

Compulsory exercise 1: Group 37

TMA4268 Statistical Learning V2022

Oskar Jørgensen, Halvor Linder Henriksen

18 februar, 2022

Problem 1

a)

$$\begin{aligned} E[(y_0 - \hat{f}(x_0))^2] &= E[y_0^2 - 2y_0\hat{f}(x_0) + \hat{f}(x_0)^2] \\ &= E[y_0^2] - 2E[y_0\hat{f}(x_0)] + E[\hat{f}(x_0)^2] \\ &= E[y_0^2] + \text{Var}[y_0] - 2E[y_0\hat{f}(x_0)] + E[\hat{f}(x_0)^2] + \text{Var}[\hat{f}(x_0)] \\ &= E[f(x_0) + \epsilon]^2 + \text{Var}[f(x_0) + \epsilon] - 2E[(f(x_0) + \epsilon)\hat{f}(x_0)] + E[\hat{f}(x_0)^2] + \text{Var}[\hat{f}(x_0)] \end{aligned}$$

It is assumed that ϵ is independent of x , and that $E[\epsilon] = 0$

$$\begin{aligned} &= E[f(x_0) + \epsilon]^2 + \text{Var}[f(x_0) + \epsilon] - 2E[\epsilon\hat{f}(x_0)] - 2E[f(x_0)\hat{f}(x_0)] + E[\hat{f}(x_0)^2] + \text{Var}[\hat{f}(x_0)] \\ &= E[f(x_0)^2] + \text{Var}[f(x_0)] + \text{Var}[\epsilon] - 2E[\epsilon\hat{f}(x_0)] - 2E[f(x_0)\hat{f}(x_0)] + E[\hat{f}(x_0)^2] + \text{Var}[\hat{f}(x_0)] \\ &= E[f(x_0)^2] - 2E[f(x_0)\hat{f}(x_0)] + E[\hat{f}(x_0)^2] + \text{Var}[\hat{f}(x_0)] + \text{Var}[\epsilon] \\ &= f(x_0)^2 - 2f(x_0)E[\hat{f}(x_0)] + E[\hat{f}(x_0)^2] + \text{Var}[\hat{f}(x_0)] + \text{Var}[\epsilon] \\ &= (f(x_0) - E[\hat{f}(x_0)])^2 + \text{Var}[\hat{f}(x_0)] + \text{Var}[\epsilon] \end{aligned}$$

Where:

- $\text{Var}[\epsilon]$ is the irreducible error
- $\text{Var}[\hat{f}(x_0)]$ is the variance of the prediction
- $(f(x_0) - E[\hat{f}(x_0)])^2$ is the bias of the prediction

b)

Irreducible error:

- The irreducible error is the error made by the true function f , implying that the irreducible error will always exist no matter how well the model is fit. It may stem from factors that are not captured by the covariates or an inherent stochasticity in the response variable.

Reducible error:

- The reducible error is caused by the discrepancy between the fitted model and the true relationship, and it is the sum of the square of the bias, and the variance
- Variance:

- The variance is typically caused by overfitting the model to the training data. This implies that a new set of training could result in a vastly different model.
- Bias:
 - The bias is typically caused by underfitting the model. A high bias is usually found in an inflexible model that fails to capture nuances in the relationship.

c)

1. True
2. False
3. True
4. False

d)

1. True
2. False
3. False
4. False

e)

(iii) 0.76

Problem 2

```
library(palmerpenguins) # Contains the data set "penguins".
data(penguins)
head(penguins)
```

```
## # A tibble: 6 x 8
##   species island bill_length_mm bill_depth_mm flipper_length_~ body_mass_g sex
##   <fct>   <fct>         <dbl>         <dbl>         <int>         <int> <fct>
## 1 Adelie  Torge~           39.1           18.7           181           3750 male
## 2 Adelie  Torge~           39.5           17.4           186           3800 fema~
## 3 Adelie  Torge~           40.3            18           195           3250 fema~
## 4 Adelie  Torge~           NA             NA             NA             NA <NA>
## 5 Adelie  Torge~           36.7           19.3           193           3450 fema~
## 6 Adelie  Torge~           39.3           20.6           190           3650 male
## # ... with 1 more variable: year <int>
```

