2.1.6        leaps_3.1          faraway_1.0.7
##  [5] xtable_1.8-4       lmtest_0.9-38      zoo_1.8-9          PairedData_1.1.1
##  [9] mvtnorm_1.1-2      gld_2.6.2          ggpubr_0.4.0       car_3.0-11
## [13] carData_3.0-4      mnormt_2.0.2       vcd_1.4-8          epiDisplay_3.5.0.1
## [17] nnet_7.3-16        foreign_0.8-81     Hmisc_4.5-0        Formula_1.2-4
## [21] survival_3.2-11    lattice_0.20-44    MASS_7.3-54        ggplot2_3.3.5
## [25] rmarkdown_2.8      knitr_1.33
##
## loaded via a namespace (and not attached):
##  [1] nlme_3.1-152        RColorBrewer_1.1-2  tools_4.1.0
##  [4] backports_1.2.1     utf8_1.2.1          R6_2.5.0
##  [7] rpart_4.1-15        colorspace_2.0-1    withr_2.4.2
## [10] tidyselect_1.1.1    gridExtra_2.3       curl_4.3.1
```

```
## [13] compiler_4.1.0      formatR_1.11        htmlTable_2.2.1
## [16] scales_1.1.1        checkmate_2.0.0     proxy_0.4-26
## [19] stringr_1.4.0       digest_0.6.27       minqa_1.2.4
## [22] rio_0.5.27          base64enc_0.1-3     jpeg_0.1-8.1
## [25] pkgconfig_2.0.3     htmltools_0.5.1.1   lme4_1.1-27.1
## [28] highr_0.9           htmlwidgets_1.5.3   rlang_0.4.11
## [31] readxl_1.3.1        rstudioapi_0.13     generics_0.1.0
## [34] dplyr_1.0.7         zip_2.2.0           magrittr_2.0.1
## [37] Rcpp_1.0.6          munsell_0.5.0       fansi_0.5.0
## [40] abind_1.4-5         lifecycle_1.0.0     stringi_1.6.1
## [43] yaml_2.2.1          parallel_4.1.0      forcats_0.5.1
## [46] crayon_1.4.1        lmom_2.8            haven_2.4.1
## [49] splines_4.1.0       hms_1.1.0           tmvnsim_1.0-2
## [52] pillar_1.6.1        boot_1.3-28         ggsignif_0.6.2
## [55] glue_1.4.2          evaluate_0.14       latticeExtra_0.6-29
## [58] data.table_1.14.0   nloptr_1.2.2.2      png_0.1-7
## [61] vctrs_0.3.8         cellranger_1.1.0    gtable_0.3.0
## [64] purrr_0.3.4         tidyr_1.1.3         xfun_0.23
## [67] openxlsx_4.2.4      broom_0.7.8         e1071_1.7-7
## [70] rstatix_0.7.0       class_7.3-19        tibble_3.1.2
## [73] cluster_2.1.2       ellipsis_0.3.2
```