Home work-2

i) (a) Given to prove whether Quadratic loss function is convex or not and its usefulness in context of linear regression.

(a) Quadratic loss - $L = \sum_{i=1}^{N} e_i^2$

A function to be convex either one of these conditions should be passed:

Let's take for two points (x_1, y_1), (x_2, y_2) laying on function's curve then to be convex

$$\boxed{f(\lambda x_1 + (1-\lambda) x_2) \leq \lambda f(x_1) + (1-\lambda) f(x_2)}$$

$$\boxed{0 \leq \lambda \leq 1}.$$

Where λ denotes a point's location on a line and its value should be from 0 to 1.

The other way to check it's second derivative and check if its value is always bigger than 0

$$\boxed{\frac{d^2 f(x)}{dx^2} > 0.}$$

(a) for quadratic loss we should the double derivative of function ≥ 0.

$$\Rightarrow \quad L = \sum_{i=1}^{n} e_i^2$$

$$\frac{dL}{de} = \sum_{i=1}^{n} 2e_i$$

$\frac{d^2L}{de^2} = 2$ which is greater than 0 and hence proved the Quadratic function is convex function and can also be said strictly convex.

Use case - In linear regression, the quadratic loss function is widely employed as a loss function, with the purpose of minimizing the sum of the squared differences between the predicted and actual values of the response variable. minimising the quadratic loss is equivalent to finding the parameters of the linear regression model that best fit the data in least square sense. The quadratic loss is particularly useful when the errors are normally distributed with mean zero and constant variance.

2(b) The mean absolute error, which is related to the L_1 norm, is defined as $L = \Sigma |e_i|$

The mean absolute error is convex loss function. The second derivative of L with respect to e_i does not exist, since the absolute value function is not differential at zero.

The mean absolute error is a piecewise linear function and any linear function is a convex function. Therefore it is a convex loss function

proof -

$|e_i| = e_i$ if $e_i >= 0$

$|e_i| = -e_i$ if $e_i < 0$

Therefore we can write mean absolute error as

$L = \Sigma e_i$ for $e_i >= 0$

$L = \Sigma(-e_i)$ for $e_i < 0$.

each of these terms is a linear function of e_i, and any linear function is a convex function. Therefore the mean absolute error is a convex loss function

Usefulness - The mean absolute error is another popular loss function in linear regression. Minimizing the mean absolute error is equivalent to absolute differences between predicted and actual values of response variables The mean absolute error has the advantage of being more robust to outliers than quadratic loss.

1(c) The huber loss is a smoothe approximation which is less susuptible to outliers than the mean squared error, is approximated smoothly by huber loss.

$L = \Sigma l(ei)$

ei → error difference between the predicted and actual values..

$$L = \sum_{i=1}^{N} l(ei) \text{ where } l(e) = \begin{cases} \frac{1}{2}e^2 & \text{if } |e| \le \delta \\ \delta|e| - \frac{1}{2}\delta^2 & \text{if } |e| > \delta \end{cases}$$

we can define the function across both condition

$$L = \frac{1}{2}e^2 + \delta|e| - \frac{1}{2}\delta^2$$

$l(ei)$ becomes $L = \sum_{i=1}^{N} \frac{1}{2}ei^2 + \delta|ei| - \frac{1}{2}\delta^2$

$$\frac{dL}{dei} = \sum_{i=1}^{N} \frac{2}{2}ei + \frac{\delta(ei)}{|ei|} = 0 \quad \text{A } \delta \text{ is constant}$$

$$\frac{d^2L}{de^2} = \sum_{i=1}^{N} 1 + \frac{\delta\left[|ei| - \frac{ei}{|ei|} \times ei\right]}{|ei|^2}$$

$$= \sum_{i=1}^{N} \left[1 + \frac{\delta \left[\frac{e_i^2 - e_i^2}{|e_i|} \right]}{|e_i|^2} \right]$$

$$\sum_{i=1}^{N} [1+0] = N.$$

As the value of N is always positive this is a convex function.

<u>Usefulness in linear regression</u> -

The Huber loss, which strikes a balance b/w mean squared error and mean absolute error, is frequently applied to regression issues when the data may contain outliers. The huber loss can be made more or less sensitive to outliers by varying the value of . The huber loss resembles the mean squared error more when is larger. whereas mean absolute error more when is lower.

2a) Given linear regression $y_i = \theta_0 + \theta_1 x_i + e_i$ $i = 1, 2 \dots n$ minimizing the squared loss where e_i is random noise variable.

First we will calculate mean square error where error is ypred - yactual or vice versa

$$\text{Meansquare error} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

$$\frac{1}{n} \sum_{i=1}^{m} (y_i - (mx_i + c))^2$$

$$\Rightarrow \frac{1}{n} \sum_{i=1}^{n} \Sigma (y_i - \theta_0 - \theta_1 x_i)^2$$

y_i is the target variable and x_i is the input variable for the ith data point

gradient of the loss function

$$\nabla_\theta J(\theta) = \begin{bmatrix} \frac{\partial J(\theta)}{\partial \theta} \\ \frac{\partial J(\theta)}{\partial(\theta_1)} \end{bmatrix}$$

Scanned with OKEN Scanner

So we should calculate individual partial derivatives. we here chain rule

$$\frac{\partial J(\theta)}{\partial \theta_0} = \frac{1}{N}\sum_{i=1}^{N} 2(y_i - \theta_0 - \theta_1 x_i)(-1)$$

$$\frac{\partial J(\theta)}{\partial \theta_1} = \frac{1}{N}\sum_{i=1}^{N} 2(y_i - \theta_0 - \theta_1 x_i)(-x_i)$$

by simplifying we get

$$\frac{\partial J(\theta)}{\partial \theta_0} = \frac{-2}{N}\sum_{i=1}^{N}(y_0 - \theta_0 - \theta_1 x_i)$$

$$\frac{\partial J\theta}{\partial \theta_1} = \frac{-2}{N}\sum_{i=1}^{N} x_i(y_i - \theta_0 - \theta_1 x_i)$$

$\therefore$ the gradient of loss function wrt parameter vector is

$$\nabla_\theta (J(\theta)) = \begin{bmatrix} -2/N \sum_{i=1}^{N} (y_i - \theta_0 - \theta_1 x_i) \\ -2/N \sum_{i=1}^{N} x_i (y_i - \theta_0 - \theta_1 x_i) \end{bmatrix}$$

2(b) steps of batch gradient descent rule

1. intialize the parameters of the model randomly or with some predefined values

2. Define a cost function or loss function that needs to be minimized

3. Calculate the gradient of the cost function with respect to each parameter of the model

4. update the parameters using the following equation

parameter = parameter - learning rate * gradient

learning rate is the hyperparameter that determines the size of steps taken in direction of gradient.

5. Repeat 3 and 4 steps until convergence ie until the change of cost function becomes negligible.

* batch gradient descent rule compiles the gradient of the entire training data set update the parameters.

2(c) Steps of stochastic gradient descent rule.

1. Intialize the parameters of model randomly (or) with some predefined values.

2. Define a cost function (or) loss function that needs to minimize

3. shuffle the data randomly

4. for each sample in the shuffled training data, compute the gradient of the cost function with respect to

each parameter of the model

5 Update the parameters using following equations

parameter = parameter - learning rate * gradient.

learning rate is the hyper parameter that determines the size of steps taken in the direction of the gradient.