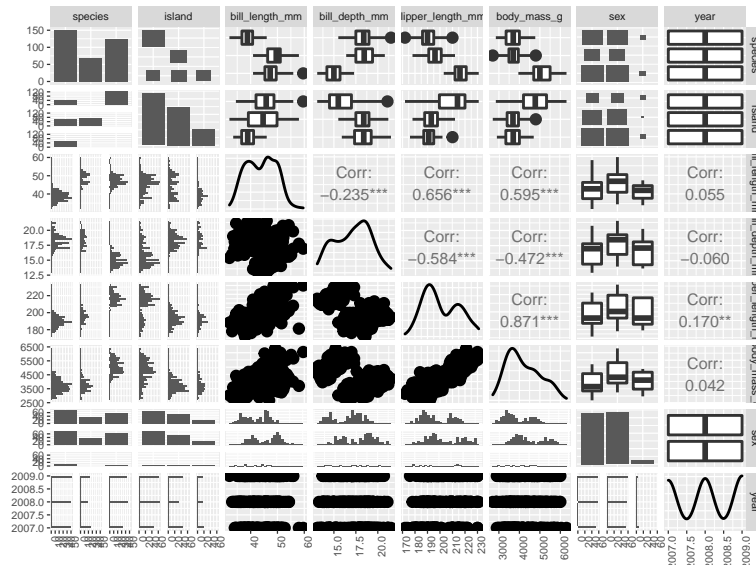
a)

- Error 1: The sex covariate is rejected on the basis of a low p-value. This is wrong because a low p-value indicates that the probability of observing a more extreme value given the null hypothesis is low. In this case, the p-value is way too low to be ignored.
- Error 2: The null hypothesis is tested separately for the categorical covariate. This is incorrect when the covariate has more than two categories, because the dummy variables truly represent the same covariate. Hence, the F-test must be applied in order to evaluate the significance of the categorical covariate, as it can test multiple variables at the same time. The basis on which he kept the species covariate is therefore invalid.

- Error 3: The assumption that the Chinstrap penguins are the largest is possibly wrong, as the interaction term between bill depth and species must also be taken into account.

b)

```
library(GGally)
ggpairs(penguins, out.height="500px", out.width="400px", fig.height=11, fig.width=10, upper=list(continuous=
```



The pairs functions helps with identifying correlations in the data. In this case, the boxplot between sex and body weight indicates a relationship.

c)

```
##### =^._.^= ~~~Oskar and Halvor's CODE~~~ =^._.^= #####
##### install.packages('palmerpenguins') # Run if you haven't installed before.
library(ggfortify)
library(palmerpenguins) # Contains the data set 'penguins'.
data(penguins)
# We do not discard island, as the pairs plot suggests a correlation with body mass.
Penguins <- subset(penguins, select = -c(year))

# Fit the model as specified in advance based on expert knowledge
# with the inclusion of island:
penguin.model <- lm(body_mass_g ~ flipper_length_mm + sex + bill_depth_mm * species + island,
  data = Penguins)

# Look at the model coefficients
summary(penguin.model)$coefficients
```

