4.1. Consider the regression on time, $y_t = \beta_0 + \beta_1 t + \epsilon_t$, with $t = 1, 2, \ldots, n$. Here, the regressor vector is $\boldsymbol{x}' = (1, 2, \ldots, n)$. Take $n = 10$. Write down the matrices $X'X$, $(X'X)^{-1}$, $V(\hat{\beta})$, and the variances of $\hat{\beta}_0$ and $\hat{\beta}_1$.

$$X = \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 3 \\ \vdots & \vdots \\ 1 & 10 \end{bmatrix}$$

solved via R

① $X'X = \begin{bmatrix} 10 & 55 \\ 55 & 385 \end{bmatrix}$

② $(X'X)^{-1} = \begin{bmatrix} 0.467 & -0.0667 \\ -0.0667 & 0.0121 \end{bmatrix}$

③ $V(\hat{\underset{\sim}{\beta}}) = \sigma^2 (X'X)^{-1} = \sigma^2 \begin{bmatrix} 0.467 & -0.0667 \\ -0.0667 & 0.0121 \end{bmatrix}$

④ $Var(\hat{\beta}_0) = \sigma^2(0.467)$

$Var(\hat{\beta}_1) = \sigma^2(0.0121)$

Homework 3 - Linear Models

Q4.1

```
# Define matrix X as a 2 by 10 matrix, first column of all 1s, second column of 1 to 10
X <- cbind(1, 1:10)
X
```

```
##       [,1] [,2]
##  [1,]    1    1
##  [2,]    1    2
##  [3,]    1    3
##  [4,]    1    4
##  [5,]    1    5
##  [6,]    1    6
##  [7,]    1    7
##  [8,]    1    8
##  [9,]    1    9
## [10,]    1   10
```

```
# Compute and print X'X matrix
t(X) %*% X
```

```
##      [,1] [,2]
## [1,]   10   55
## [2,]   55  385
```

```
# Compute and print (X'X)^-1 matrix
solve(t(X) %*% X)
```

```
##             [,1]        [,2]
## [1,]  0.46666667 -0.06666667
## [2,] -0.06666667  0.01212121
```

Variance of Beta_hat Matrix and Variance of b0_hat and b1_hat

The written expression of the matrices can be found in the written submission.

Q4.5

After fitting the regression model, we have the following data:

- Mean square error: $s^2 = 3$
- Inverse of $X'X$:

$$(X'X)^{-1} = \begin{bmatrix} 0.5 & 0.3 & 0.2 & 0.6 \\ 0.3 & 6.0 & 0.5 & 0.4 \\ 0.2 & 0.5 & 0.2 & 0.7 \\ 0.6 & 0.4 & 0.7 & 3.0 \end{bmatrix}$$

Part a: Estimate of V(β1)

```
# Given values
s_squared <- 3
XtX_inv <- matrix(c(0.5, 0.3, 0.2, 0.6,
                    0.3, 6.0, 0.5, 0.4,
                    0.2, 0.5, 0.2, 0.7,
                    0.6, 0.4, 0.7, 3.0),
                  nrow = 4, byrow = TRUE)

# Estimate of V(β1_hat)
V_beta1_hat <- s_squared * XtX_inv[2, 2]
V_beta1_hat
```

```
## [1] 18
```

Part b: Estimate of Cov(β1_hat, β3_hat)

```
# Estimate of Cov(β1_hat, β3_hat)
Cov_beta1_beta3 <- s_squared * XtX_inv[2, 4]
Cov_beta1_beta3
```

```
## [1] 1.2
```

Part c: Estimate of Corr(β1_hat, β3_hat)

```
# Estimate of V(β3_hat)
V_beta3_hat <- s_squared * XtX_inv[4, 4]

# Estimate of Corr(β1_hat, β3_hat) via pearson correlation coefficient
Corr_beta1_beta3 <- Cov_beta1_beta3 / sqrt(V_beta1_hat * V_beta3_hat)
Corr_beta1_beta3
```

```
## [1] 0.0942809
```

Part d: Estimate of V(β1_hat - β3_hat)

```
# Estimate of V(β1_hat - β3_hat)
V_beta1_minus_beta3_hat <- V_beta1_hat + V_beta3_hat - 2 * Cov_beta1_beta3
V_beta1_minus_beta3_hat
```

```
## [1] 24.6
```

Q4.7

Part a: Calculate Coefficient of determination from given table

```
ss_reg <- 504541
ss_total <- 541119

# Coefficient of determination
R_squared <- ss_reg / ss_total
R_squared
```

```
## [1] 0.932403
```

Part b: Hypothesis Testing

H0: β1 = β2 = β3 = 0 alpha = 0.05

```
# Set up F test
p <- 3
n <- 28
msr <- ss_reg / p
mse <- (ss_total - ss_reg) / (n - p - 1)
F_stat <- msr / mse

# Perform F test
p_value <- 1 - pf(F_stat, p, n - p - 1)

# Print results
cat("F statistic:", F_stat, "\n")
```

```
## F statistic: 110.3485
```

```
cat("p-value:", p_value, "\n")
```

```
## p-value: 3.552714e-14
```

Since the p-value is less than 0.05, we reject the null hypothesis. This means that at least one of the coefficients is not equal to 0.