6. Repeat 4 and 5 steps for each sample in training data

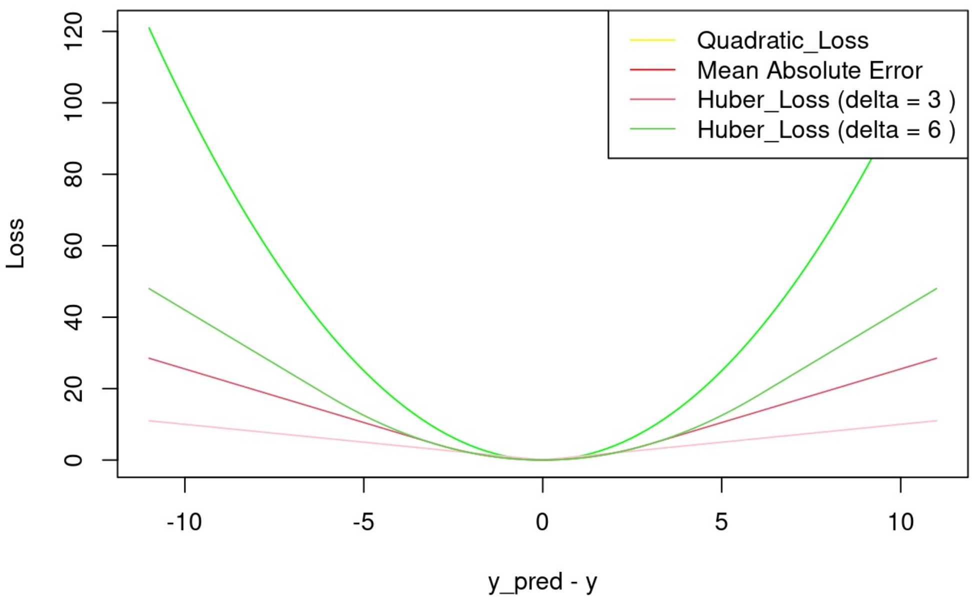
7. Repeat steps 3 to 6 for a fixed number of iterations or until convergence.

The stochastic gradient descent rule computes the gradient of one randomly selected sample at a time to update the parameters.

R Notebook

Gowreesh Gunupati

```
# Q 3a)

#implemented Quadratic Loss (L2 Norm)

quadratic_loss <- function(y, y_pred){
  sum((y - y_pred) ^ 2)
}

#implemented Mean Absolute Error (L1 Norm)

mean_absolute_error <- function(y, y_pred) {
  sum(abs(y - y_pred))
}

#implemented Huber Loss, delta

huberloss <- function(y, y_pred, d) {
  delta <- d
  e <- y - y_pred
  if (abs(e) <= delta) {
    0.5 * (e^2)
  } else {
    delta * (abs(e) - 0.5*(delta))
  }
}

# implement x-values

x <- seq(-11, 11, length.out = 99)

# implement delta values

del_vals <- c(3, 6)

# Calculate y-values for each function and delta value

y_quadratic <- sapply(x, function(x) quadratic_loss(0, x))

y_mean_absolute_error <- sapply(x, function(x) mean_absolute_error(0, x))

y_huber <- lapply(del_vals, function(delta) sapply(x, function(x) huberloss(0, x, delta)))

# Plot the lines for each function and delta value

plot(x, y_quadratic, type = "l", col = "green", xlab = "y_pred - y", ylab = "Loss")

lines(x, y_mean_absolute_error, col = "pink")
for (i in seq_along(del_vals)) {
  lines(x, y_huber[[i]], col = i+1)
}

legend("topright", legend = c("Quadratic_Loss", "Mean Absolute Error", paste("Huber_Loss (delta =", del_vals, ")"
)),
       col = c("yellow", "red", 2:(2 + length(del_vals) - 1)), lty = 1)
```


