```
################################################################################

# Brian Weinstein - bmw2148
# STAT W4201 001
# Homework 9
# 2016-04-13

# set working directory
setwd("~/Documents/advanced-data-analysis/homework_09")

# prevent R from printing large numbers in scientific notation
options(scipen=5)

# load packages
library(Sleuth3) # Data sets from Ramsey and Schafer's "Statistical Sleuth
(3rd ed)"
library(ggplot2); theme_set(theme_bw())
library(dplyr)



# Problem 1: Ramsey 20.12
#######################################################################

# load data
mdData <- Sleuth3::ex2012
mdData$Group <- relevel(mdData$Group, ref = "Control")

# Part a ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ###

# scatterplot
ggplot(mdData, aes(x=log(CK), y=H)) +
  geom_point(aes(color=Group, shape=Group), size=2)
ggsave(filename="writeup/1a.png", width=6.125, height=3.5, units="in")

# Part b ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ###

# fit a logistic regression model on CK and CK^2
glm_1b1 <- glm(formula = Group ~ CK + I(CK^2),
               data = mdData, family = binomial)
summary(glm_1b1)$coefficients

# fit a logistic regression model on log(CK) and log(CK)^2
glm_1b2 <- glm(formula = Group ~ log(CK) + I(log(CK)^2),
               data = mdData, family = binomial)
summary(glm_1b2)$coefficients

# scatterplot
ggplot(mdData, aes(x=CK, y=H)) +
  geom_point(aes(color=Group, shape=Group), size=2)
ggsave(filename="writeup/1b.png", width=6.125, height=3.5, units="in")

# Part c ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ###
# fit a logistic regression model on log(CK) and H
glm_1c <- glm(formula = Group ~ log(CK) + H,
              data = mdData, family = binomial)
summary(glm_1c)$coefficients

# Part d ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ###

# fit a reduced model
glm_1d <- glm(formula = Group ~ 1,
              data = mdData, family = binomial)
summary(glm_1d)$coefficients

# perform the likelihood ratio test (drop-in-deviance test)
anova(glm_1c, glm_1d, test="LRT")

# Part e ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ### ###

# calculate odds and probability of having DMD at CK=80, H=85
odds1 <- exp(predict(glm_1c, data.frame(CK=80, H=85)))[[1]] ; odds1
1 / (1 + exp(-odds1))

# calculate odds and probability of having DMD at CK=300, H=100
odds2 <- exp(predict(glm_1c, data.frame(CK=300, H=100)))[[1]] ; odds2
1 / (1 + exp(-odds2))

# calculate the odds ratio
odds2/odds1

rm(list = ls()) # clear working environment



# Problem 2: Ramsey 21.16
#######################################################################

# load data and create a tumor proportion variable
troutData <- Sleuth3::ex2116 %>%
  mutate(TumorProp=Tumor/Total)

# scatterplot
set.seed(1)
ggplot(troutData, aes(x=log(Dose), y=log(TumorProp/(1-TumorProp)))) +
  geom_jitter(size=2, width=0.05)
ggsave(filename="writeup/2_scatter.png", width=6.125, height=3.5, units="in")

# fit a binomial counts logistic regression model on a rich model
glm2 <- glm(formula = TumorProp ~ log(Dose) + I(log(Dose)^2),
            data = troutData, family = binomial, weights = Total)
summary(glm2)

# compute the goodness of fit p value
pchisq(q = summary(glm2)$deviance, df = summary(glm2)$df.residual,
       lower.tail = FALSE)
```

```
# examine deviance residuals
qplot(x=glm2$fitted.values, y=summary(glm2)$deviance.resid) +
  geom_point(size=2) +
  geom_hline(yintercept = c(-2, 2), linetype="dashed", color="gray")
ggsave(filename="writeup/2_devresid.png", width=6.125, height=3.5, units="in")

# estimate the dispersion parameter
dispersion_param <- summary(glm2)$deviance / summary(glm2)$df.residual
dispersion_param
sqrt(dispersion_param)

# compute the quasi-likelihood standard errors, t-statistics, and pvalues
glm2QuasiSummary <- data.frame(summary(glm2)$coefficients) %>%
  mutate(Term=row.names(.)) %>%
  select(Term, Estimate, ML_StdError=Std..Error, ML_ZValue=z.value,
ML_PValue=Pr...z..) %>%
  mutate(QL_StdError=ML_StdError * sqrt(dispersion_param),
         QL_TValue=Estimate/QL_StdError)
glm2QuasiSummary$QL_PValue <- 2 * pt(q = -1 * abs(glm2QuasiSummary$QL_TValue),
                                     df =
as.integer(summary(glm2)$df.residual))
glm2QuasiSummary[, -1] <- round(glm2QuasiSummary[, -1], 5)
glm2QuasiSummary

# final model
glm2QuasiSummary %>%
  select(Term, Estimate, QL_StdError, QL_TValue, QL_PValue)

# solved numerically in mathematica that Dose=0.0333345:
# verify that ~50% of fish are expected to get tumors at Dose=0.0333345
1/(1+exp(-1 * predict(glm2, data.frame(Dose=0.0333345))))

rm(list = ls()) # clear working environment



# Problem 3
######################################################################

# no code needed
```