Answer for 100 Math Questions for Machine Learning

Shota Komatsu

December 04, 2019

Contents

1. Linear Regression

Q 1

Let:

$$Q(\beta_0, \beta_1) := \sum_{i=1}^{N} (y_i - \beta_0 - \beta_1 x_i)^2.$$

Taking the first order conditions, we obtain:

$$\begin{cases} 0 = \frac{\partial Q}{\partial \beta_0} = -2 \sum_{i=1}^{N} (y_i - \beta_0 - \beta_1 x_i) \\ 0 = \frac{\partial Q}{\partial \beta_1} = -2 \sum_{i=1}^{N} x_i (y_i - \beta_0 - \beta_1 x_i) \end{cases}$$
$$\therefore \begin{cases} \sum_{i=1}^{N} (y_i - \beta_0 - \beta_1 x_i) = 0 \\ \sum_{i=1}^{N} x_i (y_i - \beta_0 - \beta_1 x_i) = 0 \end{cases} .$$

From the first equation,

$$\sum_{i=1}^{N} y_i - N\beta_0 - \beta_1 \sum_{i=1}^{N} x_i = 0$$
$$\frac{1}{N} \sum_{i=1}^{N} y_i - \beta_0 - \beta_1 \frac{1}{N} \sum_{i=1}^{N} x_i = 0$$
$$\therefore \beta_0 = \bar{y} - \beta_1 \bar{x}.$$

Insert this into the second equation in the first conditions,

$$\sum_{i=1}^{N} x_i (y_i - \bar{y} + \beta_1 \bar{x} - \beta_1 x_i) = 0$$
$$\sum_{i=1}^{N} x_i y_i - \bar{y} \sum_{i=1}^{N} x_i + \beta_1 \bar{x} \sum_{i=1}^{N} x_i - \beta_1 \sum_{i=1}^{N} x_i^2 = 0$$
$$\left(\sum_{i=1}^{N} x_i^2 - \bar{x} \sum_{i=1}^{N} x_i \right) \beta_1 = \sum_{i=1}^{N} x_i y_i - \bar{y} \sum_{i=1}^{N} x_i.$$

Here,

$$\begin{aligned}
\sum_{i=1}^{N} x_i^2 - \bar{x}\sum_{i=1}^{N} x_i &= \sum_{i=1}^{N} x_i^2 - 2\bar{x}\sum_{i=1}^{N} x_i + \bar{x}\sum_{i=1}^{N} x_i \\
&= \sum_{i=1}^{N} x_i^2 - 2\bar{x}\sum_{i=1}^{N} x_i + N\bar{x}\frac{1}{N}\sum_{i=1}^{N} x_i \\
&= \sum_{i=1}^{N} x_i^2 - 2\bar{x}\sum_{i=1}^{N} x_i + N\bar{x}^2 \\
&= \sum_{i=1}^{N} (x_i^2 - 2\bar{x}x_i + \bar{x}^2) \\
&= \sum_{i=1}^{N} (x_i - \bar{x})^2. \\
\therefore \sum_{i=1}^{N} x_i^2 - \bar{x}\sum_{i=1}^{N} x_i &= \sum_{i=1}^{N} (x_i - \bar{x})^2.
\end{aligned}$$

Analogously, we can show that

$$\sum_{i=1}^{N} x_i y_i - \bar{y}\sum_{i=1}^{N} x_i = \sum_{i=1}^{N} (x_i - \bar{x})(y_i - \bar{y}).$$

Therefore it follows that

$$\begin{aligned}
\left(\sum_{i=1}^{N} (x_i - \bar{x})^2\right)\beta_1 &= \sum_{i=1}^{N} (x_i - \bar{x})(y_i - \bar{y}) \\
\therefore \beta_1 &= \frac{\sum_{i=1}^{N} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{N} (x_i - \bar{x})^2}
\end{aligned}$$

Thus

$$\hat{\beta}_0 = \bar{y} - \beta_1\bar{x}, \quad \hat{\beta}_1 = \frac{\sum_{i=1}^{N} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{N} (x_i - \bar{x})^2}.$$

Since the function Q is convex, $\left(\hat{\beta}_0, \hat{\beta}_1\right)$ is the minimizer of Q.

Q 2

Let the intercept and the slope of l' be $\tilde{\beta}_0$ and $\tilde{\beta}_1$ respectively. The equation of l' is

$$y = \tilde{\beta}_0 + \tilde{\beta}_1 x.$$

Since l passes through $(x_i - \bar{x}, y_i - \bar{y}) \quad (i = 1, \cdots, N)$,

$$y_i - \bar{y} = \tilde{\beta}_0 + \tilde{\beta}_1(x_i - \bar{x}) \quad (i = 1, \cdots, N).$$

By summing up for $i = 1, \cdots, N$,

$$\sum_{i=1}^{N} y_i - N\bar{y} = N\tilde{\beta}_0 + \tilde{\beta}_1 \left(\sum_{i=1}^{N} x_i - N\bar{x} \right)$$
$$\sum_{i=1}^{N} y_i - \sum_{i=1}^{N} y_i = N\tilde{\beta}_0 + \tilde{\beta}_1 \left(\sum_{i=1}^{N} x_i - \sum_{i=1}^{N} x_i \right)$$
$$\therefore \tilde{\beta}_0 = 0.$$

Then we get

$$y_i - \bar{y} = \tilde{\beta}_1 (x_i - \bar{x}) \quad (i = 1, \cdots, N).$$

By multiplying $(x_i - \bar{x})$ on both sides,

$$(y_i - \bar{y})(x_i - \bar{x}) = \tilde{\beta}_1 (x_i - \bar{x})^2 \quad (i = 1, \cdots, N).$$

Summing up for $i = 1, \cdots, N$,

$$\sum_{i=1}^{N} (y_i - \bar{y})(x_i - \bar{x}) = \tilde{\beta}_1 \left(\sum_{i=1}^{N} (x_i - \bar{x})^2 \right)$$
$$\therefore \tilde{\beta}_1 = \frac{\sum_{i=1}^{N} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{N} (x_i - \bar{x})^2}.$$

Thus

$$\tilde{\beta}_0 = 0, \quad \tilde{\beta}_1 = \frac{\sum_{i=1}^{N} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{N} (x_i - \bar{x})^2}.$$