Part c: 95% Confidence Interval for β1 (taxes)

```
# Given values from table
taxes_β1 <- 0.18966
se_taxes_β1 <- 0.05623

baths_β2 <- 81.87
se_baths_β2 <- 47.82

# Construct 95% confidence interval for β1
t_value <- qt(0.975, n - p - 1)
CI_lower <- taxes_β1 - t_value * se_taxes_β1
CI_upper <- taxes_β1 + t_value * se_taxes_β1
cat("95% Confidence Interval for β1 (taxes): [", CI_lower, ", ", CI_upper, "]\n")
```

```
## 95% Confidence Interval for β1 (taxes): [ 0.07360698 ,  0.305713 ]
```

Since the confidence interval does not contain 0, we can conclude that the coefficient for taxes is not equal to 0. Reject the null hypothesis. Cannot simplify by dropping “taxes” from the model.

Part d: 95% Confidence Interval for β2 (baths)

```
# Construct 95% confidence interval for β2
CI_lower <- baths_β2 - t_value * se_baths_β2
CI_upper <- baths_β2 + t_value * se_baths_β2
cat("95% Confidence Interval for β2 (baths): [", CI_lower, ", ", CI_upper, "]\n")
```

```
## 95% Confidence Interval for β2 (baths): [ -16.82563 ,  180.5656 ]
```

Since the confidence interval contains 0, we cannot conclude that the coefficient for baths is not equal to 0. Fail to reject the null hypothesis. Can simplify by dropping “baths” from the model.

Q4.10

Part a: Write down the Estimated Equation

```
# Define given matrices
XtX <- matrix(c(15, 3626, 44428,
                3626, 1067614, 11419181,
                44428, 11419181, 139063428),
              nrow = 3, ncol = 3, byrow = TRUE)

Xty <- matrix(c(2259, 647107, 7096619),
              nrow = 3, ncol = 1)

XtX_inv <- matrix(c(1.2463484, 2.1296642e-4, -4.1567125e-4,
                    2.1296642e-4, 7.7329030e-6, -7.0302518e-7,
                    -4.1567125e-4, -7.0302518e-7, 1.9771851e-7),
                  nrow = 3, ncol = 3, byrow = TRUE)

beta_hat <- matrix(c(3.452613, 0.496005, 0.009191),
               nrow = 3, ncol = 1)

yty <- 394107
```

```
# estimated equation
cat("Estimated Equation: y_hat = ", beta_hat[1], " + ", beta_hat[2], "x1 + ", beta_hat[3], "x
2\n")
```

```
## Estimated Equation: y_hat =  3.452613  +  0.496005 x1 +  0.009191 x2
```

Part b: T statistic hypothesis testing for each regression coefficient

```
# Define values needed
n <- 15
p <- 2

# Estimate sigma squared with MSE, hence solve for MSE
RSS <- (yty - 2*(t(beta_hat) %*% Xty) + t(beta_hat) %*% XtX %*% beta_hat)
s_squared <- as.numeric(RSS) / (n - p - 1) #MSE

# Calculate standard errors of the regression coefficients
se_beta <- sqrt(s_squared * diag(XtX_inv))

# Ouput standard errors of the regression coefficients
cat("Standard Error of beta0_hat:", se_beta[1], "\n")
```

```
## Standard Error of beta0_hat: 2.430844
```

```
cat("Standard Error of beta1_hat:", se_beta[2], "\n")
```

```
## Standard Error of beta1_hat: 0.006054924
```

```
cat("Standard Error of beta2_hat:", se_beta[3], "\n")
```

```
## Standard Error of beta2_hat: 0.0009681911
```

Part c: T test for each regression coefficient

H0: $\beta i = 0$ for all i = 0, 1, 2

```
# Compute t statistic for regression coefficients
t_beta <- beta_hat / se_beta

# Perform t test for regression coefficients
p_values <- 2 * (1 - pt(abs(t_beta), n - p - 1))


# Create a decision column to determine if we reject the null hypothesis at alpha = 0.05
decision <- ifelse(p_values < 0.05, "Reject H0", "Fail to Reject H0")

# Output the coefficient, standard error, t statistic, p-value and decision for each regressi
on coefficient in a table
t_test_table <- data.frame(Coefficient = c("beta0", "beta1", "beta2"),
                           Estimate = c(beta_hat[1], beta_hat[2], beta_hat[3]),
                           Standard_Error = se_beta,
                           t_statistic = t_beta,
                           P_value = p_values,
                           Decision = decision)

t_test_table
```