Quadratic_Loss
Mean Absolute Error
Huber_Loss (delta = 3)
Huber_Loss (delta = 6)
Loss
0
20
40
60
80
100
120
-10
-5
0
5
10
y_pred - y

```r
#Q 3b Quadratic_Loss Batch Gradient Descent

# implement the quadratic loss function

quadratic_loss <- function(y, y_pred) {
  return(0.5 * mean((y - y_pred)^2))
}

# implement the gradient of the quadratic loss function

quadratic_loss_gradient <- function(y, y_pred) {
  return(y_pred - y)
}

# implement the batch gradient descent function

batch_gradient_descent <- function(X, y, learning_rate, num_iterations) {
  # Initialize weights and bias to zero
  w <- matrix(0, nrow=ncol(X), ncol=1)
  b <- 0

  # Loop over the specified number of iterations

  for (i in 1:num_iterations) {
    # computing the predictions using the current weights and bias

    y_pred <- X %*% w + b

    # computing the loss and its gradient

    loss <- quadratic_loss(y, y_pred)
    gradient_w <- t(X) %*% quadratic_loss_gradient(y, y_pred)
    gradient_b <- sum(quadratic_loss_gradient(y, y_pred))

    # Updating the weights and bias using the computingd gradients

    w <- w - learning_rate * gradient_w
    b <- b - learning_rate * gradient_b

    # Print the loss every 99 iterations

    if (i %% 99 == 0) {
      cat(sprintf("Iteration %d, loss = %f\n", i, loss))
    }
  }

  # Return the final weights and bias

  return(list(w=w, b=b))
}

# Generate some sample data

set.seed(240)
n <- 99
p <- 11
X <- matrix(rnorm(n * p), nrow=n, ncol=p)
y <- X %*% rnorm(p) + rnorm(n, sd=0.5)

# Run batch gradient descent with quadratic loss

result <- batch_gradient_descent(X, y, learning_rate=0.001, num_iterations=1000)
```

```
## Iteration 99, loss = 0.110439
## Iteration 198, loss = 0.110438
## Iteration 297, loss = 0.110438
## Iteration 396, loss = 0.110438
## Iteration 495, loss = 0.110438
## Iteration 594, loss = 0.110438
## Iteration 693, loss = 0.110438
## Iteration 792, loss = 0.110438
## Iteration 891, loss = 0.110438
## Iteration 990, loss = 0.110438
```

```r
# Print the learned weights and bias

cat(sprintf("Learned weights: %s\n", as.character(result$w)))
```

```
## Learned weights: 0.684265688495488
##  Learned weights: -1.72409166941043
##  Learned weights: 0.334780983835761
##  Learned weights: 0.190533831222232
##  Learned weights: 1.27127915088966
##  Learned weights: 0.272325964654812
##  Learned weights: -0.577513728268303
##  Learned weights: 0.278817448864219
##  Learned weights: 2.16493234044153
##  Learned weights: -0.45395010920518
##  Learned weights: 0.475888143661629
```

```r
cat(sprintf("Learned bias: %f\n", result$b))
```

```
## Learned bias: -0.006225
```

```
#Q 3c Quadratic Loss Stochastic Gradient Descent

# Generate some sample data

set.seed(240)
x <- runif(99, 0, 11)
y <- 2 * x + rnorm(99)

# Set learning rate and number of epochs

learning_rate <- 0.01
epochs <- 99

# Initialize coefficients

w0 <- runif(1, 0, 1)
w1 <- runif(1, 0, 1)


# Redefining the quadratic loss function for stochastic gradient descent

loss <- function(y_pred, y) {
  return((y_pred - y)^2)
}

# implement the stochastic gradient descent function

stochastic_gd <- function(x, y, w0, w1, learning_rate, epochs, loss) {
  for (i in 1:epochs) {

    # Select a random data point

    index <- sample(1:length(x), 1)
    x_i <- x[index]
    y_i <- y[index]

    # computing the predicted value and the error

    y_pred <- w0 + w1 * x_i
    error <- y_pred - y_i

    # Updating the coefficients

    w0 <- w0 - learning_rate * error
    w1 <- w1 - learning_rate * error * x_i

    # computing and print the loss

    loss_i <- loss(y_pred, y_i)
    cat("Epoch:", i, "Loss:", loss_i, "\n")
  }

  # Return the final coefficients

  return(c(w0, w1))
}

# Call the stochastic_gd function and print the final coefficients

coefficients <- stochastic_gd(x, y, w0, w1, learning_rate, epochs, loss)
```

```
## Epoch: 1 Loss: 26.18205
## Epoch: 2 Loss: 6.935858
## Epoch: 3 Loss: 18.85495
## Epoch: 4 Loss: 1.386032
## Epoch: 5 Loss: 1.307735
## Epoch: 6 Loss: 28.82946
## Epoch: 7 Loss: 7.57542
## Epoch: 8 Loss: 1.305584
## Epoch: 9 Loss: 4.242676
## Epoch: 10 Loss: 0.001317783
## Epoch: 11 Loss: 2.165961
## Epoch: 12 Loss: 2.012126
## Epoch: 13 Loss: 5.162982
## Epoch: 14 Loss: 0.01106934
## Epoch: 15 Loss: 2.559979
## Epoch: 16 Loss: 0.1774025
```

```
## Epoch: 17 Loss: 0.728834
## Epoch: 18 Loss: 0.1198967
## Epoch: 19 Loss: 0.444151
## Epoch: 20 Loss: 2.781822
## Epoch: 21 Loss: 0.01273339
## Epoch: 22 Loss: 0.2529201
## Epoch: 23 Loss: 0.8332565
## Epoch: 24 Loss: 0.07963704
## Epoch: 25 Loss: 2.990129
## Epoch: 26 Loss: 0.3654734
## Epoch: 27 Loss: 1.750165
## Epoch: 28 Loss: 0.07297894
## Epoch: 29 Loss: 1.497567
## Epoch: 30 Loss: 2.898738
## Epoch: 31 Loss: 1.281208
## Epoch: 32 Loss: 0.2137889
## Epoch: 33 Loss: 5.573137
## Epoch: 34 Loss: 1.183862
## Epoch: 35 Loss: 0.217685
## Epoch: 36 Loss: 0.009620644
## Epoch: 37 Loss: 2.289029
## Epoch: 38 Loss: 3.763673
## Epoch: 39 Loss: 1.850005
## Epoch: 40 Loss: 1.043849
## Epoch: 41 Loss: 3.156817
## Epoch: 42 Loss: 1.363095
## Epoch: 43 Loss: 3.401006
## Epoch: 44 Loss: 6.863369
## Epoch: 45 Loss: 3.790352
## Epoch: 46 Loss: 15.91991
## Epoch: 47 Loss: 0.4654995
## Epoch: 48 Loss: 0.2151234
## Epoch: 49 Loss: 4.902011
## Epoch: 50 Loss: 0.001700726
## Epoch: 51 Loss: 3.794054
## Epoch: 52 Loss: 0.3128379
## Epoch: 53 Loss: 7.014309
## Epoch: 54 Loss: 1.090968
## Epoch: 55 Loss: 1.547928
## Epoch: 56 Loss: 1.607968
## Epoch: 57 Loss: 0.001123673
## Epoch: 58 Loss: 0.04655177
## Epoch: 59 Loss: 0.005107004
## Epoch: 60 Loss: 6.260269
## Epoch: 61 Loss: 1.985161
## Epoch: 62 Loss: 0.05921465
## Epoch: 63 Loss: 0.07777627
## Epoch: 64 Loss: 0.01884548
## Epoch: 65 Loss: 0.02006302
## Epoch: 66 Loss: 3.808302
## Epoch: 67 Loss: 0.1160004
## Epoch: 68 Loss: 1.154414
## Epoch: 69 Loss: 0.1987237
## Epoch: 70 Loss: 1.511457
## Epoch: 71 Loss: 0.03775612
## Epoch: 72 Loss: 4.102683
## Epoch: 73 Loss: 2.022195
## Epoch: 74 Loss: 0.9563194
## Epoch: 75 Loss: 0.117095
## Epoch: 76 Loss: 0.4127148
## Epoch: 77 Loss: 0.3058775
## Epoch: 78 Loss: 0.01371048
## Epoch: 79 Loss: 0.0545029
## Epoch: 80 Loss: 1.109487
## Epoch: 81 Loss: 3.906092
## Epoch: 82 Loss: 0.1985293
## Epoch: 83 Loss: 0.04925679
## Epoch: 84 Loss: 0.130149
## Epoch: 85 Loss: 0.1856741
## Epoch: 86 Loss: 0.1620717
## Epoch: 87 Loss: 0.1188139
## Epoch: 88 Loss: 0.2675432
## Epoch: 89 Loss: 1.596756
## Epoch: 90 Loss: 2.357324
## Epoch: 91 Loss: 0.1119586
## Epoch: 92 Loss: 0.3739728
## Epoch: 93 Loss: 0.5947721
## Epoch: 94 Loss: 2.767201
## Epoch: 95 Loss: 0.0493028
```