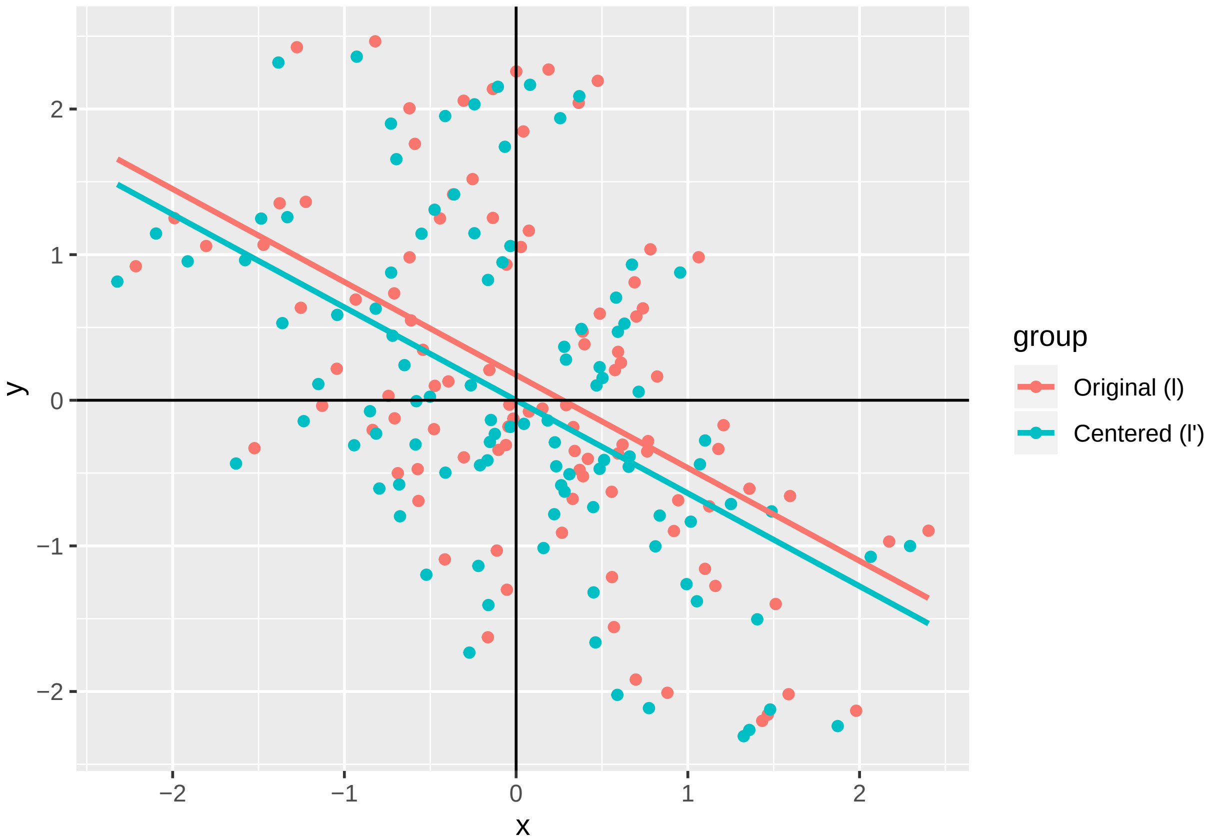
After obtaining $\hat{\beta}_1$, we can get $\hat{\beta}_0$ from

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}.$$

Q 3

```r
# Set seed
set.seed(1)
# Number of observations
N <- 100
# Intercept and slope
a <- rnorm(1)
b <- rnorm(1)
# Data points
x <- rnorm(N)
y <- a * x + b + rnorm(N)
# Mean centering
x_center <- x - mean(x)
y_center <- y - mean(y)
# Dataframe
data_3 <- data.frame(group = "Original (1)", x, y) %>%
  dplyr::bind_rows(
    data.frame(group = "Centered (1')", x = x_center, y = y_center)
  ) %>%
```

```r
  dplyr::mutate(
    group = forcats::fct_relevel(group, "Original (l)", "Centered (l')")
  )
# Graph
g_3 <- ggplot(data = data_3, aes(x = x, y = y, color = group)) +
  geom_point() +
  geom_smooth(method = lm, se = FALSE, fullrange = TRUE) +
  geom_vline(xintercept = 0, linetype = "solid") +
  geom_hline(yintercept = 0, linetype = "solid")
# Plot the graph
plot(g_3)
```



Q 4

(a)

For any $x \in \mathbb{R}^m$,

$$\begin{aligned} x^\top A x &= x^\top B^\top B x \\ &= (Bx)^\top B x \\ &= \|Bx\|^2 \\ &\geq 0. \end{aligned}$$

Thus A is positive semi-definite.

(b)

Let $\Lambda := \mathrm{diag}(\lambda_1, \cdots, \lambda_m) \in \mathbb{R}^{m\times m}$ and $\sqrt{\Lambda} := \mathrm{diag}(\sqrt{\lambda_1}, \cdots, \sqrt{\lambda_m}) \in \mathbb{R}^{m\times m}$. For any $x \in \mathbb{R}$,

$$\begin{aligned} x^\top A x &= x^\top P^\top \Lambda P x \\ &= y^\top \sqrt{\Lambda}^\top \sqrt{\Lambda} y \\ &= \left\| \sqrt{\Lambda} y \right\|^2 \\ &= \sum_{i=1}^m \sqrt{\lambda_i}^2 y_i^2 \\ &= \sum_{i=1}^m \lambda_i y_i^2. \end{aligned}$$

We will prove $\lambda_i \geq 0$ for any $i \in \{1, \cdots, m\}$ by proof by contradiction.

Suppose there exists $i \in \{1, \cdots, m\}$ such that $\lambda_i < 0$. Take $x = P^{-1} y$ such that $y_i = 1$ and $y_j = 0 \quad (j \neq i)$. Then

$$\begin{aligned} \|Bx\|^2 &= x^\top B^\top B x \\ &= x^\top A x \\ &= \sum_{k=1}^m \lambda_k y_k^2 \quad \text{(From the result above)} \\ &= \lambda_i \\ &< 0. \end{aligned}$$

However, this contraticts to the fact that $\|Bx\|^2 \geq 0$, which completes the proof.