Coefficient	Estimate	Standard_Error	t_statistic	P_value	Decision
<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>
beta0	3.452613	2.4308443870	1.420335	1.809675e-01	Fail to Reject H0
beta1	0.496005	0.0060549242	81.917624	0.000000e+00	Reject H0
beta2	0.009191	0.0009681911	9.492961	6.265054e-07	Reject H0

3 rows

Based on the t test results, we can see that the p-value for β0 is greater than 0.05, so we do not reject the null hypothesis that β1 = 0. For the other coefficients, the p-values are less than 0.05, so there is sufficient evidence to reject the null hypothesis that β1 and β2 are equal to 0.

Q4.14

```
# read the data 'bsemen.txt'
bsemen <- read.table("bsemen.txt", header = TRUE)
bsemen
```

Y	X1	X2	X3
<dbl>	<dbl>	<dbl>	<dbl>
25.5	1.74	5.30	10.8
31.2	6.32	5.42	9.4
25.9	6.22	8.41	7.2
38.4	10.52	4.63	8.5
18.4	1.19	11.60	9.4
26.7	1.22	5.85	9.9
26.4	4.10	6.62	8.0
25.9	6.32	8.72	9.1
32.0	4.08	4.42	8.7
25.2	4.15	7.60	9.2

1-10 of 13 rows Previous **1** 2 Next

Part a: Compute X'X, (X'X)^-1, and X'y

```
# Define X
X <- cbind(1, bsemen$X1, bsemen$X2, bsemen$X3)

# Compute X'X
XtX <- t(X) %*% X
XtX
```

```
##        [,1]     [,2]     [,3]     [,4]
## [1,]  13.00  59.4300  81.8200  115.400
## [2,]  59.43 394.7255 360.6621  522.078
## [3,]  81.82 360.6621 576.7264  728.310
## [4,] 115.40 522.0780 728.3100 1035.960
```

```
# Compute (X'X)^-1
XtX_inv <- solve(XtX)
XtX_inv
```

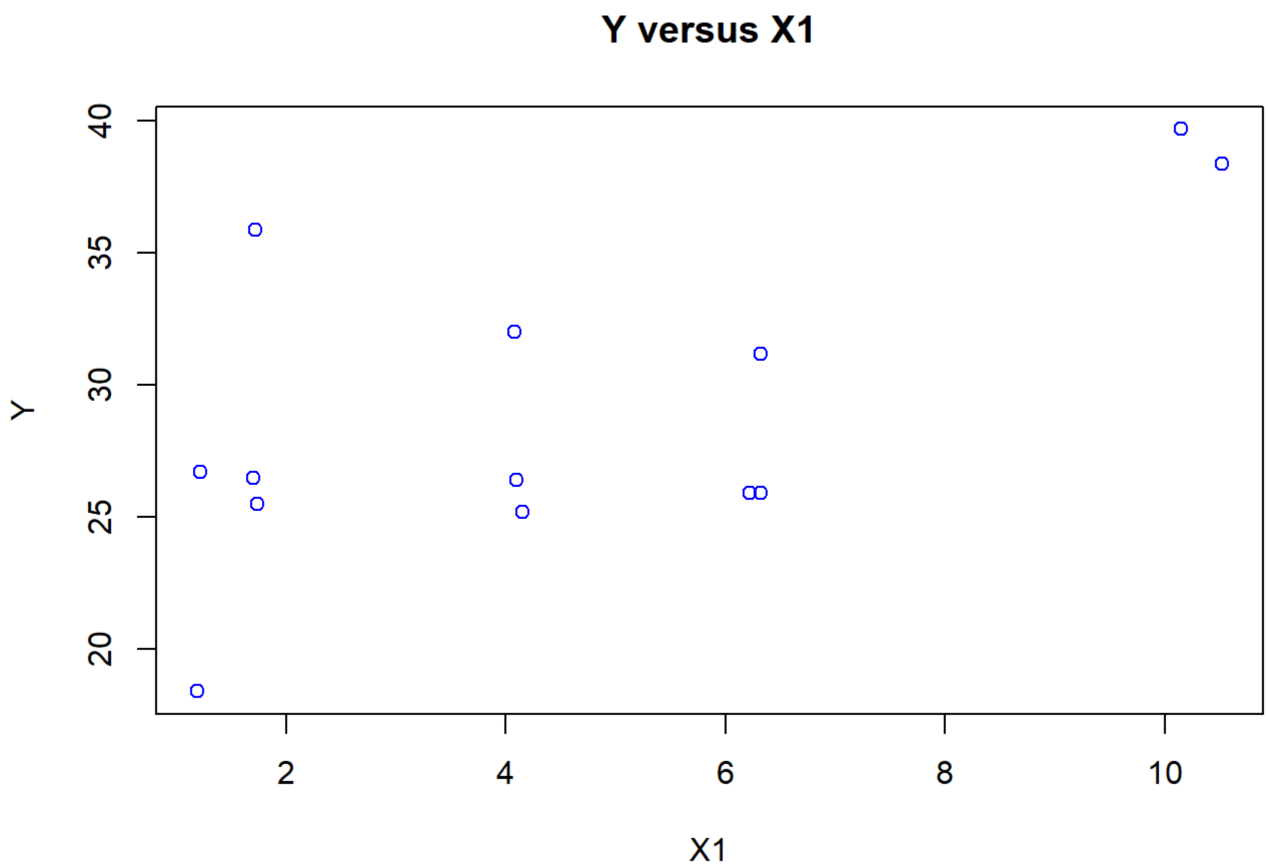
```
##             [,1]         [,2]         [,3]         [,4]
## [1,]  8.06479464 -0.082592705 -0.094195115 -0.790526876
## [2,] -0.08259271  0.008479816  0.001716687  0.003720020
## [3,] -0.09419511  0.001716687  0.016629424 -0.002063308
## [4,] -0.79052688  0.003720020 -0.002063308  0.088601286
```

```
# Compute X'y
Xty <- t(X) %*% bsemen$Y
Xty
```

```
##          [,1]
## [1,]  377.700
## [2,] 1877.911
## [3,] 2247.285
## [4,] 3339.300
```