```
## Epoch: 96 Loss: 0.03022918
## Epoch: 97 Loss: 1.847188
## Epoch: 98 Loss: 0.3347333
## Epoch: 99 Loss: 0.003588998
```

```
cat("Final coefficients:", coefficients[1], coefficients[2], "\n")
```

```
## Final coefficients: 0.9173339 1.919696
```

```
#Q 3b Mean Absolute Error Loss Batch Gradient Descent

# implement the mean absolute error function

mean_absolute_error_loss <- function(y, y_pred) {
  return(mean(abs(y - y_pred)))
}

# implement the gradient of the mean absolute error function

mean_absolute_error_loss_gradient <- function(y, y_pred) {
  return(ifelse(y > y_pred, -1, 1))
}

# implement the batch gradient descent function

batch_gradient_descent_mean_absolute_error <- function(X, y, learning_rate, num_iterations) {
  # Initialize weights and bias to zero
  w <- matrix(0, nrow=ncol(X), ncol=1)
  b <- 0

  # Loop over the specified number of iterations

  for (i in 1:num_iterations) {
    # computing the predictions using the current weights and bias

    y_pred <- X %*% w + b

    # computing the loss and its gradient

    loss <- mean_absolute_error_loss(y, y_pred)
    gradient_w <- t(X) %*% mean_absolute_error_loss_gradient(y, y_pred)
    gradient_b <- sum(mean_absolute_error_loss_gradient(y, y_pred))

    # Updating the weights and bias using the computingd gradients

    w <- w - learning_rate * gradient_w
    b <- b - learning_rate * gradient_b

    # Printing the loss every 99 iterations

    if (i %% 99 == 0) {
      cat(sprintf("Iteration %d, loss = %f\n", i, loss))
    }
  }

  # Return the final weights and bias

  return(list(w=w, b=b))
}
```

```
# Generate some sample data

set.seed(240)
n <- 99
p <- 11
X <- matrix(rnorm(n * p), nrow=n, ncol=p)
y <- X %*% rnorm(p) + rnorm(n, sd=0.5)

# Run batch gradient descent with mean absolute error

result <- batch_gradient_descent(X, y, learning_rate=0.001, num_iterations=1000)
```

```
## Iteration 99, loss = 0.110439
## Iteration 198, loss = 0.110438
## Iteration 297, loss = 0.110438
## Iteration 396, loss = 0.110438
## Iteration 495, loss = 0.110438
## Iteration 594, loss = 0.110438
## Iteration 693, loss = 0.110438
## Iteration 792, loss = 0.110438
## Iteration 891, loss = 0.110438
## Iteration 990, loss = 0.110438
```

```
# Print the learned weights and bias

cat(sprintf("Learned weights: %s\n", as.character(result$w)))
```

```
## Learned weights: 0.684265688495488
##  Learned weights: -1.72409166941043
##  Learned weights: 0.334780983835761
##  Learned weights: 0.190533831222232
##  Learned weights: 1.27127915088966
##  Learned weights: 0.272325964654812
##  Learned weights: -0.577513728268303
##  Learned weights: 0.278817448864219
##  Learned weights: 2.16493234044153
##  Learned weights: -0.45395010920518
##  Learned weights: 0.475888143661629
```

```
cat(sprintf("Learned bias: %f\n", result$b))
```

```
## Learned bias: -0.006225
```

```r
#Q 3c Mean Absolute Error Loss Stochastic Gradient Descent

# Generate some sample data

set.seed(240)
x <- runif(99, 0, 11)
y <- 2 * x + rnorm(99)

# Set learning rate and number of epochs

learning_rate <- 0.01
epochs <- 99

# Initialize coefficients

w0 <- runif(1, 0, 1)
w1 <- runif(1, 0, 1)

# implement the mean absolute error loss function

loss <- function(y_pred, y) {
  return(abs(y_pred - y))
}

# implement the stochastic gradient descent function

stochastic_gd <- function(x, y, w0, w1, learning_rate, epochs, loss) {
  for (i in 1:epochs) {
    # Select a random data point
    index <- sample(1:length(x), 1)
    x_i <- x[index]
    y_i <- y[index]

    # computing the predicted value and the error

    y_pred <- w0 + w1 * x_i
    error <- y_pred - y_i

    # Updating the coefficients

    if (error > 0) {
      w0 <- w0 - learning_rate
      w1 <- w1 - learning_rate * x_i
    } else if (error < 0) {
      w0 <- w0 + learning_rate
      w1 <- w1 + learning_rate * x_i
    }

    # computing and print the loss

    loss_i <- loss(y_pred, y_i)
    cat("Epoch:", i, "Loss:", loss_i, "\n")
  }

  # Return the final coefficients

  return(c(w0, w1))
}

# Call the stochastic_gd function and print the final coefficients

coefficients <- stochastic_gd(x, y, w0, w1, learning_rate, epochs, loss)
```

```
## Epoch: 1 Loss: 5.11684
## Epoch: 2 Loss: 3.201255
## Epoch: 3 Loss: 5.380958
## Epoch: 4 Loss: 1.769249
## Epoch: 5 Loss: 1.740593
## Epoch: 6 Loss: 8.216935
## Epoch: 7 Loss: 6.222266
## Epoch: 8 Loss: 4.223126
## Epoch: 9 Loss: 9.220364
## Epoch: 10 Loss: 1.33198
## Epoch: 11 Loss: 6.972149
## Epoch: 12 Loss: 4.167625
## Epoch: 13 Loss: 1.762121
## Epoch: 14 Loss: 4.169817
```

```
## Epoch: 15 Loss: 5.280339
## Epoch: 16 Loss: 5.933908
## Epoch: 17 Loss: 3.66722
## Epoch: 18 Loss: 3.130474
## Epoch: 19 Loss: 2.415732
## Epoch: 20 Loss: 3.146818
## Epoch: 21 Loss: 0.6828873
## Epoch: 22 Loss: 1.403082
## Epoch: 23 Loss: 1.148057
## Epoch: 24 Loss: 0.8240276
## Epoch: 25 Loss: 1.921041
## Epoch: 26 Loss: 0.05528238
## Epoch: 27 Loss: 1.389899
## Epoch: 28 Loss: 0.06551079
## Epoch: 29 Loss: 1.780733
## Epoch: 30 Loss: 1.167325
## Epoch: 31 Loss: 0.9164181
## Epoch: 32 Loss: 0.4112307
## Epoch: 33 Loss: 2.291833
## Epoch: 34 Loss: 0.7597662
## Epoch: 35 Loss: 0.1001778
## Epoch: 36 Loss: 0.7654397
## Epoch: 37 Loss: 1.705447
## Epoch: 38 Loss: 1.332586
## Epoch: 39 Loss: 1.312131
## Epoch: 40 Loss: 1.037363
## Epoch: 41 Loss: 1.758273
## Epoch: 42 Loss: 1.170304
## Epoch: 43 Loss: 1.724532
## Epoch: 44 Loss: 1.813533
## Epoch: 45 Loss: 2.287857
## Epoch: 46 Loss: 2.805608
## Epoch: 47 Loss: 0.5883196
## Epoch: 