ADVANCED MATHEMATICAL STATISTICS

MTH – 522

HOMEWORK – 2

Jeevan Kumar Banoth

1. In the toy data example (price Vs age of used car).

a) What is the Hat matrix?

b) Verify that the Hat matrix is symmetric and idempotent.

c) Verify that all diagonal entries are in the range of [0,1].

d) What is the trace of the Hat matrix?

Ans:

a) What is the Hat matrix?

The Hat matrix is a tool in linear regression that: Calculates predictions for the response variable. Determines residuals (errors) in the predictions.

Evaluates the impact of each data point on the estimated relationship between independent and dependent variables (regression coefficients).

$Y\hat{} = X\beta\hat{} = X\ (X\hat{}T\ X)\ \hat{}{-}1\ X\ \hat{}T\ y$

Where,

X is Age of the used car

y is the Price of the used car

Hat Matrix is H = $X\ (X\hat{}T\ X)\ \hat{}{-}1\ X\ \hat{}T$

The below is the Markdown file of my work

---

title: "MTH-522_HW2"

author: "Jeevan"

date: "2024-03-25"

output: word_document

```{r}

Age <- c(6,5,4,3,2,2,1,1)
```

Price <- c(6,9,8,10,11,12,11,13)

df <- data.frame(Age, Price)

str(df)

```

'data.frame':	8 obs. of  2 variables:
 $ Age  : num  6 5 4 3 2 2 1 1
 $ Price: num  6 9 8 10 11 12 11 13
```

```{r}
cat("Is the Hat matrix symmetric?", is_symmetric, "\n")
```

```
 Is the Hat matrix symmetric? TRUE
```

```{r}
cat("Is the Hat matrix idempotent?", is_idempotent, "\n")
```

```
 Is the Hat matrix idempotent? TRUE
```

```{r}
cat("Are all diagonal entries in [0,1] range?", in_range, "\n")
```

```
 Are all diagonal entries in [0,1] range? TRUE
```

```{r}
cat("Trace of the Hat matrix:", trace_Hat_matrix, "\n")
```

```
 Trace of the Hat matrix: 2
```

```{r}
X[,1] <- 1

X[,2] <- c(6,5,4,3,2,2,1,1) # Data from the Toy data example
```

Y[,1] <- matrix(c(6,9,8,10,11,12,11,13),8,1) # Data from the Toy data example

Hat_matrix <- X %*% solve(t(X) %*% X) %*% t(X) # To calculate Hat Matrix

y_Hat <- Hat_matrix %*% Y # To verify the hat matrix

all.equal(Hat_matrix,t(Hat_matrix))

```
[1] TRUE
```

```{r}
Hat_Idp <- Hat_matrix %*% Hat_matrix

all.equal(Hat_matrix, Hat_Idp)
```

```
[1] TRUE
```

```{r}
range(diag(Hat_matrix))  # To check the range
```

```
[1] 0.125 0.500
```

```{r}
sum(diag(Hat_matrix)) # To find the trace
```

```
[1] 2
```

2. Verify the following property of matrix trace

trace(AB)=trace(BA)

by running the following R code three times:

A<-matrix(sample(1:5,16, replace=TRUE),4,4);

B<-matrix(sample(1:5,16, replace=TRUE),4,4);

sum(diag(A %*% B))

sum(diag(B %*% A))

Ans:

```{r}
for (i in 1:3){
    A<-matrix(sample(1:5,16, replace=TRUE),4,4);
    B<-matrix(sample(1:5,16, replace=TRUE),4,4);
    AB_trace <- sum(diag(A %*% B))
    BA_trace <- sum(diag(B %*% A))
    print(paste("Round no = ",i))
    print(paste("Trace of matrix AB = ", AB_trace))
    print(paste("Trace of matrix BA = ", BA_trace))
    print(paste("Trace AB is equal to Trace BA is",all.equal(AB_trace, BA_trace)))
}
```

```
[1] "Round no =  1"
[1] "Trace of matrix AB =  144"
[1] "Trace of matrix BA =  144"
[1] "Trace AB is equal to Trace BA is TRUE"
[1] "Round no =  2"
[1] "Trace of matrix AB =  146"
[1] "Trace of matrix BA =  146"
[1] "Trace AB is equal to Trace BA is TRUE"
[1] "Round no =  3"
[1] "Trace of matrix AB =  141"
[1] "Trace of matrix BA =  141"
[1] "Trace AB is equal to Trace BA is TRUE"
```

```{r}
model <- lm(Price~Age, df)
summary(model)
```

```
Call:
lm(formula = Price ~ Age, data = df)
```

```
Residuals:
   Min     1Q  Median     3Q    Max
-1.2500 -0.6875 -0.0625  0.7812  1.2500


Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  13.3750     0.6847  19.535 1.17e-06 ***
Age          -1.1250     0.1976  -5.692  0.00127 **
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


Residual standard error: 0.9682 on 6 degrees of freedom
Multiple R-squared:  0.8437,                Adjusted R-squared:  0.8177
F-statistic:  32.4 on 1 and 6 DF,  p-value: 0.001269
```

```{r}

H