| | Estimate | Std. Error | t value | Pr(> t) |
|----------------------|-------------|------------|------------|--------------|
| ## (Intercept) | -1385.27845 | 652.738847 | -2.1222552 | 3.457699e-02 |
| ## flipper_length_mm | 17.71249 | 2.941999 | 6.0205607 | 4.721257e-09 |
| ## sexmale | 430.56261 | 44.848586 | 9.6003609 | 2.304069e-19 |
| ## bill_depth_mm | 83.59180 | 22.449873 | 3.7234864 | 2.317818e-04 |
| ## speciesChinstrap | 1474.29960 | 681.997594 | 2.1617372 | 3.137259e-02 |

```

## speciesGentoo          632.57985 545.520709 1.1595891 2.470726e-01
## islandDream            -21.35204 58.304464 -0.3662163 7.144434e-01
## islandTorgersen        -51.77579 60.839901 -0.8510170 3.953904e-01
## bill_depth_mm:speciesChinstrap -84.58980 37.117784 -2.2789561 2.332189e-02
## bill_depth_mm:speciesGentoo    34.87807 34.595615 1.0081645 3.141301e-01
anova(penguin.model)

## Analysis of Variance Table
##
## Response: body_mass_g
##              Df      Sum Sq   Mean Sq    F value    Pr(>F)
## flipper_length_mm 1 164047703 164047703 1986.9804 < 2.2e-16 ***
## sex                1   9416589   9416589  114.0557 < 2.2e-16 ***
## bill_depth_mm      1   3667377   3667377   44.4200 1.145e-10 ***
## species            2  10670525   5335262   64.6218 < 2.2e-16 ***
## island             2    59488    29744    0.3603 0.69777
## bill_depth_mm:species 2    730681   365341    4.4251 0.01271 *
## Residuals         323 26667303    82561
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# We see from the f test that island is likely not important after all.
# We do, however, see that the interaction term and species are indeed significant.

# Fit final model with sex
final.model <- lm(body_mass_g ~ flipper_length_mm + bill_depth_mm * species + sex, data = Penguins)

summary(final.model)

##
## Call:
## lm(formula = body_mass_g ~ flipper_length_mm + bill_depth_mm *
##     species + sex, data = Penguins)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -751.2  -183.8   -9.8   191.1   906.9
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -1336.58     646.92  -2.066 0.039615 *
## flipper_length_mm      17.38       2.91   5.971 6.17e-09 ***
## bill_depth_mm       82.98       22.32   3.717 0.000237 ***
## speciesChinstrap   1460.15     680.39   2.146 0.032610 *
## speciesGentoo      644.88     542.57   1.189 0.235481
## sexmale           432.90      44.63   9.699 < 2e-16 ***
## bill_depth_mm:speciesChinstrap  -83.53      37.01  -2.257 0.024666 *
## bill_depth_mm:speciesGentoo     36.17      34.48   1.049 0.294955
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 286.8 on 325 degrees of freedom
## (11 observations deleted due to missingness)
## Multiple R-squared:  0.8758, Adjusted R-squared:  0.8732

```

```
## F-statistic: 327.5 on 7 and 325 DF, p-value: < 2.2e-16
```

```
anova(final.model)
```

```
## Analysis of Variance Table
```

```
##
```

```
## Response: body_mass_g
```