(c)

Take any $z \in \mathbb{R}^m$.

Suppose that $Az = 0$. Then

$$\begin{aligned} Az = 0 &\Leftrightarrow B^\top B z = 0 \\ &\Rightarrow z^\top B^\top B z = 0 \\ &\Rightarrow \|Bz\|^2 = 0 \\ &\Leftrightarrow Bz = 0 \\ \therefore Az = 0 &\Rightarrow Bz = 0. \end{aligned}$$

Suppose that $Bz = 0$. Then

$$\begin{aligned} Bz = 0 &\Rightarrow B^\top B z = 0 \\ &\Leftrightarrow Az = 0 \\ \therefore Bz = 0 &\Rightarrow Az = 0. \end{aligned}$$

Therefore

$$Az = 0 \Leftrightarrow Bz = 0.$$

(d)

Suppose A is non-singular. Then

$$\begin{aligned} m = \operatorname{rank}(A) = \operatorname{rank}(B^\top B) \le \operatorname{rank}(B) = \min(m,n). \\ \therefore m \le \min(m,n). \\ \therefore \min(m,n) = m. \\ \therefore m \le n. \end{aligned}$$

Then $\operatorname{rank}(B) = m$.

Suppose $m \le n$ and $\operatorname{rank}(B) = m$. Then we have $\dim(\ker(B)) = 0$. Note that $\dim(\operatorname{Im}(A)) + \dim(\ker(A)) = m$ and that $\dim(\ker(A)) = \dim(\ker(B)) = 0$ from (c). Then we have $\dim(\operatorname{Im}(A)) = m$. Therefore, A is non-singular.

Thus we obtain

$$A \text{ is non-singular} \Leftrightarrow \operatorname{rank}(B) = m \text{ and } m \le n.$$

Q 5

(a) When $N < p+1$:

We prove this by proof by contradiction. Suppose that $X^\top X$ is invertible. Then $\operatorname{rank}(X^\top X) = p+1$. From the result from Q 4,

$$\begin{aligned} p+1 = \operatorname{rank}(X^\top X) \le \operatorname{rank}(X) = \min(n, p+1) = N \quad (\text{since } N < p+1) \\ \therefore p+1 \le N, \end{aligned}$$

which is a contradiction.

(b) When $N \ge p+1$ and there are two identical columns in X:

Since X is not of full column rank, $\operatorname{rank}(X) < p+1$. Then

$$\begin{aligned} p+1 > \operatorname{rank}(X) \ge \operatorname{rank}(X^\top X) \\ \therefore \operatorname{rank}(X^\top X) < p+1 \end{aligned}$$

Therefore, $X^\top X$ is not invertible.

Q 7

(a)

$$\begin{aligned} \hat{\beta} &= (X^\top X)^{-1} X^\top y \\ &= (X^\top X)^{-1} X^\top (X\beta + \epsilon) \\ &= (X^\top X)^{-1} (X^\top X)\beta + (X^\top X)^{-1} X^\top \epsilon \\ &= \beta + (X^\top X)^{-1} X^\top \epsilon. \end{aligned}$$

(b)

What we need to show is $\mathbb{E}(\hat{\beta}) = \beta$.

$$\begin{aligned}
\mathbb{E}(\hat{\beta}) &= \mathbb{E}(\beta + (X^\top X)^{-1} X^\top \epsilon) \\
&= \beta + \mathbb{E}((X^\top X)^{-1} X^\top \epsilon) \\
&= \beta + \mathbb{E}[\mathbb{E}\{(X^\top X)^{-1} X^\top \epsilon | X\}] \quad \text{(by the law of iterated expectation)} \\
&= \beta + \mathbb{E}[(X^\top X)^{-1} X^\top \mathbb{E}\{\epsilon | X\}] \\
&= \beta + \mathbb{E}[(X^\top X)^{-1} X^\top \mathbb{E}\{\epsilon\}] \quad \text{(since } X \text{ and } \epsilon \text{ are independent)} \\
&= \beta \quad (\mathbb{E}(\epsilon) = 0).
\end{aligned}$$

(c)

$$\begin{aligned}
(\hat{\beta} - \beta)(\hat{\beta} - \beta)^\top &= (X^\top X)^{-1} X^\top \epsilon [(X^\top X)^{-1} X^\top \epsilon]^\top \\
&= (X^\top X)^{-1} X^\top \epsilon \epsilon^\top X (X^\top X)^{-1}
\end{aligned}$$

Since $\sigma^2 I = \mathbb{E}[(\epsilon - \mathbb{E}(\epsilon)(\epsilon - \mathbb{E}(\epsilon)^\top] = \mathbb{E}[\epsilon\epsilon^\top]$, and X and ϵ are independent, it follows that

$$\mathbb{E}[\epsilon\epsilon^\top | X] = \mathbb{E}[\epsilon\epsilon^\top] = \sigma^2 I.$$

Then

$$\begin{aligned}
\mathbb{E}[(\hat{\beta} - \beta)(\hat{\beta} - \beta)^\top | X] &= \mathbb{E}[(X^\top X)^{-1} X^\top \epsilon\epsilon^\top X (X^\top X)^{-1} | X] \\
&= (X^\top X)^{-1} X^\top \underbrace{\mathbb{E}[\epsilon\epsilon^\top | X]}_{=\sigma^2 I} X (X^\top X)^{-1} \\
&= \sigma^2 (X^\top X)^{-1} X^\top X (X^\top X)^{-1} \\
&= \sigma^2 (X^\top X)^{-1}
\end{aligned}$$

Q 8

(a)

$$\begin{aligned}
H^2 &= X \underbrace{(X^\top X)^{-1} X^\top X}_{=I} (X^\top X)^{-1} X^\top \\
&= X (X^\top X)^{-1} X^\top \\
&= H.
\end{aligned}$$

(b)

$$\begin{aligned}
(I - H)^2 &= (I - H)(I - H) \\
&= I - 2H + H^2 \\
&= I - 2H + H \\
&= I - H.
\end{aligned}$$

(c)

$$
\begin{aligned}
HX &= X \underbrace{(X^\top X)^{-1} X^\top X}_{=I} \\
&= X.
\end{aligned}
$$