48 Loss: 0.7280578
## Epoch: 49 Loss: 2.059164
## Epoch: 50 Loss: 0.3602608
## Epoch: 51 Loss: 2.124909
## Epoch: 52 Loss: 1.412058
## Epoch: 53 Loss: 2.506205
## Epoch: 54 Loss: 0.8819522
## Epoch: 55 Loss: 1.381605
## Epoch: 56 Loss: 0.8840212
## Epoch: 57 Loss: 0.05012445
## Epoch: 58 Loss: 0.5785105
## Epoch: 59 Loss: 0.4352524
## Epoch: 60 Loss: 3.239986
## Epoch: 61 Loss: 0.4731195
## Epoch: 62 Loss: 1.226675
## Epoch: 63 Loss: 1.902509
## Epoch: 64 Loss: 0.8363992
## Epoch: 65 Loss: 0.1901623
## Epoch: 66 Loss: 1.510528
## Epoch: 67 Loss: 0.7740541
## Epoch: 68 Loss: 1.049984
## Epoch: 69 Loss: 0.4693508
## Epoch: 70 Loss: 1.683864
## Epoch: 71 Loss: 0.4997212
## Epoch: 72 Loss: 2.334763
## Epoch: 73 Loss: 0.8358674
## Epoch: 74 Loss: 1.374188
## Epoch: 75 Loss: 0.6258258
## Epoch: 76 Loss: 0.8238019
## Epoch: 77 Loss: 0.2937455
## Epoch: 78 Loss: 0.01416117
## Epoch: 79 Loss: 0.222946
## Epoch: 80 Loss: 0.4458417
## Epoch: 81 Loss: 2.032024
## Epoch: 82 Loss: 0.4582846
## Epoch: 83 Loss: 0.3624989
## Epoch: 84 Loss: 0.2380425
## Epoch: 85 Loss: 0.808055
## Epoch: 86 Loss: 0.3194402
## Epoch: 87 Loss: 0.06907352
## Epoch: 88 Loss: 0.1868126
## Epoch: 89 Loss: 1.822473
## Epoch: 90 Loss: 0.7811836
## Epoch: 91 Loss: 0.1233316
## Epoch: 92 Loss: 0.6041156
## Epoch: 93 Loss: 0.9172902
```

```
## Epoch: 94 Loss: 1.449845
## Epoch: 95 Loss: 0.3775952
## Epoch: 96 Loss: 0.2679533
## Epoch: 97 Loss: 0.7789746
## Epoch: 98 Loss: 1.015268
## Epoch: 99 Loss: 0.1188147
```

```
cat("Final coefficients:", coefficients[1], coefficients[2], "\n")
```

```
## Final coefficients: 0.8992389 2.004871
```

```
#Q 3b Huber Loss Batch Gradient Descent

# implement the Huber loss function

huber_loss <- function(y, y_pred, delta) {
  abs_diff <- abs(y - y_pred)
  return(ifelse(abs_diff <= delta, 0.5 * abs_diff^2, delta * (abs_diff - 0.5 * delta)))
}

# implement the gradient of the Huber loss function

huber_loss_gradient <- function(y, y_pred, delta) {
  abs_diff <- abs(y - y_pred)
  return(ifelse(abs_diff <= delta, y_pred - y, delta * sign(y_pred - y)))
}

# implement the batch gradient descent function

batch_gradient_descent <- function(X, y, learning_rate, num_iterations, delta) {
  # Initialize weights and bias to zero
  w <- matrix(0, nrow=ncol(X), ncol=1)
  b <- 0

  # Loop over the specified number of iterations

  for (i in 1:num_iterations) {
    # computing the predictions using the current weights and bias

    y_pred <- X %*% w + b

    # computing the loss and its gradient

    loss <- sum(huber_loss(y, y_pred, delta))
    gradient_w <- t(X) %*% huber_loss_gradient(y, y_pred, delta)
    gradient_b <- sum(huber_loss_gradient(y, y_pred, delta))

    # Updating the weights and bias using the computing gradients

    w <- w - learning_rate * gradient_w
    b <- b - learning_rate * gradient_b

    # Print the loss every 99 iterations

    if (i %% 99 == 0) {
      cat(sprintf("Iteration %d, loss = %f\n", i, loss))
    }
  }

  # Return the final weights and bias

  return(list(w=w, b=b))
}
```

```
# Generate some sample data

set.seed(240)
n <- 99
p <- 11
X <- matrix(rnorm(n * p), nrow=n, ncol=p)
y <- X %*% rnorm(p) + rnorm(n, sd=0.5)

# Run batch gradient descent with Huber loss

result <- batch_gradient_descent(X, y, learning_rate=0.001, num_iterations=1000, delta=1.0)
```

```
## Iteration 99, loss = 10.892480
## Iteration 198, loss = 10.890754
## Iteration 297, loss = 10.890754
## Iteration 396, loss = 10.890754
## Iteration 495, loss = 10.890754
## Iteration 594, loss = 10.890754
## Iteration 693, loss = 10.890754
## Iteration 792, loss = 10.890754
## Iteration 891, loss = 10.890754
## Iteration 990, loss = 10.890754
```

```
# Print the learned weights and bias

cat(sprintf("Learned weights: %s\n", as.character(result$w)))
```

```
## Learned weights: 0.685944178369784
##  Learned weights: -1.72372068391087
##  Learned weights: 0.333353519374068
##  Learned weights: 0.188446078495092
##  Learned weights: 1.27051245038548
##  Learned weights: 0.270929470018464
##  Learned weights: -0.575230100430467
##  Learned weights: 0.279947852324893
##  Learned weights: 2.16466930561089
##  Learned weights: -0.451796910672755
##  Learned weights: 0.477031355469777
```

```
cat(sprintf("Learned bias: %f\n", result$b))
```

```
## Learned bias: -0.005722
```

```r
#Q 3c Huber Loss Stochastic Gradient Descent

# Generate some sample data

set.seed(240)
x <- runif(99, 0, 11)
y <- 2 * x + rnorm(99)

# Set learning rate, delta value, and number of epochs

learning_rate <- 0.01
delta <- 1
epochs <- 99

# Initialize coefficients

w0 <- runif(1, 0, 1)
w1 <- runif(1, 0, 1)

# implement the Huber loss function

huber_loss <- function(y_pred, y, delta) {
  abs_error <- abs(y_pred - y)
  if (abs_error <= delta) {
    return(0.5 * abs_error^2)
  } else {
    return(delta * abs_error - 0.5 * delta^2)
  }
}

# implement the derivative of the Huber loss function

huber_loss_deriv <- function(y_pred, y, delta) {
  abs_error <- abs(y_pred - y)
  if (abs_error <= delta) {
    return(abs_error)
  } else {
    return(delta)
  }
}

# implement the stochastic gradient descent function

stochastic_gd <- function(x, y, w0, w1, learning_rate, epochs, loss, loss_deriv, delta) {
  for (i in 1:epochs) {
    # Select a random data point
    index <- sample(1:length(x), 1)
    x_i <- x[index]
    y_i <- y[index]

    # computing the predicted value and the error

    y_pred <- w0 + w1 * x_i
    error <- y_pred - y_i

    # computing the gradient of the loss function

    loss_grad <- loss_deriv(y_pred, y_i, delta)

    # Updating the coefficients

    w0 <- w0 - learning_rate * loss_grad
    w1 <- w1 - learning_rate * loss_grad * x_i

    # computing and print the loss

    loss_i <- loss(y_pred, y_i, delta)
    cat("Epoch:", i, "Loss:", loss_i, "\n")
  }

  # Return the final coefficients

  return(c(w0, w1))
}

# Call the stochastic_gd function and print the final coefficients

coefficients <- stochastic_gd(x, y, w0, w1, learning_rate, epochs, huber_loss, huber_loss_deriv, delta)
```