```

```
     [,1]      [,2]      [,3]    [,4]      [,5]
[1,] 0.3750 0.31250000 0.25000000 0.18750 0.12500000
[2,] 0.3125 0.26041667 0.20833333 0.15625 0.10416667
[3,] 0.2500 0.20833333 0.16666667 0.12500 0.08333333
[4,] 0.1875 0.15625000 0.12500000 0.09375 0.06250000
[5,] 0.1250 0.10416667 0.08333333 0.06250 0.04166667
[6,] 0.1250 0.10416667 0.08333333 0.06250 0.04166667
[7,] 0.0625 0.05208333 0.04166667 0.03125 0.02083333
[8,] 0.0625 0.05208333 0.04166667 0.03125 0.02083333
         [,6]      [,7]      [,8]
[1,] 0.12500000 0.06250000 0.06250000
[2,] 0.10416667 0.05208333 0.05208333
[3,] 0.08333333 0.04166667 0.04166667
[4,] 0.06250000 0.03125000 0.03125000
[5,] 0.04166667 0.02083333 0.02083333
[6,] 0.04166667 0.02083333 0.02083333
[7,] 0.02083333 0.01041667 0.01041667
[8,] 0.02083333 0.01041667 0.01041667
```

```{r}

H %*% H # Idempotent check