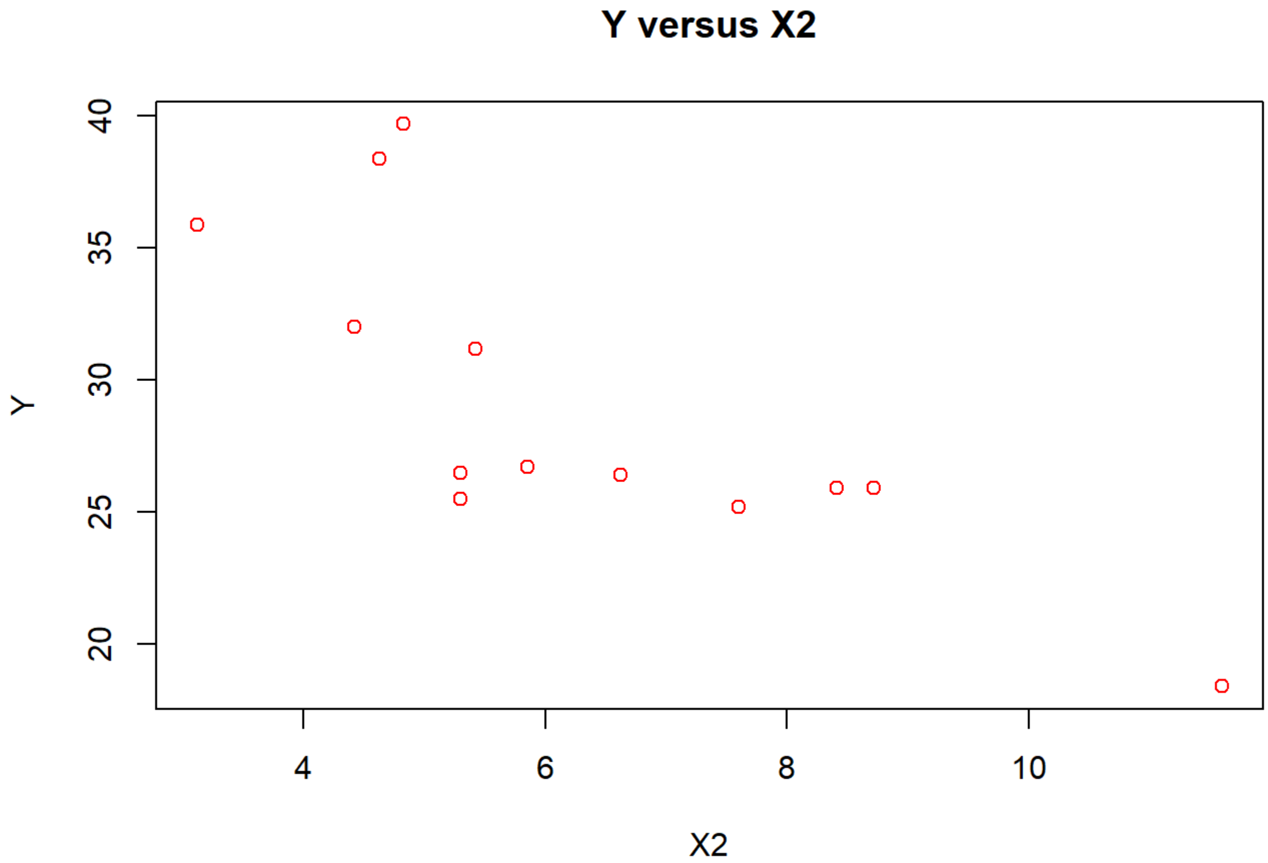
Part b: Plot y versus each predictor variable

```
# y versus x1
plot(bsemen$X1, bsemen$Y, xlab = "X1", ylab = "Y", main = "Y versus X1", col = "blue")
```