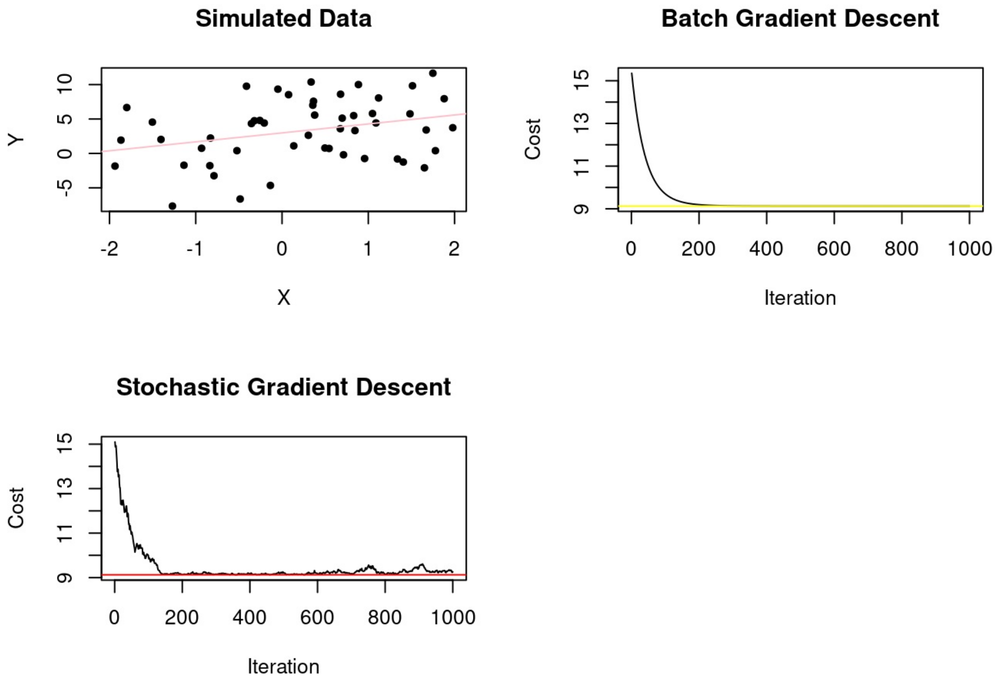
```
## Epoch: 1 Loss: 4.61684
## Epoch: 2 Loss: 2.977026
## Epoch: 3 Loss: 5.583217
## Epoch: 4 Loss: 1.651272
## Epoch: 5 Loss: 1.679928
## Epoch: 6 Loss: 9.971055
## Epoch: 7 Loss: 7.944896
## Epoch: 8 Loss: 5.869825
## Epoch: 9 Loss: 14.32241
## Epoch: 10 Loss: 2.088541
## Epoch: 11 Loss: 14.48419
## Epoch: 12 Loss: 8.082088
## Epoch: 13 Loss: 0.3971415
## Epoch: 14 Loss: 10.99533
## Epoch: 15 Loss: 12.72124
## Epoch: 16 Loss: 17.72546
## Epoch: 17 Loss: 15.6247
## Epoch: 18 Loss: 17.02006
## Epoch: 19 Loss: 12.82061
## Epoch: 20 Loss: 13.0847
## Epoch: 21 Loss: 5.346174
## Epoch: 22 Loss: 7.507688
## Epoch: 23 Loss: 17.27415
## Epoch: 24 Loss: 20.48686
## Epoch: 25 Loss: 26.84823
## Epoch: 26 Loss: 15.29162
## Epoch: 27 Loss: 1.841095
## Epoch: 28 Loss: 6.939218
## Epoch: 29 Loss: 19.76523
## Epoch: 30 Loss: 14.02786
## Epoch: 31 Loss: 8.328557
## Epoch: 32 Loss: 0.6783433
## Epoch: 33 Loss: 0.003873385
## Epoch: 34 Loss: 17.94776
## Epoch: 35 Loss: 23.93101
## Epoch: 36 Loss: 22.22067
## Epoch: 37 Loss: 30.20247
## Epoch: 38 Loss: 26.79713
## Epoch: 39 Loss: 14.42279
## Epoch: 40 Loss: 11.96992
## Epoch: 41 Loss: 2.936401
## Epoch: 42 Loss: 3.211521
## Epoch: 43 Loss: 35.47117
## Epoch: 44 Loss: 37.42626
## Epoch: 45 Loss: 9.294467
## Epoch: 46 Loss: 33.51535
## Epoch: 47 Loss: 39.48287
## Epoch: 48 Loss: 18.86277
## Epoch: 49 Loss: 25.3546
## Epoch: 50 Loss: 28.39302
## Epoch: 51 Loss: 33.22757
## Epoch: 52 Loss: 33.94042
## Epoch: 53 Loss: 12.37053
## Epoch: 54 Loss: 30.21421
## Epoch: 55 Loss: 28.24979
## Epoch: 56 Loss: 44.97484
## Epoch: 57 Loss: 36.52734
## Epoch: 58 Loss: 22.77594
## Epoch: 59 Loss: 51.8425
## Epoch: 60 Loss: 49.24589
## Epoch: 61 Loss: 20.65194
## Epoch: 62 Loss: 22.06239
## Epoch: 63 Loss: 43.79034
## Epoch: 64 Loss: 31.00812
## Epoch: 65 Loss: 6.807593
## Epoch: 66 Loss: 52.02721
## Epoch: 67 Loss: 42.17542
## Epoch: 68 Loss: 23.94434
## Epoch: 69 Loss: 47.49644
## Epoch: 70 Loss: 53.55521
## Epoch: 71 Loss: 56.25509
## Epoch: 72 Loss: 33.94474
## Epoch: 73 Loss: 30.61266
## Epoch: 74 Loss: 24.50549
## Epoch: 75 Loss: 59.88579
## Epoch: 76 Loss: 20.45615
## Epoch: 77 Loss: 43.52726
## Epoch: 78 Loss: 8.514611
```

```
## Epoch: 79 Loss: 53.54953
## Epoch: 80 Loss: 62.06003
## Epoch: 81 Loss: 7.083066
## Epoch: 82 Loss: 2.351993
## Epoch: 83 Loss: 14.72572
## Epoch: 84 Loss: 19.24321
## Epoch: 85 Loss: 36.02241
## Epoch: 86 Loss: 19.51696
## Epoch: 87 Loss: 51.40673
## Epoch: 88 Loss: 36.99879
## Epoch: 89 Loss: 44.35048
## Epoch: 90 Loss: 48.0249
## Epoch: 91 Loss: 48.88018
## Epoch: 92 Loss: 2.382391
## Epoch: 93 Loss: 28.91057
## Epoch: 94 Loss: 57.1285
## Epoch: 95 Loss: 33.86431
## Epoch: 96 Loss: 70.06557
## Epoch: 97 Loss: 53.46373
## Epoch: 98 Loss: 67.13954
## Epoch: 99 Loss: 68.036
```

```
cat("Final coefficients:", coefficients[1], coefficients[2], "\n")
```

```
## Final coefficients: -0.2705535 -5.328012
```

```r
# 4a

# Simulate data

set.seed(240)  # for reproducibility
N <- 50
X <- runif(N, -2, 2)
e <- rnorm(N, mean = 0, sd = 4)
Y <- 3 + 2*X + e

# implement the cost function

cost <- function(beta, X, Y) {
  sum((Y - beta[1] - beta[2]*X)^2)/(2*length(Y))
}

# Analytical_solution solution

Xmat <- matrix(c(rep(1, N), X), ncol = 2)
beta_Analytical_solution <- solve(t(Xmat) %*% Xmat) %*% t(Xmat) %*% Y
cost_Analytical_solution <- cost(beta_Analytical_solution, X, Y)

# Batch gradient descent

alpha <- 0.01
beta_batch <- c(0, 0)
niter <- 1000
cost_batch <- numeric(niter)
for (i in 1:niter) {
  grad <- c(-sum(Y - beta_batch[1] - beta_batch[2]*X)/length(Y),
            -sum(X*(Y - beta_batch[1] - beta_batch[2]*X))/length(Y))
  beta_batch <- beta_batch - alpha*grad
  cost_batch[i] <- cost(beta_batch, X, Y)
}

# Stochastic gradient descent

set.seed(456)  # for reproducibility
beta_stochastic <- c(0, 0)
cost_stochastic <- numeric(niter)
for (i in 1:niter) {
  index <- sample(N, 1)
  grad <- c(-(Y[index] - beta_stochastic[1] - beta_stochastic[2]*X[index]),
            -X[index]*(Y[index] - beta_stochastic[1] - beta_stochastic[2]*X[index]))
  beta_stochastic <- beta_stochastic - alpha*grad
  cost_stochastic[i] <- cost(beta_stochastic, X, Y)
}

# Plot the results

par(mfrow = c(2, 2))
plot(X, Y, pch = 20, xlab = "X", ylab = "Y", main = "Simulated Data")
abline(beta_Analytical_solution[1], beta_Analytical_solution[2], col = "pink")
plot(cost_batch, type = "l", xlab = "Iteration", ylab = "Cost",
     main = "Batch Gradient Descent")
abline(h = cost_Analytical_solution, col = "yellow")
plot(cost_stochastic, type = "l", xlab = "Iteration", ylab = "Cost",
     main = "Stochastic Gradient Descent")
abline(h = cost_Analytical_solution, col = "red")
```




```r
# 4B

set.seed(240)  # for reproducibility
N <- 50
niter <- 1000
alpha <- 0.01
true_beta <- 2
beta_Analytical_solution <- numeric(niter)
beta_batch <- numeric(niter)
beta_stochastic <- numeric(niter)

for (i in 1:niter) {
  # Simulate data
  X <- runif(N, -2, 2)
  e <- rnorm(N, mean = 0, sd = 4)
  Y <- 3 + true_beta*X + e

  # Analytical_solution solution

  Xmat <- matrix(c(rep(1, N), X), ncol = 2)
  beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat) %*% Y

  # Batch gradient descent

  beta_batch_i <- c(0, 0)
  for (j in 1:niter) {
    grad <- c(-sum(Y - beta_batch_i[1] - beta_batch_i[2]*X)/length(Y),
              -sum(X*(Y - beta_batch_i[1] - beta_batch_i[2]*X))/length(Y))
    beta_batch_i <- beta_batch_i - alpha*grad
  }
  beta_batch[i] <- beta_batch_i[2]

  # Stochastic gradient descent

  set.seed(i)  # for reproducibility
  beta_stochastic_i <- c(0, 0)
  for (j in 1:niter) {
    index <- sample(N, 1)
    grad <- c(-(Y[index] - beta_stochastic_i[1] - beta_stochastic_i[2]*X[index]),
              -X[index]*(Y[index] - beta_stochastic_i[1] - beta_stochastic_i[2]*X[index]))
    beta_stochastic_i <- beta_stochastic_i - alpha*grad
  }
  beta_stochastic[i] <- beta_stochastic_i[2]
}
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
```

```
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
```

```
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
```

```
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
```

```
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
```

```
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
```

```
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
```

```
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
```

```
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
```

```
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
```

```
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
```

```
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

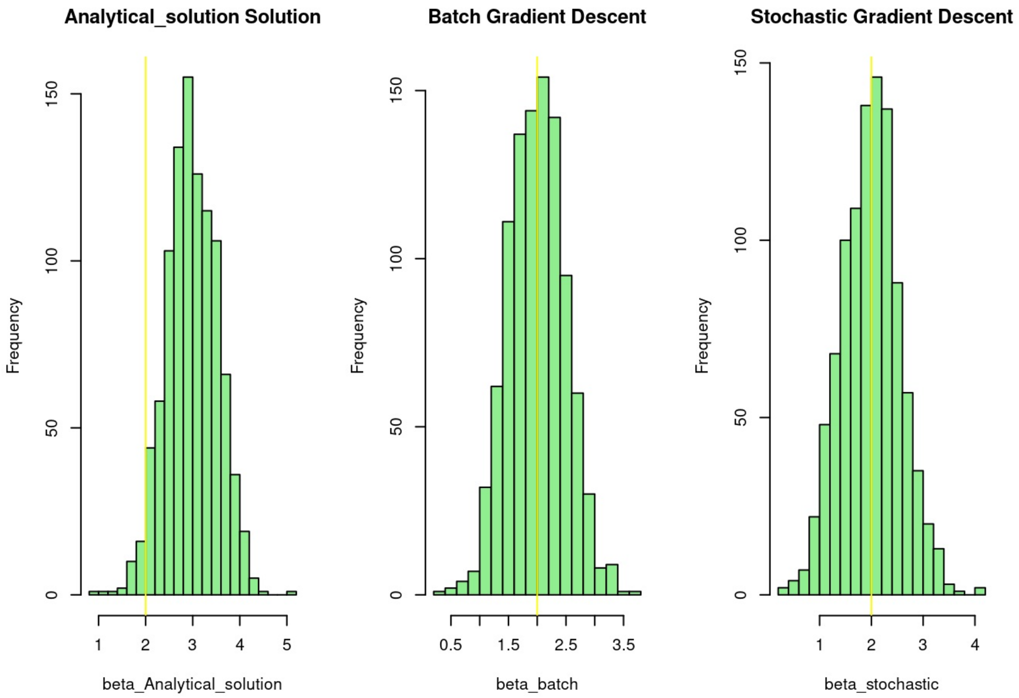
```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
```