(d)

$$
\begin{aligned}
\hat{y} :=& X\hat{\beta} \\
=& \underbrace{X(X^\top X)^{-1} X^\top}_{=:H} y \\
=& Hy.
\end{aligned}
$$

(e)

$$
\begin{aligned}
y - \hat{y} :=& y - Hy \\
=& (I - H)y \\
=& (I - H)(X\beta + \epsilon) \\
=& X\beta - \underbrace{HX}_{=X} \beta + (I - H)\epsilon \\
=& X\beta - X\beta + (I - H)\epsilon \\
=& (I - H)\epsilon.
\end{aligned}
$$

(f)

Note that

$$
\begin{aligned}
(I - H)^\top =& I - H^\top \\
=& I - [X(X^\top X)X^\top]^\top \\
=& I - X[(X^\top X)^{-1}]^\top X^\top \\
=& I - X(X^\top X)^{-1} X^\top \\
=& I - H.
\end{aligned}
$$

Then

$$
\begin{aligned}
\|y - \hat{y}\| =& (y - \hat{y})^\top (y - \hat{y}) \\
=& [(I - H)\epsilon]^\top (I - H)\epsilon \\
=& \epsilon^\top (I - H)^\top (I - H)\epsilon \\
=& \epsilon^\top (I - H)^2 \epsilon \\
=& \epsilon^\top (I - H)\epsilon.
\end{aligned}
$$

Q 12

```r
# Number of observations
N <- 100
# Generate data
x <- rnorm(N)
y <- rnorm(N)
# Compute mean
x_bar <- mean(x)
y_bar <- mean(y)
# Compute OLS
beta0 <- sum(y_bar * sum(x^2) - x_bar * sum(x * y)) / sum((x - x_bar)^2)
beta1 <- sum((x - x_bar) * (y - y_bar)) / sum((x - x_bar)^2)
# Compute residual sum of squares
RSS <- sum((y - beta0 - beta1 * x)^2)
RSE <- sqrt(RSS / (N - 1 - 1))
B0 <- (sum(x^2) / N) / sum((x - x_bar)^2)
B1 <- 1 / sum((x - x_bar)^2)
# Standard errors
se0 <- RSE * sqrt(B0)
se1 <- RSE * sqrt(B1)
# t statistics
t0 <- beta0/se0
t1 <- beta1/se1
# p values
p0 <- 2 * (1 - pt(abs(t0), N - 2))
p1 <- 2 * (1 - pt(abs(t1),N - 2))
# Show the outputs
estimate_12 <-
  data.frame(
    param = c('intercept', 'slope'),
    beta = c(beta0, beta1),
    se = c(se0, se1),
    t_stat = c(t0, t1),
    p_value = c(p0, p1)
  )
estimate_12
```

```
##       param       beta         se    t_stat   p_value
## 1 intercept 0.08203939 0.10024758 0.8183678 0.4151327
## 2     slope 0.11177110 0.09789483 1.1417467 0.2563414
```

```r
summary(lm(y ~ x))
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.78066 -0.56388 -0.02833  0.68122  1.93067
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.08204    0.10025   0.818    0.415
## x            0.11177    0.09789   1.142    0.256
```

```
##
## Residual standard error: 1.002 on 98 degrees of freedom
## Multiple R-squared:  0.01313,    Adjusted R-squared:  0.003057
## F-statistic: 1.304 on 1 and 98 DF,  p-value: 0.2563
```

2. Classification

Q 20

Take any $x \in \mathbb{R}^p$. What we need to show is that for each $y \in \{1, -1\}$,

$$P(Y = y) = \frac{1}{1 + \exp(-y(\beta_0 + \beta^T x))}$$

holds.

When $y = 1$,

$$\begin{aligned}
\frac{1}{1 + \exp(-y(\beta_0 + \beta^T x))} &= \frac{1}{1 + \exp(-(\beta_0 + \beta^T x))} \\
&= \frac{\exp(\beta_0 + \beta^T x)}{[1 + \exp(-(\beta_0 + \beta^T x))] \exp(\beta_0 + \beta^T x)} \\
&= \frac{\exp(\beta_0 + \beta^T x)}{\exp(\beta_0 + \beta^T x) + 1} \\
&= P(Y = 1).
\end{aligned}$$

When $y = -1$,

$$\begin{aligned}
\frac{1}{1 + \exp(-y(\beta_0 + \beta^T x))} &= \frac{1}{1 + \exp(\beta_0 + \beta^T x)} \\
&= P(Y = -1),
\end{aligned}$$

which completes the proof.

Q 21

Taking the derivative of f,

$$\begin{aligned}
f'(x) &= \frac{\beta \exp(-\beta_0 - \beta x)}{[1 + \exp(-\beta_0 - \beta x)]^2} \\
&= \beta \frac{\exp(-\beta_0 - \beta x)}{1 + \exp(-\beta_0 - \beta x)} \cdot \frac{1}{1 + \exp(-\beta_0 - \beta x)} \\
&= \beta \frac{\exp(-\beta_0 - \beta x)}{1 + \exp(-\beta_0 - \beta x)} \cdot f(x) \\
&= \beta \frac{1 + \exp(-\beta_0 - \beta x) - 1}{1 + \exp(-\beta_0 - \beta x)} \cdot f(x) \\
&= \beta \left(1 - \frac{1}{1 + \exp(-\beta_0 - \beta x)}\right) f(x) \\
&= \beta \left(1 - f(x)\right) f(x) \\
&= \beta f(x) \left(1 - f(x)\right)
\end{aligned}$$

Since $f(x) > 0$ for all $x \in \mathbb{R}$, $f'(x) > 0$ for all $x \in \mathbb{R}$. Thus f is monotonically increasing.