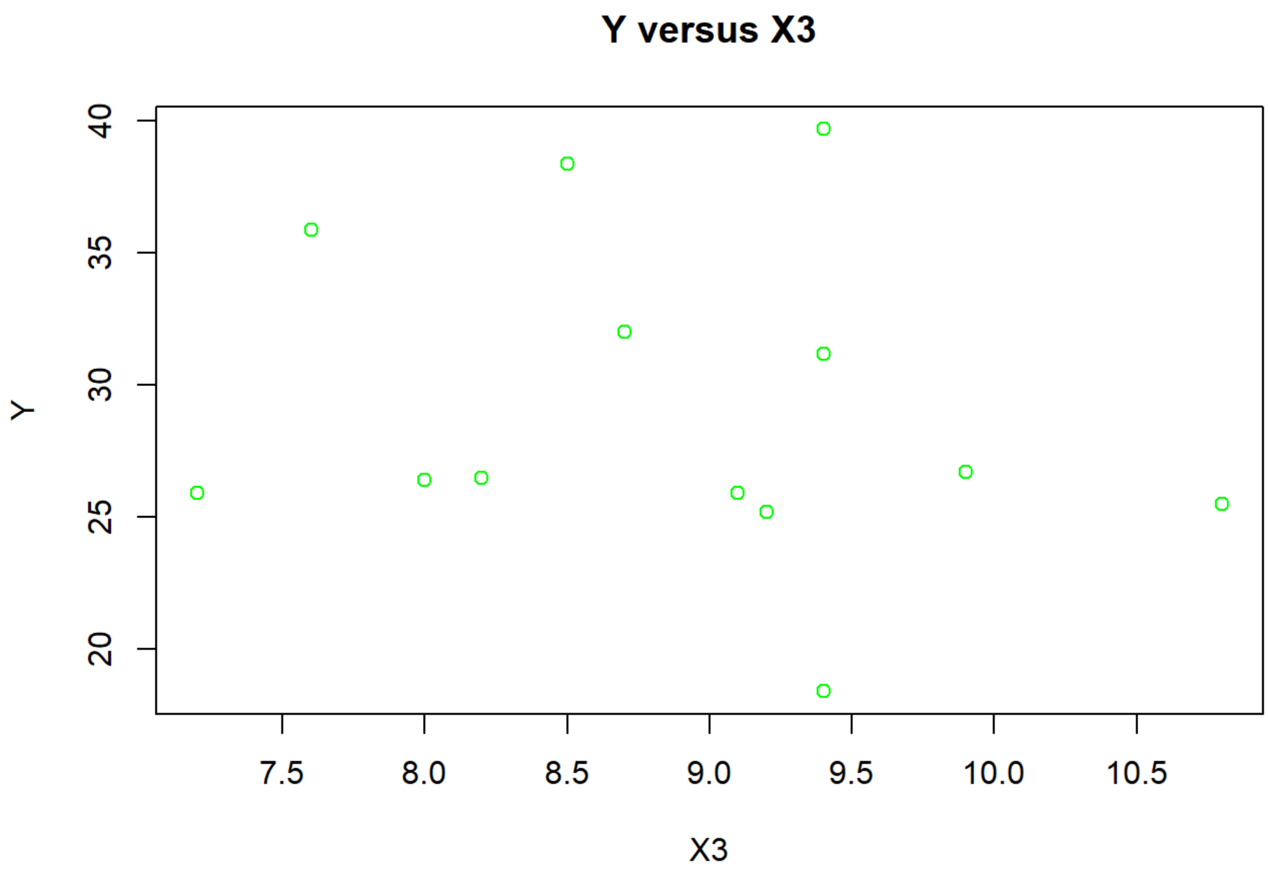
From the plot of Y versus X1, we can see that there is a positive linear relationship between Y and X1. As X1 increases, Y also increases.

```
# y versus x2
plot(bsemen$X2, bsemen$Y, xlab = "X2", ylab = "Y", main = "Y versus X2", col = "red")
```



From the plot of Y versus X2, there seems to be a slight negative relationship between Y and X2. As X2 increases, Y decreases.

```
# y versus x3
plot(bsemen$X3, bsemen$Y, xlab = "X3", ylab = "Y", main = "Y versus X3", col = "green")
```



From the plot of Y versus X3, there does not seem to be any clear relationship between Y and X3. The points are scattered and do not show a clear pattern.