```

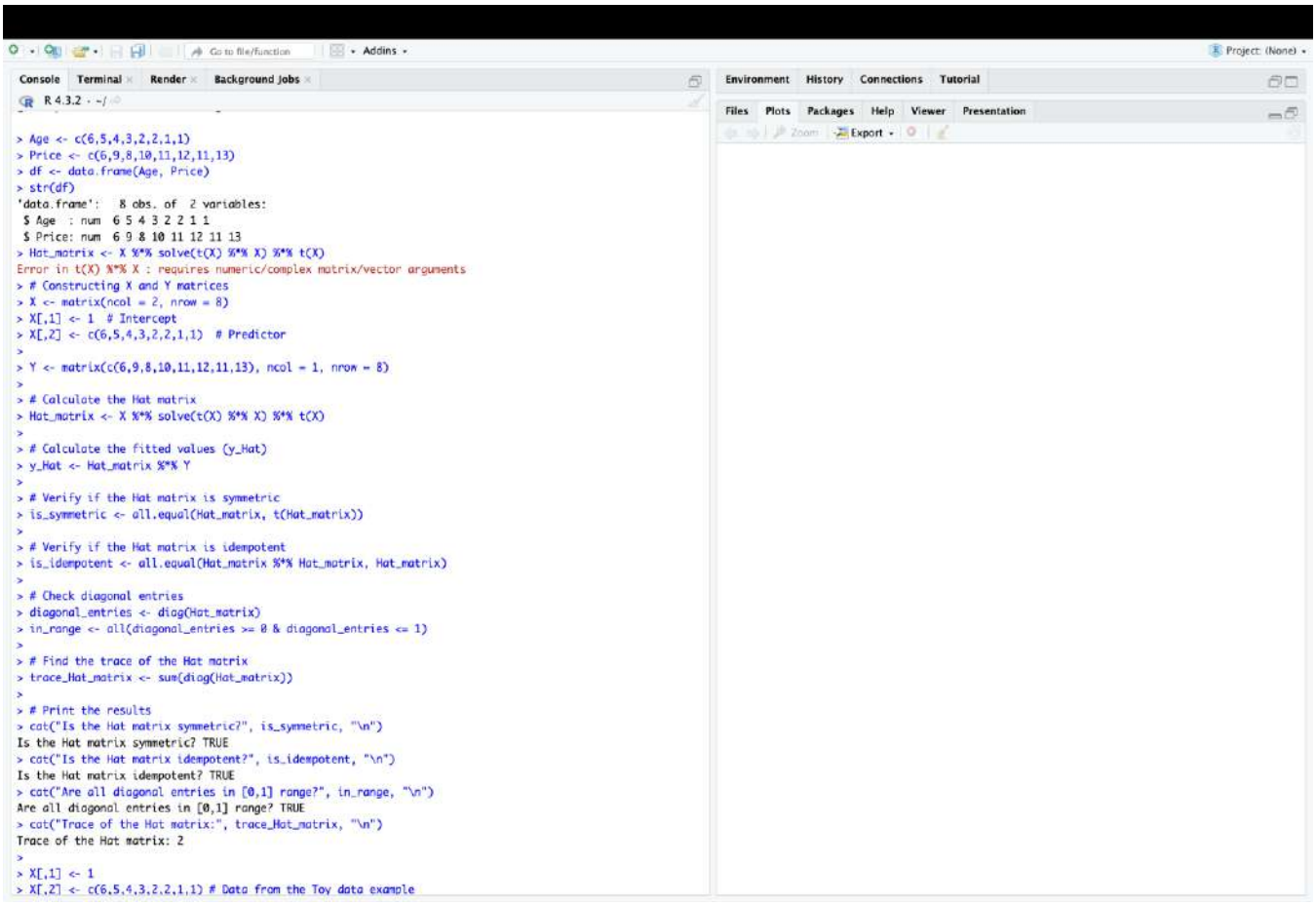
```
     [,1]      [,2]      [,3]    [,4]      [,5]
[1,] 0.3750 0.31250000 0.25000000 0.18750 0.12500000
[2,] 0.3125 0.26041667 0.20833333 0.15625 0.10416667
[3,] 0.2500 0.20833333 0.16666667 0.12500 0.08333333
```