Let $A := 1 + \exp(-\beta_0 - \beta x)$. Using the result above,

$$
\begin{aligned}
\frac{f''(x)}{\beta} &= f'(x)\,(1 - f(x)) + f(x)(-f'(x)) \\
&= \beta f(x)\,(1 - f(x))^2 - \beta [f(x)]^2\,(1 - f(x)) \\
&= \beta f(x)\,(1 - f(x))\,(1 - f(x) - f(x)) \\
&= f'(x)(1 - 2f(x)) \\
&= f'(x)\left[1 - \frac{2}{1 + \exp(-\beta_0 - \beta x)}\right] \\
&= f'(x)\left[\frac{\exp(-\beta_0 - \beta x) - 1}{1 + \exp(-\beta_0 - \beta x)}\right] \\
&= \frac{f'(x)}{f(x)}[\exp(-\beta_0 - \beta x) - 1] \\
\therefore f''(x) &= \beta \frac{f'(x)}{f(x)}[\exp(-\beta_0 - \beta x) - 1]
\end{aligned}
$$

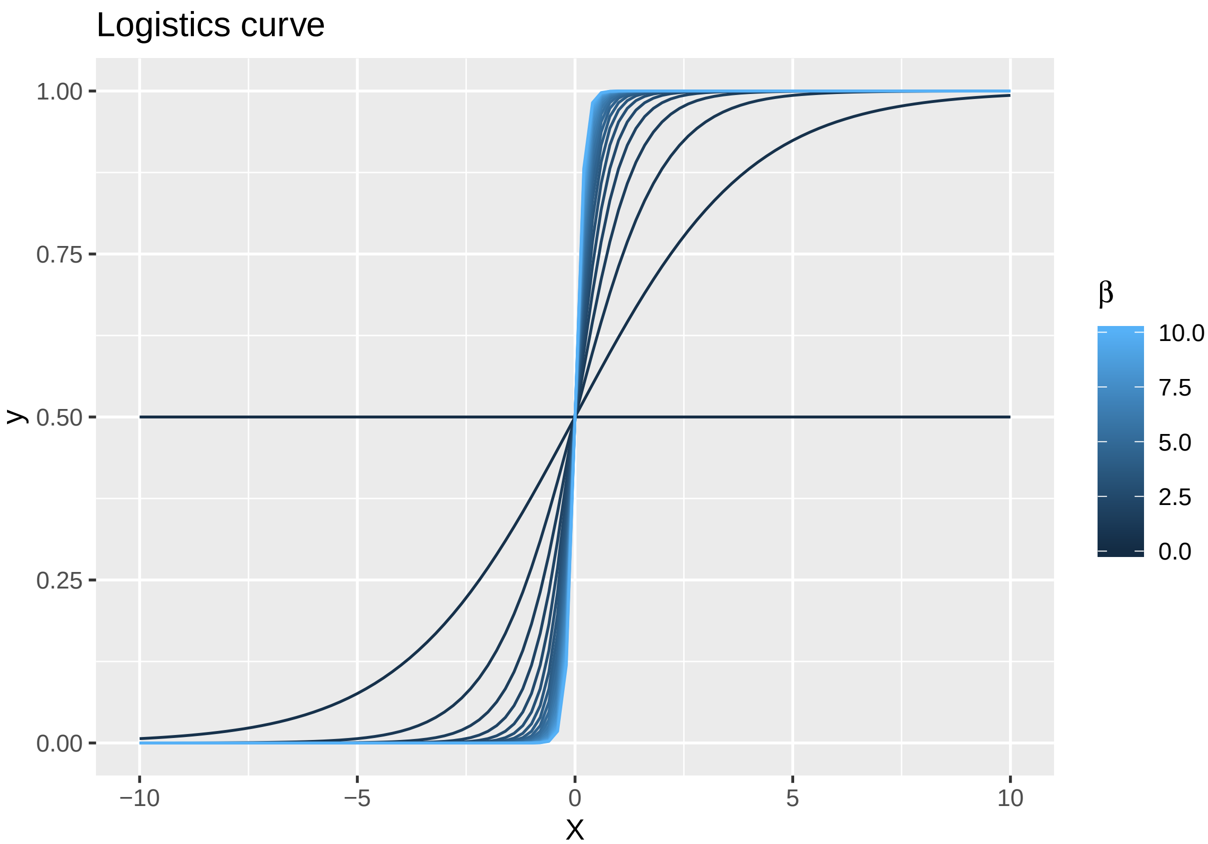
Note that $\beta \frac{f'(x)}{f(x)} > 0$. Thus $f(x)$ is convex if and only if:

$$
\begin{aligned}
f''(x) > 0 &\Leftrightarrow \exp(-\beta_0 - \beta x) - 1 > 0 \\
&\Leftrightarrow \exp(-\beta_0 - \beta x) > 1 \\
&\Leftrightarrow -\beta_0 - \beta x > 0 \\
&\Leftrightarrow x < -\frac{\beta_0}{\beta}.
\end{aligned}
$$

Analogously, we can show that $f(x)$ is concave if and only if:

$$
f''(x) < 0 \Leftrightarrow x > -\frac{\beta_0}{\beta}.
$$

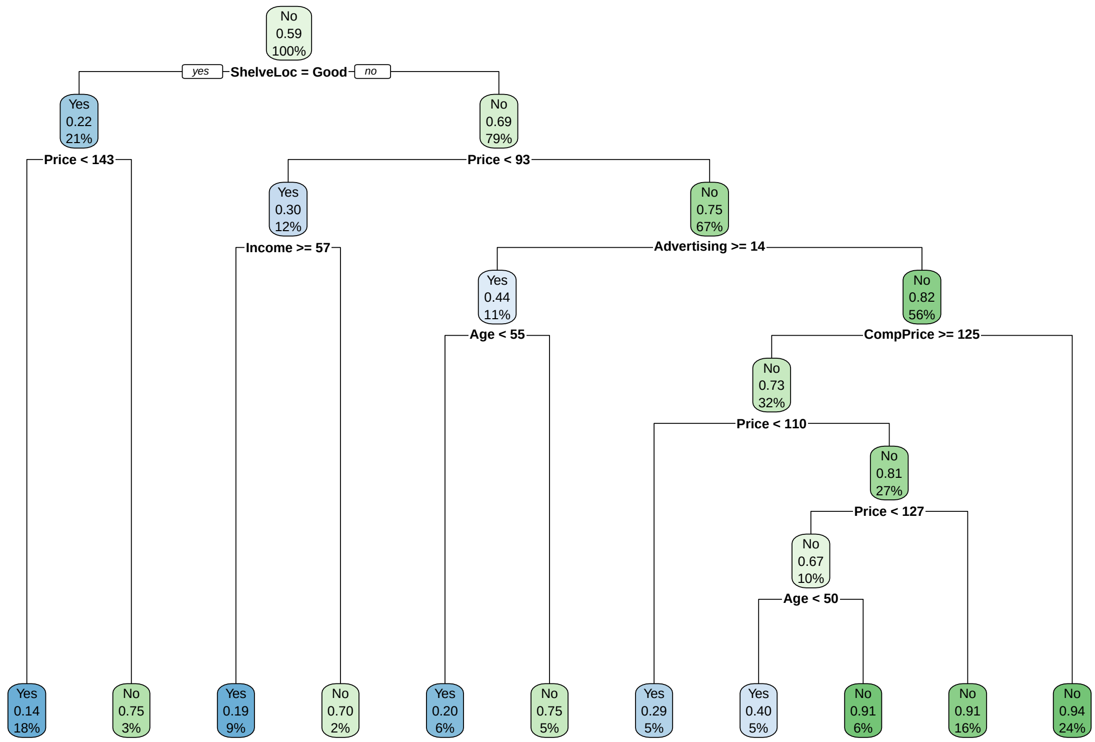
```r
# Define a function
logit <- function(x, beta = 1, beta_0 = 0){
  output <- exp(beta_0 + beta * x) / (1 + exp(beta_0 +beta * x))
  return(output)
}
# Plot the function for each beta
beta_seq_expand <- expand.grid(b = seq(0, 10, by = .5))
g_21 <- ggplot(data.frame(X = c(-10, 10)), aes(x = X)) +
  mapply(
    function(b, co) stat_function(fun = logit, args = list(beta = b), aes_q(color = co)),
    beta_seq_expand$b, beta_seq_expand$b
  ) +
  ggtitle("Logistics curve") +
  labs(color = latex2exp::TeX("$\\beta$"))
plot(g_21)
```



7. Decision Tree

Q 68

```
library(ISLR)
# Make a dataset for the excercise
df_68 <- Carseats %>%
  dplyr::mutate(High = dplyr::if_else(Sales <= 8, "No", "Yes")) %>% # Classify based on sales
  dplyr::mutate_each(
    dplyr::funs(
      forcats::fct_relevel(., "Yes", "No")
    ),
    Urban, US, High
  ) %>%
  dplyr::mutate(ShelveLoc = forcats::fct_relevel(ShelveLoc, "Good", "Medium", "Bad"))
# Create a decision tree without the variable 'Sales'
set.seed(1)
tree_68 <- rpart::rpart(
  formula = High ~ . -Sales,
  data = df_68
)
# Plot the decision tree
rpart.plot::rpart.plot(tree_68)
```




```r
# Select 200 observations randomly to create a training dataset.
df_68_train <- sample(1:nrow(df_68), 200)
# The rest of observations goes to a test dataset.
df_68_test <- df_68[-df_68_train,]
# Train the model
tree_68_train <- rpart::rpart(
  formula = High ~ .,
  data = df_68,
  subset = df_68_train
)
# Predict
High_pred <- predict(tree_68_train, df_68_test, type = "class")
# Evaluate the prediction performance
High_true <- df_68_test$High
table(High_pred, High_true)
```

```
##          High_true
## High_pred Yes  No
##       Yes  74   2
##       No    0 124
```

Q 69

(a)

Since $0 < \frac{\alpha_{m,k}}{\sum_{k'=1}^K \alpha_{m,k'}} \leq 1$ for any $m \in \{1, \cdots, N\}$ and $k \in \{1, \cdots, K\}$, $\log \frac{\alpha_{m,k}}{\sum_{k'=1}^K \alpha_{m,k'}} \leq 0$. It follows that $-\frac{\alpha_{m,k}}{N} \log \frac{\alpha_{m,k}}{\sum_{k'=1}^K \alpha_{m,k'}} \geq 0$ for any $m \in \{1, \cdots, N\}$ and $k \in \{1, \cdots, K\}$. Thus

$$H := \sum_{m=1}^M \sum_{k=1}^K \left(-\frac{\alpha_{m,k}}{N} \log \frac{\alpha_{m,k}}{\sum_{k'=1}^K \alpha_{m,k'}} \right) \geq 0.$$

(b)

Suppose $H = 0$. Then

$$\begin{aligned} H = 0 &\Leftrightarrow \sum_{k=1}^K \left(-\frac{\alpha_{m,k}}{N} \log \frac{\alpha_{m,k}}{\sum_{k'=1}^K \alpha_{m,k'}} \right) = 0 \quad \forall m \\ &\Leftrightarrow \alpha_{m,k} = 0 \vee \log \frac{\alpha_{m,k}}{\sum_{k'=1}^K \alpha_{m,k'}} = 0 \quad \forall m. \end{aligned}$$

Since $\sum_{k=1}^K \alpha_{m,k} \geq 1$ for any m, there exists k such that $\alpha_{m,k} \geq 1$ and $\log \frac{\alpha_{m,k}}{\sum_{k'=1}^K \alpha_{m,k'}} = 0$ for any m. For that m,

$$\alpha_{m,k} = \sum_{k'=1}^K \alpha_{m,k'}$$

holds for any m.

Suppose for any m, there exists $j \in \{1, \cdots, K\}$ such that $\alpha_{m,j} = \sum_{k'=1}^K \alpha_{m,k'}$. Take any m. Since $\alpha_{m,j} = \sum_{k'=1}^K \alpha_{m,k'}$, it follows that $\alpha_{m,k'} = 0$ for any $k' \in \{1, \cdots, K\} \setminus \{j\}$. Thus we have

$$\begin{aligned} \sum_{k=1}^K \left(-\frac{\alpha_{m,k}}{N} \log \frac{\alpha_{m,k}}{\sum_{k'=1}^K \alpha_{m,k'}} \right) &= -\frac{\alpha_{m,j}}{N} \log \underbrace{\frac{\alpha_{m,j}}{\sum_{k'=1}^K \alpha_{m,k'}}}_{=1} + \sum_{k \neq j} \left(-\underbrace{\frac{\alpha_{m,k}}{N}}_{=0} \log \frac{\alpha_{m,k}}{\sum_{k'=1}^K \alpha_{m,k'}} \right) \\ &= 0. \end{aligned}$$