Part c: Obtain least square estimates of β0, β1, β2, and β3, and state the fitted equation

```
# Compute least square estimates of beta
beta_hat <- XtX_inv %*% Xty
beta_hat
```

```
##             [,1]
## [1,] 39.4815184
## [2,]  1.0092246
## [3,] -1.8726572
## [4,] -0.3666997
```

```
# Write down the fitted equation
cat("Fitted Equation: mu_hat = ", beta_hat[1], " + ", beta_hat[2], "X1 + ", beta_hat[3], "X2
+ ", beta_hat[4], "X3\n")
```

```
## Fitted Equation: mu_hat =  39.48152  +  1.009225 X1 +  -1.872657 X2 +  -0.3666997 X3
```

Part d: Construct 90% Confidence Interval for:

i. the predicted mean value of y when X1 = 3, X2 = 8, X3 = 9
ii. the predicted individual value of y when X1 = 3, X2 = 8, X3 = 9

```
# Given values
X_new <- c(1, 3, 8, 9)
alpha <- 0.1

# fit the model with built in R functions
fit <- lm(Y ~ X1 + X2 + X3, data = bsemen)

# Construct 90% confidence interval for the predicted mean value of y
predict_mean <- predict(fit, newdata = data.frame(X1 = 3, X2 = 8, X3 = 9), interval = "confid
ence", level = 1 - alpha)
predict_mean
```

```
##        fit      lwr      upr
## 1 24.22764 22.80225 25.65302
```

```
# Construct 90% confidence interval for the predicted individual value of y
predict_ind <- predict(fit, newdata = data.frame(X1 = 3, X2 = 8, X3 = 9), interval = "predict
ion", level = 1 - alpha)
predict_ind
```

```
##        fit      lwr      upr
## 1 24.22764 20.10946 28.34582
```

Part e: Construct ANOVA table and test for significant linear relationship between y and the 3 predictor variables

H0: $\beta1 = \beta2 = \beta3 = 0$

```
# Construct ANOVA table
anova(fit)
```

	Df <int>	Sum Sq <dbl>	Mean Sq <dbl>	F value <dbl>	Pr(>F) <dbl>
X1	1	185.907091	185.907091	41.8486981	1.155412e-04
X2	1	213.426270	213.426270	48.0434152	6.824618e-05
X3	1	1.517683	1.517683	0.3416388	5.732428e-01
Residuals	9	39.981263	4.442363	NA	NA

4 rows

```
# Test for significant linear relationship between y and the 3 predictor variables
summary(fit)
```

```
##
## Call:
## lm(formula = Y ~ X1 + X2 + X3, data = bsemen)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.8888 -1.4696 -0.2933  0.5726  3.3122
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  39.4815     5.9855   6.596 9.97e-05 ***
## X1            1.0092     0.1941   5.200 0.000564 ***
## X2           -1.8727     0.2718  -6.890 7.15e-05 ***
## X3           -0.3667     0.6274  -0.584 0.573243
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.108 on 9 degrees of freedom
## Multiple R-squared:  0.9093, Adjusted R-squared:  0.8791
## F-statistic: 30.08 on 3 and 9 DF,  p-value: 5.072e-05
```

Since the p-value is less than 0.05, we reject the null hypothesis. This means that there is a significant linear relationship between y and the 3 predictor variables.

Q4.16

```
# define given matrices
# X'X matrix
XtX <- matrix(c(9, 136, 269, 260,
                136, 2114, 4176, 3583,
                269, 4176, 8257, 7104,
                260, 3583, 7104, 12276),
              nrow = 4, ncol = 4, byrow = TRUE)

# X'y vector
Xty <- c(45, 648, 1283, 1821)

# (X'X)^-1 matrix
XtX_inv <- matrix(c(9.610, 0.008, -0.279, -0.044,
                    0.008, 0.509, -0.258, 0.001,
                    -0.279, -0.258, 0.139, 0.001,
                    -0.044, 0.001, 0.001, 0.0003),
                  nrow = 4, ncol = 4, byrow = TRUE)

# beta_hat vector
beta_hat <- c(-1.163461, 0.135270, 0.019950, 0.121954)

# y'y scalar
yty <- 285

# ouput the matrices
XtX
```

```
##      [,1] [,2] [,3]  [,4]
## [1,]    9  136  269   260
## [2,]  136 2114 4176  3583
## [3,]  269 4176 8257  7104
## [4,]  260 3583 7104 12276
```

```
Xty
```

```
## [1]   45  648 1283 1821
```

```
XtX_inv
```

```
##        [,1]   [,2]   [,3]    [,4]
## [1,]  9.610  0.008 -0.279 -0.0440
## [2,]  0.008  0.509 -0.258  0.0010
## [3,] -0.279 -0.258  0.139  0.0010
## [4,] -0.044  0.001  0.001  0.0003
```

```
beta_hat
```

```
## [1] -1.163461  0.135270  0.019950  0.121954
```