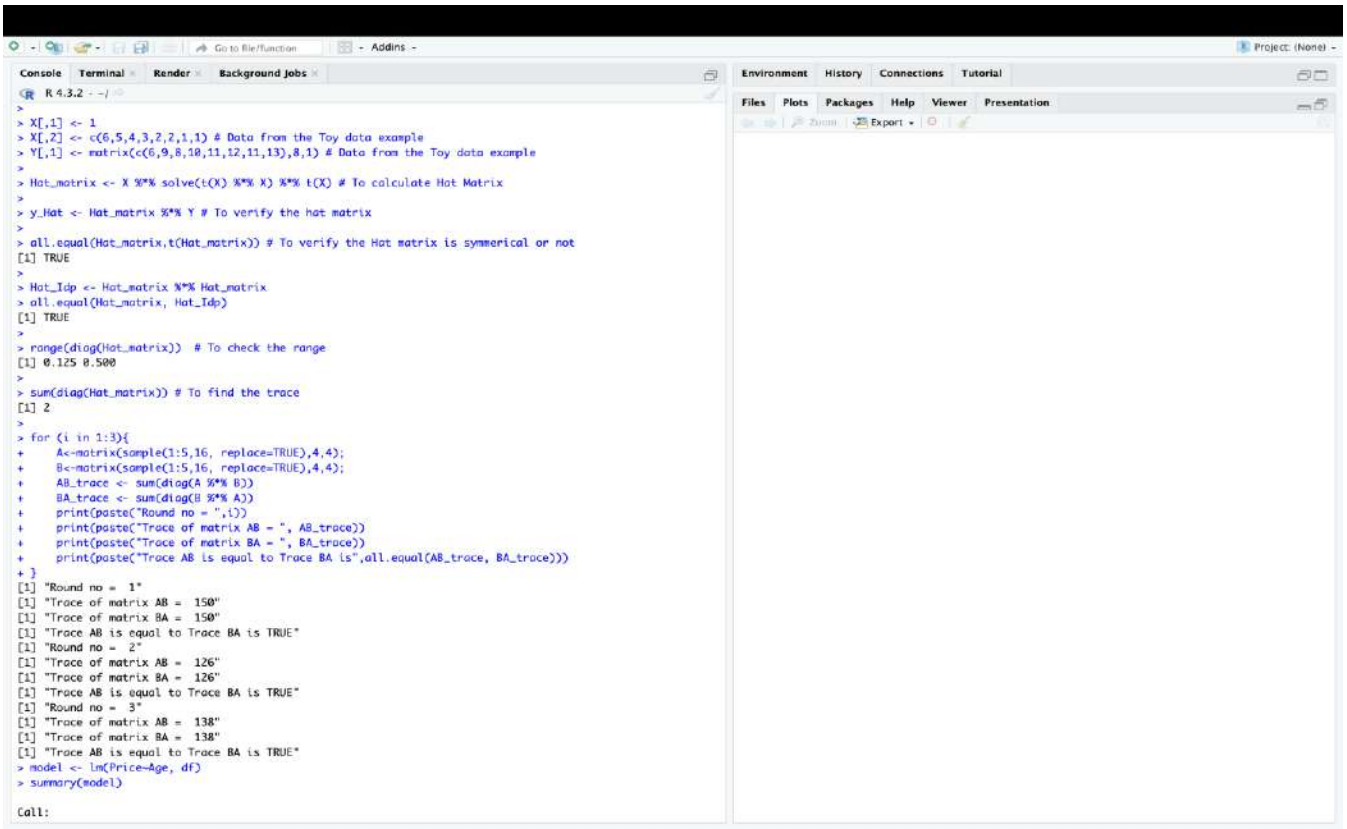
```
[4,] 0.1875 0.15625000 0.12500000 0.09375 0.06250000
[5,] 0.1250 0.10416667 0.08333333 0.06250 0.04166667
[6,] 0.1250 0.10416667 0.08333333 0.06250 0.04166667
[7,] 0.0625 0.05208333 0.04166667 0.03125 0.02083333
[8,] 0.0625 0.05208333 0.04166667 0.03125 0.02083333
        [,6]      [,7]      [,8]
[1,] 0.12500000 0.06250000 0.06250000
[2,] 0.10416667 0.05208333 0.05208333
[3,] 0.08333333 0.04166667 0.04166667
[4,] 0.06250000 0.03125000 0.03125000
[5,] 0.04166667 0.02083333 0.02083333
[6,] 0.04166667 0.02083333 0.02083333
[7,] 0.02083333 0.01041667 0.01041667
[8,] 0.02083333 0.01041667 0.01041667
```

```{r}
t(H)
```