```
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

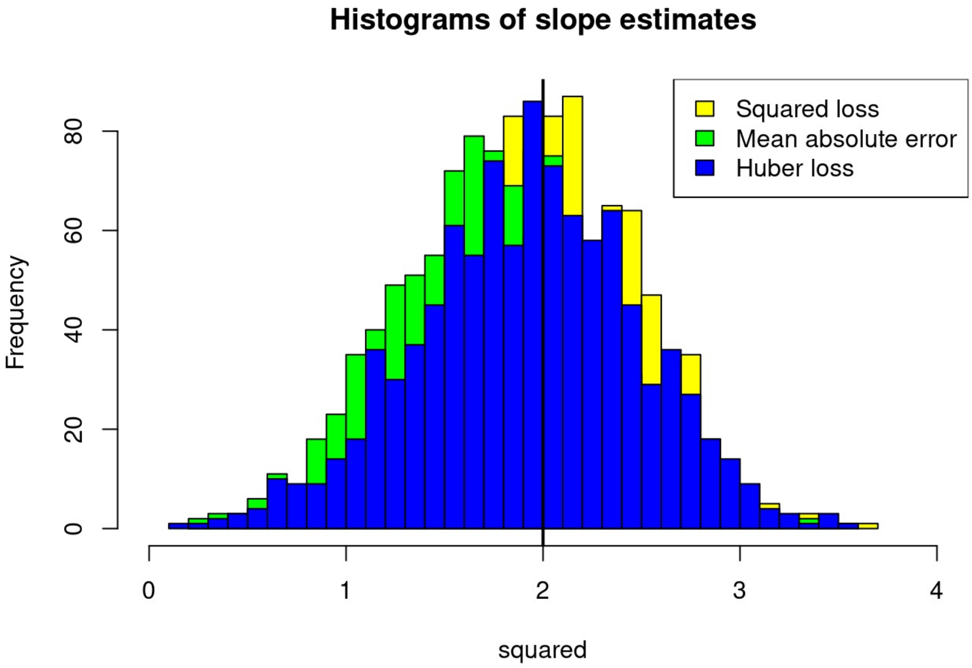
```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```
## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length

## Warning in beta_Analytical_solution[i] <- solve(t(Xmat) %*% Xmat) %*% t(Xmat)
## %*% : number of items to replace is not a multiple of replacement length
```

```r
# Overlay histograms

par(mfrow = c(1, 3))
hist(beta_Analytical_solution, breaks = 20, col = "lightgreen", main = "Analytical_solution Solution")
abline(v = true_beta, col = "yellow")
hist(beta_batch, breaks = 20, col = "lightgreen", main = "Batch Gradient Descent")
abline(v = true_beta, col = "yellow")
hist(beta_stochastic, breaks = 20, col = "lightgreen", main = "Stochastic Gradient Descent")
abline(v = true_beta, col = "yellow")
```



```r
# 4 c, 4D

# Simulation parameters

set.seed(240)
N <- 50
alpha <- 0.01
delta <- 1.345
n_iter <- 1000
true_beta <- 2

# Function to fit linear regression with given loss function

fit_regression <- function(loss) {
  # Simulate the data
  X <- runif(N, -2, 2)
  e <- rnorm(N, mean = 0, sd = 4)
  Y <- 3 + true_beta*X + e

  # Fit linear regression

  X_mat <- cbind(1, X)
  w <- c(0, 0)
  for (i in 1:n_iter) {
    z <- X_mat %*% w - Y
    if (loss == "squared") {
      gradient <- t(X_mat) %*% z / N
    } else if (loss == "absolute") {
      gradient <- t(X_mat) %*% sign(z) / N
    } else if (loss == "huber") {
      gradient <- t(X_mat) %*% (ifelse(abs(z) <= delta, z, delta * sign(z))) / N
    }
    w <- w - alpha * gradient
  }
  return(w[2]) # return the estimate of the slope
}

# Perform 1,000 simulations with different loss functions

n_simulations <- 1000
squared <- replicate(n_simulations, fit_regression("squared"))
absolute <- replicate(n_simulations, fit_regression("absolute"))
huber <- replicate(n_simulations, fit_regression("huber"))

# Plot histograms of the estimates of the slopes

hist(squared, col = "yellow", breaks = 30, xlim = c(0, 4), main = "Histograms of slope estimates")
hist(absolute, col = "green", breaks = 30, add = TRUE)
hist(huber, col = "blue", breaks = 30, add = TRUE)
abline(v = true_beta, col = "black", lwd = 2) # overlay true value
legend("topright", c("Squared loss", "Mean absolute error", "Huber loss"), fill = c("yellow", "green", "blue"))
```