Part a: Construct ANOVA table

```
# Compute degrees of freedom
p <- 4 # number of predictor variables including intercept
n <- 9
df_R <- p
df_E <- n - p - 1
df_T <- n - 1

# Compute SSR, SSE, and SST
SSE <- yty - t(beta_hat) %*% Xty
SSR <- t(beta_hat) %*% Xty - ((1/n) * 45^2)
SST <- SSR + SSE

# Compute MSR, MSE, and F statistic
MSR <- SSR / df_R
MSE <- SSE / df_E
F_stat <- MSR / MSE

# Construct ANOVA table
anova_table <- data.frame(Source = c("Regression", "Error", "Total"),
                          DF = c(df_R, df_E, df_T),
                          SS = c(SSR, SSE, SST),
                          MS = c(MSR, MSE, NA),
                          F = c(F_stat, NA, NA),
                          P_value = c(pf(F_stat, df_R, df_E), NA, NA))

anova_table
```

Source <chr>	DF <dbl>	SS <dbl>	MS <dbl>	F <dbl>	P_value <dbl>
Regression	4	57.973299	14.4933247	28.60476	0.9966541
Error	4	2.026701	0.5066753	NA	NA
Total	8	60.000000	NA	NA	NA

3 rows

Part b: State the computed regression equation and standard errors of the regression coefficients

```
# Write down the computed regression equation
cat("Computed Regression Equation: y_hat = ", beta_hat[1], " + ", beta_hat[2], "x1 + ", beta_hat[3], "x2 + ", beta_hat[4], "x3\n")
```

```
## Computed Regression Equation: y_hat =  -1.163461  +  0.13527 x1 +  0.01995 x2 +  0.121954 x3
```

```
s_squared <- MSE
se_beta0_hat <- sqrt(s_squared * XtX_inv[1, 1])
se_beta1_hat <- sqrt(s_squared * XtX_inv[2, 2])
se_beta2_hat <- sqrt(s_squared * XtX_inv[3, 3])
se_beta3_hat <- sqrt(s_squared * XtX_inv[4, 4])

# Output standard errors of the regression coefficients
cat("Standard Error of beta0_hat:", se_beta0_hat, "\n")
```

```
## Standard Error of beta0_hat: 2.206615
```

```
cat("Standard Error of beta1_hat:", se_beta1_hat, "\n")
```

```
## Standard Error of beta1_hat: 0.5078363
```

```
cat("Standard Error of beta2_hat:", se_beta2_hat, "\n")
```

```
## Standard Error of beta2_hat: 0.2653825
```

```
cat("Standard Error of beta3_hat:", se_beta3_hat, "\n")
```

```
## Standard Error of beta3_hat: 0.01232893
```

Part c: Compare the standard error of each regression coefficient to the estimated regression coefficient and use the t test to test if each regression coefficient is equal to 0

H0: $\beta_i = 0$ for all i = 0, 1, 2, 3

```
# Compute t statistic for each regression coefficient
t_beta0 <- beta_hat[1] / se_beta0_hat
t_beta1 <- beta_hat[2] / se_beta1_hat
t_beta2 <- beta_hat[3] / se_beta2_hat
t_beta3 <- beta_hat[4] / se_beta3_hat

# Perform t test for each regression coefficient
p_value_beta0 <- 2 * (1 - pt(abs(t_beta0), df_E))
p_value_beta1 <- 2 * (1 - pt(abs(t_beta1), df_E))
p_value_beta2 <- 2 * (1 - pt(abs(t_beta2), df_E))
p_value_beta3 <- 2 * (1 - pt(abs(t_beta3), df_E))

# Create a decision column to determine if we reject the null hypothesis at alpha = 0.05
decision <- c(ifelse(p_value_beta0 < 0.05, "Reject H0", "Fail to Reject H0"),
              ifelse(p_value_beta1 < 0.05, "Reject H0", "Fail to Reject H0"),
              ifelse(p_value_beta2 < 0.05, "Reject H0", "Fail to Reject H0"),
              ifelse(p_value_beta3 < 0.05, "Reject H0", "Fail to Reject H0"))

# Output the coefficient, standard error, t statistic, p-value and decision for each regressi
on coefficient in a table
t_test_table <- data.frame(Coefficient = c("beta0", "beta1", "beta2", "beta3"),
                           Estimate = beta_hat,
                           Standard_Error = c(se_beta0_hat, se_beta1_hat, se_beta2_hat, se_b
eta3_hat),
                           t_statistic = c(t_beta0, t_beta1, t_beta2, t_beta3),
                           P_value = c(p_value_beta0, p_value_beta1, p_value_beta2, p_value_
beta3),
                           Decision = decision)



t_test_table
```