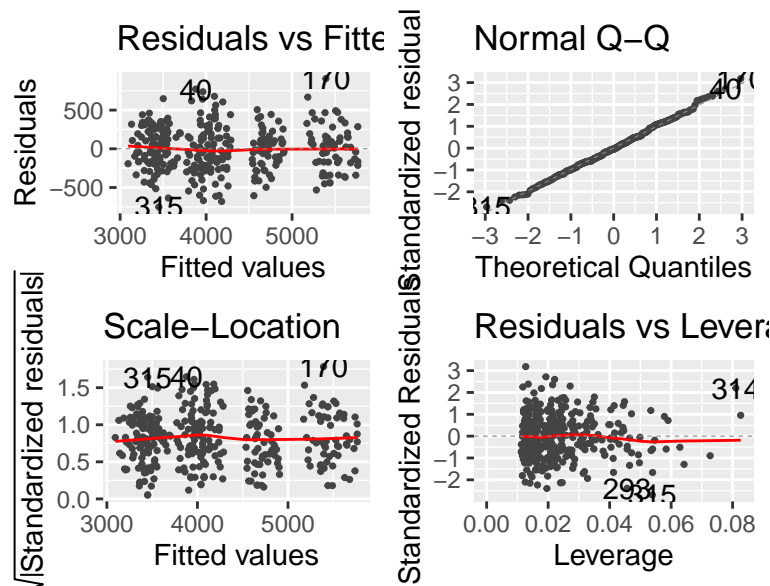
```
##
```

| | Df | Sum Sq | Mean Sq | F value | Pr(>F) |
|-----------------------|-----|-----------|-----------|-----------|-------------|
| flipper_length_mm | 1 | 164047703 | 164047703 | 1994.7424 | < 2e-16 *** |
| bill_depth_mm | 1 | 338887 | 338887 | 4.1207 | 0.04318 * |
| species | 2 | 15483131 | 7741565 | 94.1338 | < 2e-16 *** |
| sex | 1 | 7932472 | 7932472 | 96.4551 | < 2e-16 *** |
| bill_depth_mm:species | 2 | 729458 | 364729 | 4.4349 | 0.01258 * |
| Residuals | 325 | 26728014 | 82240 | | |

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
autoplot(final.model, smooth.colour="red", size=0.6)
```



REPORT: PREDICTION OF PENGUIN BODY MASS, by Oskar and Halvor :3 We begin with a linear regression model with body mass as the response, and flipper length, bill depth, species, island and sex as covariates, as well as an interaction effect between bill depth and species. We see that the island term has a large p-value and it is therefore discarded, as it does not seem to be correlated with body mass. From the f-test we see that both the categorical variables are probably significant. The final model can be described depending on the species of the penguin:

$$\begin{aligned}
 \hat{y}_{adelie} &= \hat{\beta}_0 + \hat{\beta}_{flipper_length} x_{flipper_length} + \hat{\beta}_{sex_male} x_{sex_male} \\
 &\quad + \hat{\beta}_{bill_depth} x_{bill_depth} \\
 \hat{y}_{chinstrap} &= \hat{\beta}_0 + \hat{\beta}_{flipper_length} x_{flipper_length} + \hat{\beta}_{sex_male} x_{sex_male} \\
 &\quad + (\hat{\beta}_{bill_depth} + \hat{\beta}_{bill_depth:chinstrap}) x_{bill_depth} + \hat{\beta}_{chinstrap} \\
 \hat{y}_{gentoo} &= \hat{\beta}_0 + \hat{\beta}_{flipper_length} x_{flipper_length} + \hat{\beta}_{sex_male} x_{sex_male} \\
 &\quad + (\hat{\beta}_{bill_depth} + \hat{\beta}_{bill_depth:gentoo}) x_{bill_depth} + \hat{\beta}_{gentoo}
 \end{aligned}$$

(where \hat{y}_{adelie} is the predicted body mass for Adelie penguins, $\hat{\beta}_0$ is the estimated intercept, $x_{flipper_length}$ is the flipper length covariate, $\hat{\beta}_{flipper_length}$ is the estimated flipper length coefficient, etc.) Note that

x_{sex_male} is 1 if the penguin is male and 0 if female. In the final model, all terms have small (<0.05) f-values, signaling a strong correlation with the response. The assumptions of a linear model can be checked with the four plots supplied by the `autoplot()` function. From the qq-plot we see strong evidence that the residuals are normally distributed. From the Tukey-Anscombe diagram we see evidence of the residuals having expectation 0 and equal σ^2 . This is also backed up by the scale location plot. There is also no apparent pattern in the T-A diagram, suggesting that the residuals are independent. The leverage plot is useful for determining outliers with high leverage. There is one data point (314) in particular that warrants further inspection, as it has quite high leverage and might be an outlier.

Problem 3

a)

```
library(tidyverse)
library(GGally)
# Create a new boolean variable indicating whether or not the penguin is an
# Adelie penguin
Penguins$adelie <- ifelse(Penguins$species == "Adelie", 1, 0)
# Select only relevant variables and remove all rows with missing values in body
# mass, flipper length, sex or species.
Penguins_reduced <- Penguins %>% dplyr::select(body_mass_g, flipper_length_mm, adelie) %>%
mutate(body_mass_g = as.numeric(body_mass_g),
       flipper_length_mm = as.numeric(flipper_length_mm)) %>% drop_na()
set.seed(4268)
# 70% of the sample size for training set
training_set_size <- floor(0.7 * nrow(Penguins_reduced))
train_ind <- sample(seq_len(nrow(Penguins_reduced)), size = training_set_size)
train <- Penguins_reduced[train_ind, ]
test <- Penguins_reduced[-train_ind, ]
```

(i)