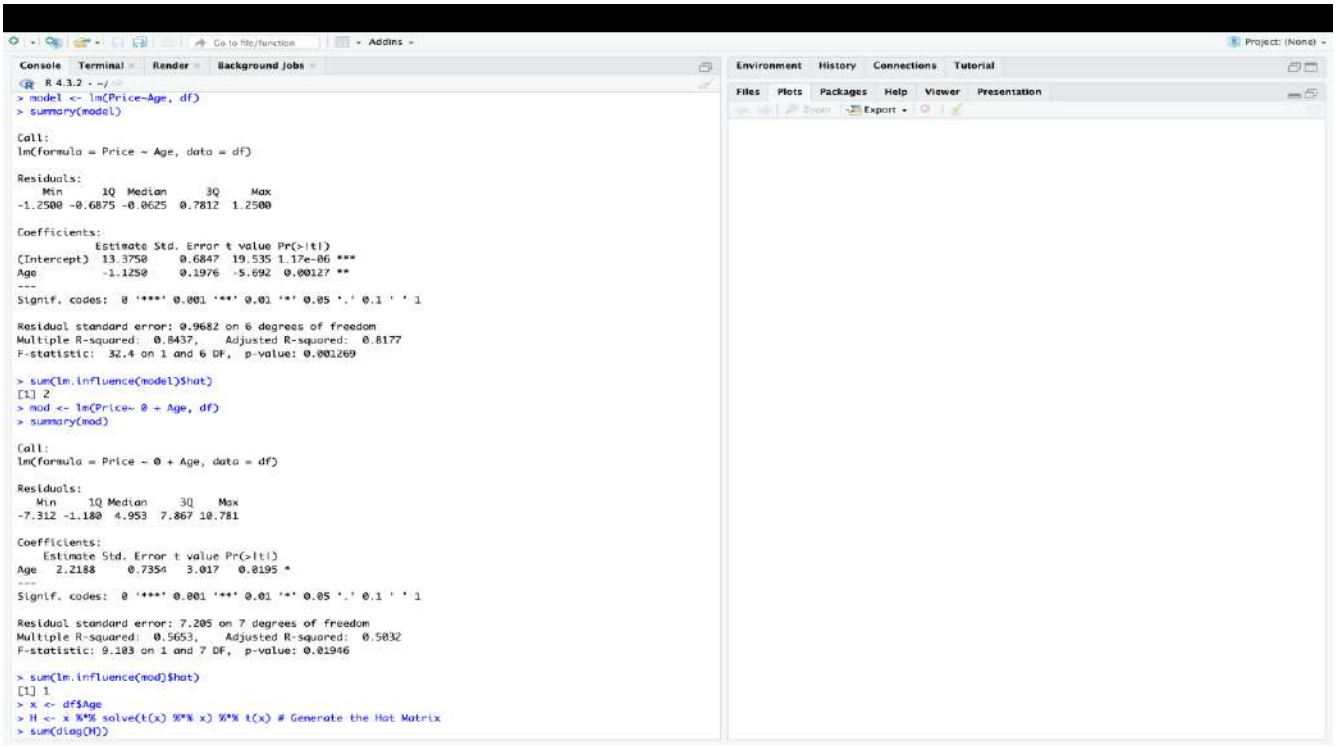
```
     [,1]      [,2]      [,3]    [,4]      [,5]
[1,] 0.3750 0.31250000 0.25000000 0.18750 0.12500000
[2,] 0.3125 0.26041667 0.20833333 0.15625 0.10416667
[3,] 0.2500 0.20833333 0.16666667 0.12500 0.08333333
[4,] 0.1875 0.15625000 0.12500000 0.09375 0.06250000
[5,] 0.1250 0.10416667 0.08333333 0.06250 0.04166667
[6,] 0.1250 0.10416667 0.08333333 0.06250 0.04166667
[7,] 0.0625 0.05208333 0.04166667 0.03125 0.02083333
[8,] 0.0625 0.05208333 0.04166667 0.03125 0.02083333
        [,6]      [,7]      [,8]
[1,] 0.12500000 0.06250000 0.06250000
[2,] 0.10416667 0.05208333 0.05208333
[3,] 0.08333333 0.04166667 0.04166667
[4,] 0.06250000 0.03125000 0.03125000
[5,] 0.04166667 0.02083333 0.02083333
[6,] 0.04166667 0.02083333 0.02083333
[7,] 0.02083333 0.01041667 0.01041667
[8,] 0.02083333 0.01041667 0.01041667
```

Below are the screen shots for the proof that I performed these tasks in my system.