Therefore,

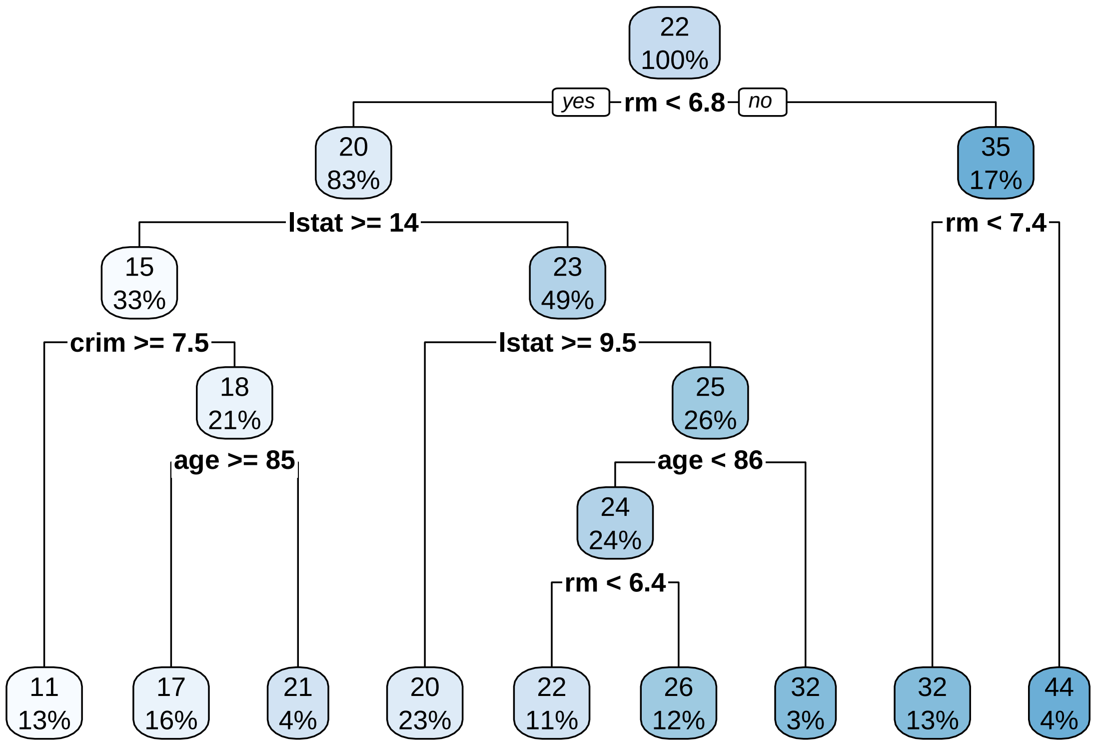
$$H := \sum_{m=1}^M \sum_{k=1}^K \left(-\frac{\alpha_{m,k}}{N} \log \frac{\alpha_{m,k}}{\sum_{k'=1}^K \alpha_{m,k'}} \right) = 0.$$

Q 70

```r
# Import dataset
library(MASS)
df_70 <- Boston %>%
  tibble::as_tibble()
names(df_70)
```

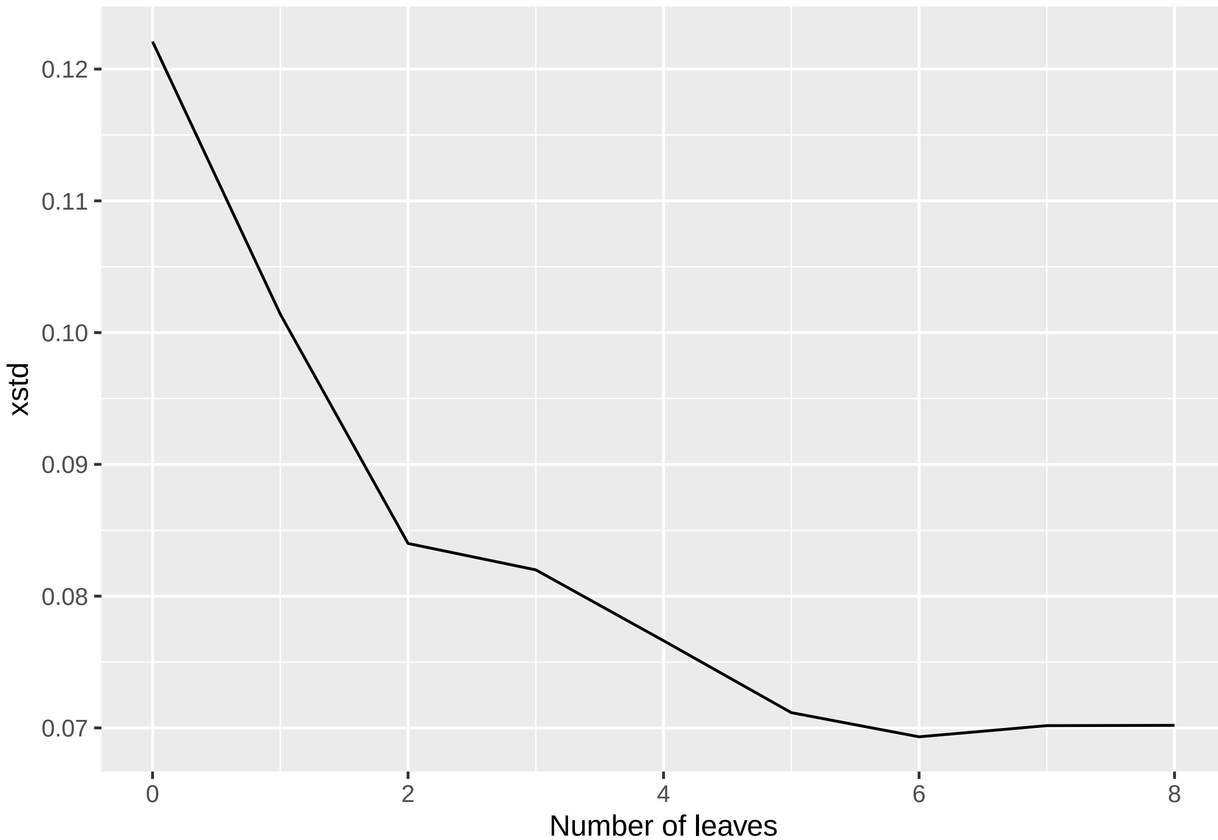
```
##  [1] "crim"    "zn"      "indus"   "chas"    "nox"     "rm"      "age"
##  [8] "dis"     "rad"     "tax"     "ptratio" "black"   "lstat"   "medv"
```

```r
# Training and test data
train_70 <- sample(1:nrow(Boston), nrow(Boston)/2)
tree_70 <- rpart::rpart(
  formula = medv ~ .,
  data = df_70,
  subset = train_70
)
# Plot the decision tree
rpart.plot::rpart.plot(tree_70)
```