```
penguin.glm = glm(adelie ~ ., data=train, family="binomial")
penguin.glm.probs = predict(penguin.glm, type="response", newdata=test)
penguin.glm.preds = ifelse(penguin.glm.probs > 0.5, 1, 0)
conf.glm = table(penguin.glm.preds, test$adelie)
```

(ii)

```
library(MASS)
penguin.qda = qda(adelie ~ ., data=train)
penguin.qda.probs = predict(penguin.qda, newdata=test)$posterior
penguin.qda.preds = predict(penguin.qda, newdata=test)$class
conf.qda = table(penguin.qda.preds, test$adelie)
```

(iii)

```
library(class)
penguin.knn = knn(train = train, test = test, cl = train$adelie, k=25, prob=T)
penguin.knn.probs = attributes(penguin.knn)$prob
not.adelie = which(penguin.knn == 0)
```

```
penguin.knn.probs[not.adelie] = 1-penguin.knn.probs[not.adelie]
conf.knn = table(penguin.knn, test$adelie)
```

(iv)

```
#True positive rate
sensitivity = function(table){
  return (table[2,2]/(table[2,2]+table[1,2]))
}
#True negative rate
specificity = function(table){
  return (table[1,1]/(table[1,1]+table[2,1]))
}

cat("Sensitivity for logistic regression:", sensitivity(conf.glm), "\n")

## Sensitivity for logistic regression: 0.9767442
cat("Specificity for logistic regression:", specificity(conf.glm), "\n")

## Specificity for logistic regression: 0.8666667
cat("Sensitivity for QDA:", sensitivity(conf.qda), "\n")

## Sensitivity for QDA: 0.9767442
cat("Specificity for QDA:", specificity(conf.qda), "\n")

## Specificity for QDA: 0.7666667
cat("Sensitivity for KNN:", sensitivity(conf.knn), "\n")

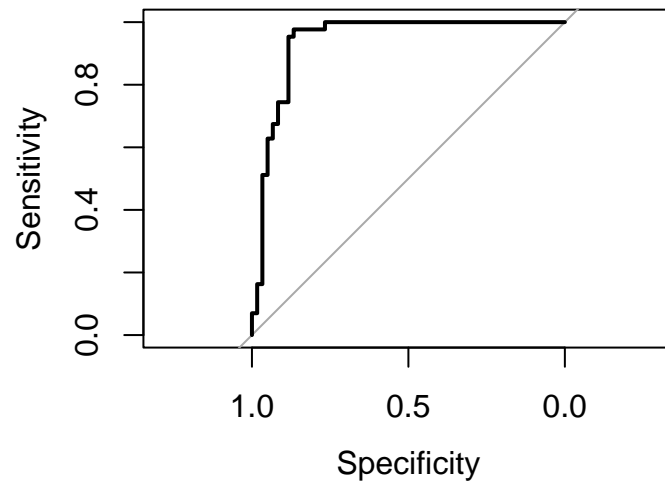
## Sensitivity for KNN: 0.9534884
cat("Specificity for KNN:", specificity(conf.knn), "\n")