Coefficient <chr>	Estimate <dbl>	Standard_Error <dbl>	t_statistic <dbl>	P_value <dbl>	Decision <chr>
beta0	-1.163461	2.20661486	-0.52726057	0.6259022259	Fail to Reject H0
beta1	0.135270	0.50783629	0.26636536	0.8031249360	Fail to Reject H0
beta2	0.019950	0.26538248	0.07517452	0.9436853921	Fail to Reject H0
beta3	0.121954	0.01232893	9.89169181	0.0005861963	Reject H0

4 rows

Based on the t test results, we can see that the p-value for β3 is less than 0.05, so we reject the null hypothesis that β3 = 0. For the other coefficients, the p-values are greater than 0.05, so there is insufficient evidence to reject the null hypothesis that β0, β1, and β2 are equal to 0.

4.18. Consider the model

$$y = X\beta + \epsilon, \quad \epsilon \sim N(\mathbf{0}, \sigma^2 I)$$

Let $\hat{\beta} = (X'X)^{-1}X'y$, $\hat{\mu} = Hy$, and $e = (I - H)y$, where $H = X(X'X)^{-1}X'$. Show that $\hat{\mu}$ and e are statistically independent.

Given: $\hat{\underset{\sim}{\mu}} = H\underset{\sim}{y}$

$\underset{\sim}{e} = (I - H)\underset{\sim}{y}$

From Lecture Slides, we know that H and $(I-H)$ are idempotent.

Taking $H \cdot (I-H)$ gives $H - H^2 = H - H = 0$ (H and $(I-H)$ are orthogonal)

If $\hat{\underset{\sim}{\mu}}$ and $\underset{\sim}{e}$ are statistically independent $\Rightarrow$ they are uncorrelated $\Rightarrow$ $Corr(\hat{\underset{\sim}{\mu}}, \underset{\sim}{e}) = 0$

$\therefore\ Corr(\hat{\underset{\sim}{\mu}}, \underset{\sim}{e}) = \dfrac{Cov(\hat{\underset{\sim}{\mu}}, \underset{\sim}{e})}{\sqrt{Var(\hat{\underset{\sim}{\mu}})} \cdot \sqrt{Var(\underset{\sim}{e})}}$ → must $= 0$ for proof to hold

Hence: $Cov(\hat{\underset{\sim}{\mu}}, \underset{\sim}{e}) = E(\hat{\underset{\sim}{\mu}} \cdot \underset{\sim}{e}) - E(\hat{\underset{\sim}{\mu}})E(\underset{\sim}{e})$

$= E[H\underset{\sim}{y}(I-H)\underset{\sim}{y}] - E(H\underset{\sim}{y})E((I-H)\underset{\sim}{y})$

$= H(I-H)E(\underset{\sim}{y}'\underset{\sim}{y}) - H(I-H) \cdot E(\underset{\sim}{y}) \cdot E(\underset{\sim}{y})$

since $H(I-H) = 0$

$= 0$

Hence: $Cov(\hat{\underset{\sim}{\mu}}, \underset{\sim}{e}) = 0 \Rightarrow Corr(\hat{\underset{\sim}{\mu}}, \underset{\sim}{e}) = 0$

Since $\epsilon \sim N(\underset{\sim}{0}, \sigma^2 I)$ and so $\underset{\sim}{y}$ also follows a normal distribution

Thus, since $\hat{\underset{\sim}{\mu}}$ and $\underset{\sim}{e}$ are linear transformations of $\underset{\sim}{y}$ $\Rightarrow$ both $\hat{\underset{\sim}{\mu}}$ and $\underset{\sim}{e}$ are also normally distributed

$\Rightarrow$ Therefore ① $\hat{\underset{\sim}{\mu}}, \underset{\sim}{e}$ are normally distributed

② $corr(\hat{\underset{\sim}{\mu}}, \underset{\sim}{e}) = 0$ ($\hat{\underset{\sim}{\mu}}$ and $\underset{\sim}{e}$ are uncorrelated)

$\Rightarrow$ $\hat{\underset{\sim}{\mu}}$ and $\underset{\sim}{e}$ are statistically independent.