Histograms of slope estimates
Squared loss
Mean absolute error
Huber loss
Frequency
0
20
40
60
80
0
1
2
3
4
squared

```r
# 4 e

# Set seed for reproducibility

set.seed(240)

# Simulate X and e

N <- 50
X <- runif(N, -2, 2)
e <- rnorm(N, 0, 4)

# Simulate Y

Y <- 3 + 2*X + e

# Introduce outliers

for (i in 1:N) {
  if (runif(1) < 0.1) {
    if (runif(1) < 0.5) {
      Y[i] <- Y[i] * 3
    } else {
      Y[i] <- Y[i] / 3
    }
  }
}

# implement squared loss function

loss <- function(beta, X, Y) {
  Y_hat <- beta[1] + beta[2]*X
  sum((Y - Y_hat)^2)
}

# computing Analytical_solution solution

beta_Analytical_solution <- lm(Y ~ X)$coefficients

# Initialize beta with small random values

beta <- runif(2, -1, 1)

# Set learning rate

alpha <- 0.01

# Run stochastic gradient descent

for (i in 1:1000) {

  # Sample a random observation

  j <- sample(1:N, 1)

  # computing gradient of loss function at beta using the selected observation

  grad <- c(2*(beta[1] + beta[2]*X[j] - Y[j]), 2*X[j]*(beta[1] + beta[2]*X[j] - Y[j]))

  # Updating beta

  beta <- beta - alpha*grad
}

# Print results

cat("Analytical_solution solution:\n")
```

```
## Analytical_solution solution:
```

```r
cat(paste0("beta0 = ", round(beta_Analytical_solution[1], 2), ", beta1 = ", round(beta_Analytical_solution[2], 2)
, "\n"))
```

```
## beta0 = 3.15, beta1 = 1.45
```

```
cat("Stochastic gradient descent solution:\n")
```

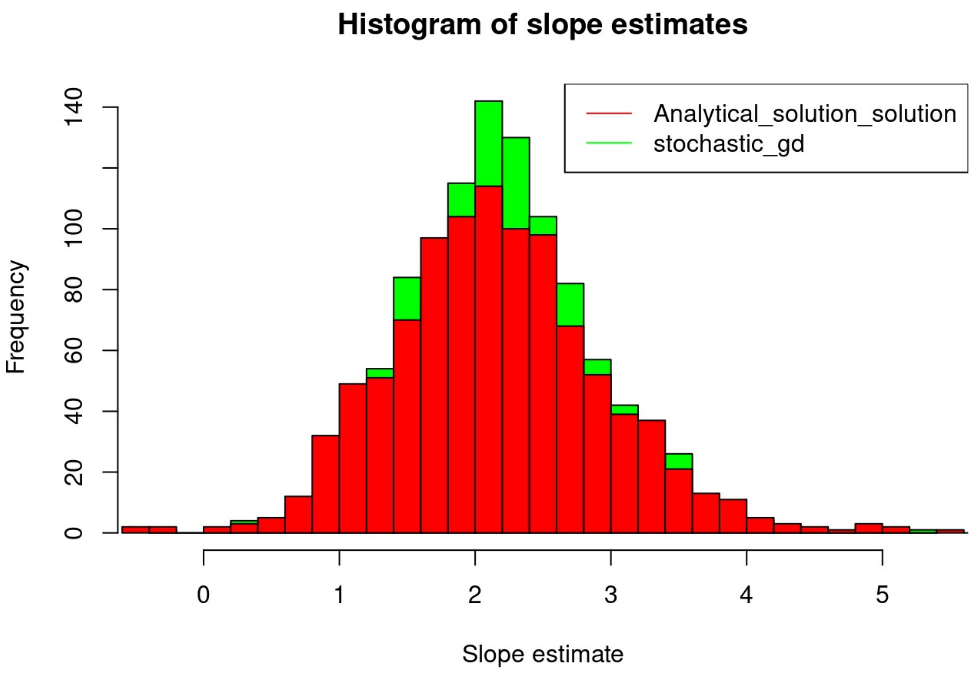
```
## Stochastic gradient descent solution:
```

```
cat(paste0("beta0 = ", round(beta[1], 2), ", beta1 = ", round(beta[2], 2), "\n"))
```

```
## beta0 = 3.24, beta1 = 1.5
```

```
# 4 f

# Set seed for reproducibility

set.seed(240)

# implement function to simulate data and fit model

simulate_and_fit <- function() {
  # Simulate X and e

  N <- 50
  X <- runif(N, -2, 2)
  e <- rnorm(N, 0, 4)

  # Simulate Y

  Y <- 3 + 2*X + e

  # Introduce outliers

  for (i in 1:N) {
    if (runif(1) < 0.1) {
      if (runif(1) < 0.5) {
        Y[i] <- Y[i] * 3
      } else {
        Y[i] <- Y[i] / 3
      }
    }
  }

  # implement squared loss function

  loss <- function(beta, X, Y) {
    Y_hat <- beta[1] + beta[2]*X
    sum((Y - Y_hat)^2)
  }

  # computing Analytical_solution solution

  beta_Analytical_solution <- lm(Y ~ X)$coefficients

  # Initialize beta with small random values

  beta <- runif(2, -1, 1)

  # Set learning rate

  alpha <- 0.01

  # Run stochastic gradient descent

  for (i in 1:1000) {
    # Sample a random observation
    j <- sample(1:N, 1)

    # computing gradient of loss function at beta using the selected observation

    grad <- c(2*(beta[1] + beta[2]*X[j] - Y[j]), 2*X[j]*(beta[1] + beta[2]*X[j] - Y[j]))

    # Updating beta

    beta <- beta - alpha*grad
  }

  # Return estimates
```

```
  c(beta_Analytical_solution[2], beta[2])
}

# Repeat simulation 1000 times and store results

slopes <- replicate(1000, simulate_and_fit())

# Plot histograms of estimates of slopes and true value
hist(slopes[1,], breaks = 30, col = "green", xlab = "Slope estimate", main = "Histogram of slope estimates")
hist(slopes[2,], breaks = 30, col = "red", add = TRUE)
legend("topright", legend = c("Analytical_solution_solution", "stochastic_gd"), col = c("red", "green"), lwd = 1)
```




```
# computing mean squared errors

mse_Analytical_solution <- mean((slopes[1,] - 2)^2)
mse_stochastic_gd <- mean((slopes[2,] - 2)^2)

# Print mean squared errors

cat("Mean squared error of Analytical_solution solution: ", mse_Analytical_solution, "\n")
```

```
## Mean squared error of Analytical_solution solution:  0.4432911
```

```
cat("Mean squared error of stochastic_gd solution: ", mse_stochastic_gd, "\n")
```

```
## Mean squared error of stochastic_gd solution:  0.6792896
```