## Specificity for KNN: 0.5833333
```

b)

(i)

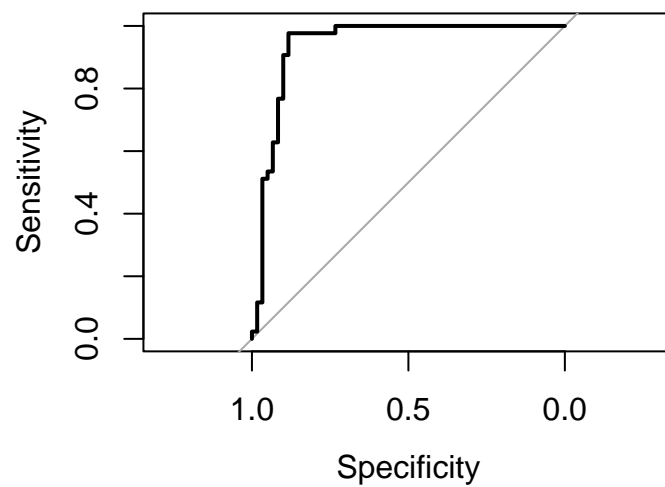
```
library(pROC)
library(plotROC)
roc.glm = roc(response = test$adelie, predictor = penguin.glm.probs, direction = "<")
plot(roc.glm)
```



```
auc(roc.glm)
```

```
## Area under the curve: 0.9391
```

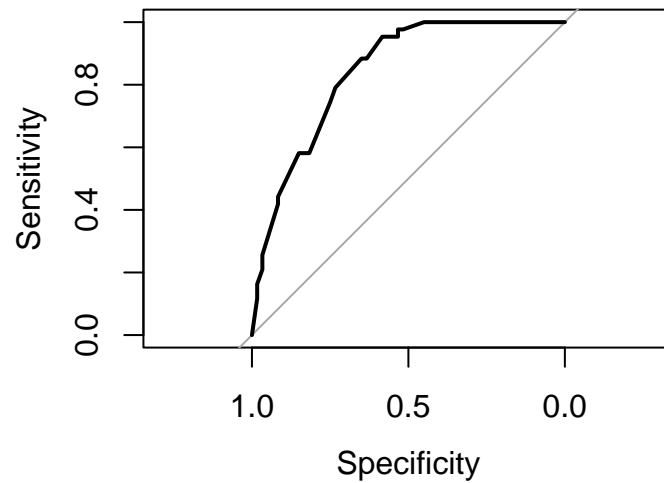
```
roc.qda = roc(response = test$adelie, predictor = penguin.qda.probs[,2], direction = "<")
plot(roc.qda)
```



```
auc(roc.qda)
```

```
## Area under the curve: 0.938
```

```
roc.knn = roc(response = test$adelie, predictor = penguin.knn.probs, direction = "<")
plot(roc.knn)
```

```
auc(roc.knn)
```

```
## Area under the curve: 0.8417
```

(ii)

From the ROC plots it seems like the logistic model and the QDA perform the best. This is backed up by their respective AUC values (0.9391 and 0.938). The KNN model, however, only has an AUC value of 0.8417, implying that it performs quite a bit worse when predicting the test data.

(iii)

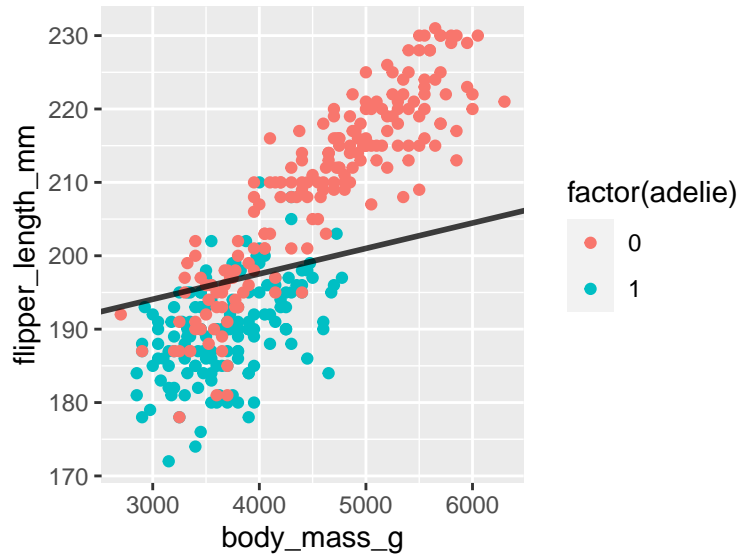
Since $p = 2$ we can easily visualize the three different models, which helps with interpretability. The logistic regression might be the easiest to interpret, as each of its weights correlate to the magnitude a change to the corresponding predictor has on the response value. As a result of this, increasing a variable will always increase or decrease the response value, depending on the sign of the weight (unlike QDA or KNN, where increasing the value of a predictor might increase the response value in some interval, while decreasing it in another interval).

c)

(iii) We multiply by 2.038

d)