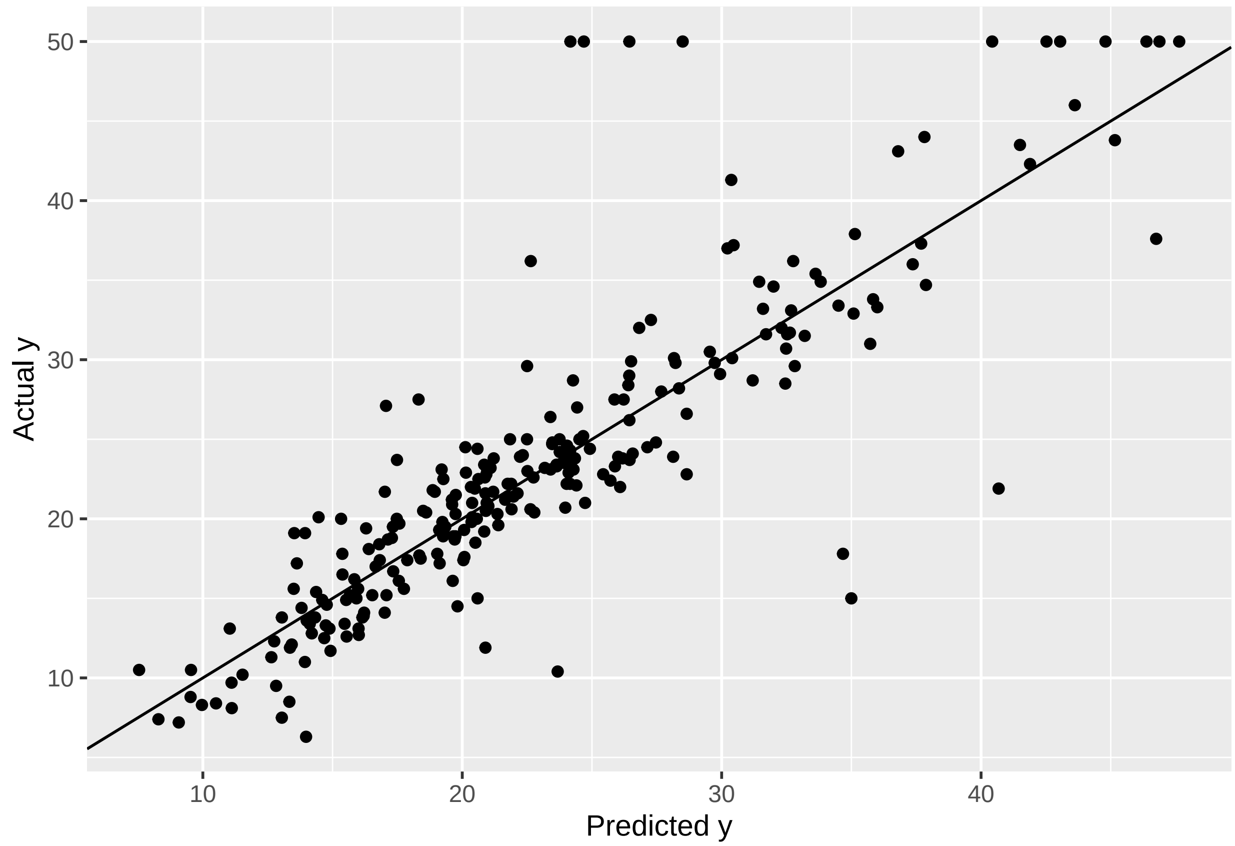

```r
# Cross validation
```

```r
# PLot CP and xstd
df_g_70 <- tree_70$cptable %>%
  tibble::as_tibble() %>%
  dplyr::select(nsplit, xstd)
g_70 <- ggplot(data = df_g_70, aes(x = nsplit, y = xstd)) +
  geom_line() +
  xlab("Number of leaves")
plot(g_70)
```



Q 71

```
# Import dataset
library(MASS)
df_71 <- Boston %>%
  tibble::as_tibble()
# Set seed
set.seed(1)
# Split into training and test data
train_71 <- sample(1:nrow(Boston), nrow(Boston)/2)
test_71 <- Boston[-train_71, "medv"]
# Traing the model
bag_71 <- randomForest::randomForest(medv ~ ., data = Boston, subset = train_71, mtry = 13)
# Make prediction
yhat_71 <- predict(bag_71, newdata = Boston[-train_71,])
# Draw scatterplot
g_71 <- ggplot(data = data.frame(yhat_71, test_71), aes(x = yhat_71, test_71)) +
  geom_point() +
  geom_abline(intercept = 0, slope = 1) +
  xlab("Predicted y") +
  ylab("Actual y")
plot(g_71)
```



Q 72

```
# Import dataset
library(MASS)
df_72 <- Boston %>%
  tibble::as_tibble()
# Set seed
set.seed(1)
# Split into training and test data
train_72 <- sample(1:nrow(Boston), nrow(Boston)/2)
test_72 <- Boston[-train_72, "medv"]
# Create dataset
output_72 <- foreach(i = 2:13, .combine = 'rbind') %dopar% {
  # Traing the model
  bag <- randomForest::randomForest(medv ~ ., data = Boston, subset = train_72, mtry = i)
  # Make prediction
  yhat <- predict(bag, newdata = Boston[-train_72,])
  # Calculate MSE
  mse <- mean((yhat - test_72)^2)
  output <- data.frame(
    mtry = i,
    mse = mse
  )
}
# Plot the relationship between mtry and mse
```

```
g_72 <- ggplot(data = output_72, aes(x = mtry, y = mse)) +
  geom_line() +
  scale_x_continuous(breaks = 2:13)
plot(g_72)
```