```
penguin.glm.plot = ggplot(Penguins_reduced, aes(x=body_mass_g, y=flipper_length_mm, color=factor(adelie)))
penguin.glm.plot
```



The plot shows the true species (the colored dots) and our prediction boundary. All dots that fall below the boundary would be classified as 1 (adelie), while dots above the boundary would be classified as 0 (not adelie) by our logistic regression model.

The formula directly follows from the definition of the logistic model:

$$\ln \frac{p}{1-p} = \beta_0 + \beta_1 x_1 + \beta_2 x_2$$

Reordering the terms, we can express the relation between x_1 and x_2 linearly:

$$x_2 = \frac{\ln \frac{p}{1-p} - \beta_0}{\beta_2} - \frac{\beta_1 x_1}{\beta_2}$$

We previously set the boundary at $p = 1/2$. Inserting this gives:

$$x_2 = -\frac{\beta_0}{\beta_2} - \frac{\beta_1 x_1}{\beta_2}$$

Problem 4

a)

1. True
2. False
3. False
4. False

b)

```
id <- "1chRpybM5cJn4Eow3-_xwDKPKyddL9M2N" # google file ID
d.chd <- read.csv(sprintf("https://docs.google.com/uc?id=%s&export=download", id))
# head(d.chd)
chd.glm = glm(chd~sbp + sex + smoking ,data = d.chd, family = "binomial")
summary(chd.glm)
```

```
##
## Call:
## glm(formula = chd ~ sbp + sex + smoking, family = "binomial",
##      data = d.chd)
##
```

```
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.0184  -0.5950  -0.3790  -0.2954   2.5570
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.65884    2.36740  -2.813  0.00491 **
## sbp          0.03877    0.01794   2.162  0.03066 *
## sex         -1.34351    0.32148  -4.179 2.93e-05 ***
## smoking      0.41031    0.31014   1.323  0.18584
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 370.89  on 499  degrees of freedom
## Residual deviance: 342.91  on 496  degrees of freedom
## AIC: 350.91
##
## Number of Fisher Scoring iterations: 5
chance1 = predict(chd.glm, type = "response", newdata = data.frame(sbp=150, sex=1, smoking=0))
cat("Chance of chd for non-smoking male with sbp 150: ", chance1 * 100, "%", sep = "")

## Chance of chd for non-smoking male with sbp 150: 10.096%
```

c)

```
library(boot)
set.seed(123)
boot.fn <- function(data,index){
  chd.glm = glm(chd ~ sbp + sex + smoking, family="binomial",data=data,subset=index)
  return(predict(chd.glm, type = "response", newdata = data.frame(sbp=150, sex=1, smoking=0)))
}

chd.boot = boot(data = d.chd, statistic = boot.fn, R = 1000)
summary(chd.boot$t)

##           V1
##      Min.   :0.01499
##    1st Qu.:0.07633
##     Median :0.09997
##      Mean  :0.10584
##    3rd Qu.:0.12867
##     Max.   :0.31140

# The expected value is 0.10329, or 10.329%
chd.boot.ci = boot.ci(chd.boot)
print(chd.boot.ci)

## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 1000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = chd.boot)
```

```
##
## Intervals :
## Level      Normal      Basic
## 95%   ( 0.0135, 0.1787 ) (-0.0021, 0.1625 )
##
## Level      Percentile      BCa
## 95%   ( 0.0395, 0.2040 ) ( 0.0432, 0.2100 )
## Calculations and Intervals on Original Scale
```

The plausible values (within the 95 percentile interval) range from 0.0388 to 0.2019.

d)

1. False
2. False
3. False
4. True