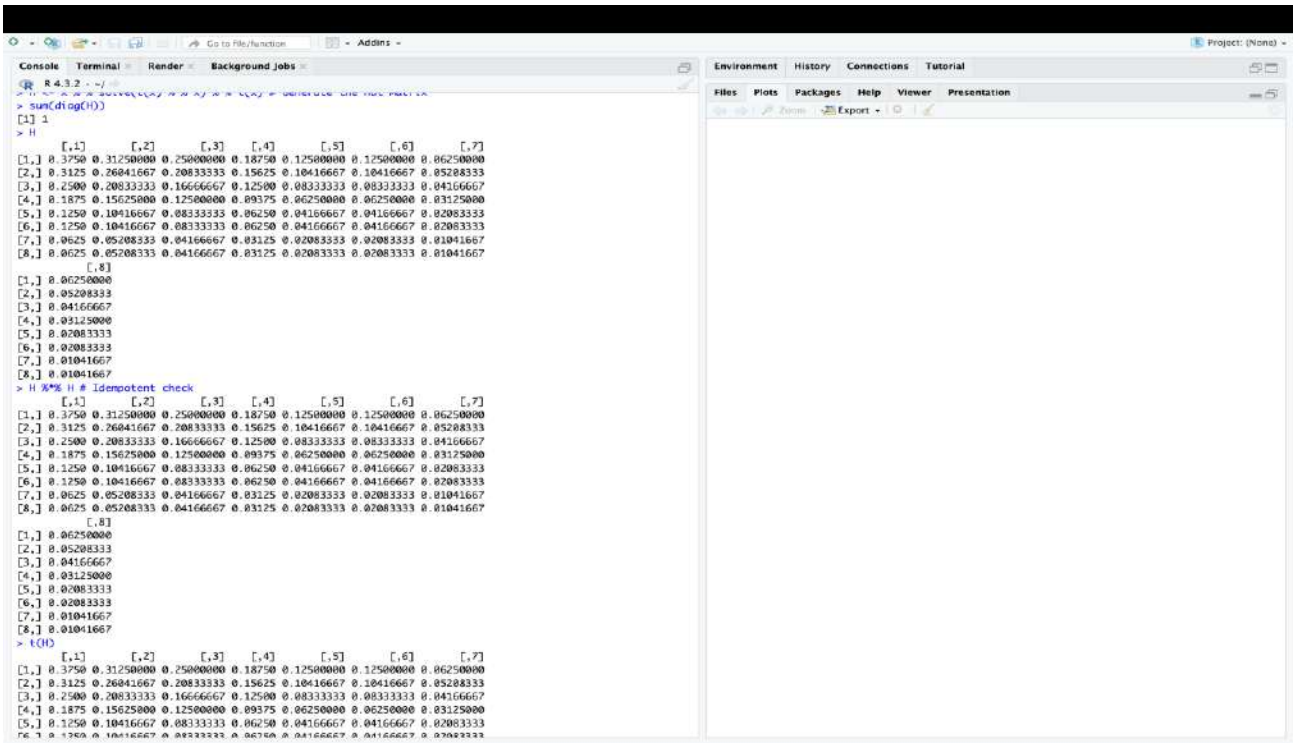

Console
Terminal
Render
Background Jobs
R 4.3.2 · ~/
> Age <- c(6,5,4,3,2,2,1,1)
> Price <- c(6,9,8,10,11,12,11,13)
> df <- data.frame(Age, Price)
> str(df)
'data.frame': 8 obs. of 2 variables:
$ Age : num 6 5 4 3 2 2 1 1
$ Price: num 6 9 8 10 11 12 11 13
> Hat_matrix <- X %*% solve(t(X) %*% X) %*% t(X)
Error in t(X) %*% X : requires numeric/complex matrix/vector arguments
> # Constructing X and Y matrices
> X <- matrix(ncol = 2, nrow = 8)
> X[,1] <- 1 # Intercept
> X[,2] <- c(6,5,4,3,2,2,1,1) # Predictor
>
> Y <- matrix(c(6,9,8,10,11,12,11,13), ncol = 1, nrow = 8)
>
> # Calculate the Hat matrix
> Hat_matrix <- X %*% solve(t(X) %*% X) %*% t(X)
>
> # Calculate the fitted values (y_Hat)
> y_Hat <- Hat_matrix %*% Y
>
> # Verify if the Hat matrix is symmetric
> is_symmetric <- all.equal(Hat_matrix, t(Hat_matrix))
>
> # Verify if the Hat matrix is idempotent
> is_idempotent <- all.equal(Hat_matrix %*% Hat_matrix, Hat_matrix)
>
> # Check diagonal entries
> diagonal_entries <- diag(Hat_matrix)
> in_range <- all(diagonal_entries >= 0 & diagonal_entries <= 1)
>
> # Find the trace of the Hat matrix
> trace_Hat_matrix <- sum(diag(Hat_matrix))
>
> # Print the results
> cat("Is the Hat matrix symmetric?", is_symmetric, "\n")
Is the Hat matrix symmetric? TRUE
> cat("Is the Hat matrix idempotent?", is_idempotent, "\n")
Is the Hat matrix idempotent? TRUE
> cat("Are all diagonal entries in [0,1] range?", in_range, "\n")
Are all diagonal entries in [0,1] range? TRUE
> cat("Trace of the Hat matrix:", trace_Hat_matrix, "\n")
Trace of the Hat matrix: 2
>
> X[,1] <- 1
> X[,2] <- c(6,5,4,3,2,2,1,1) # Data from the Toy data example
Environment
History
Connections
Tutorial
Files
Plots
Packages
Help
Viewer
Presentation
Export



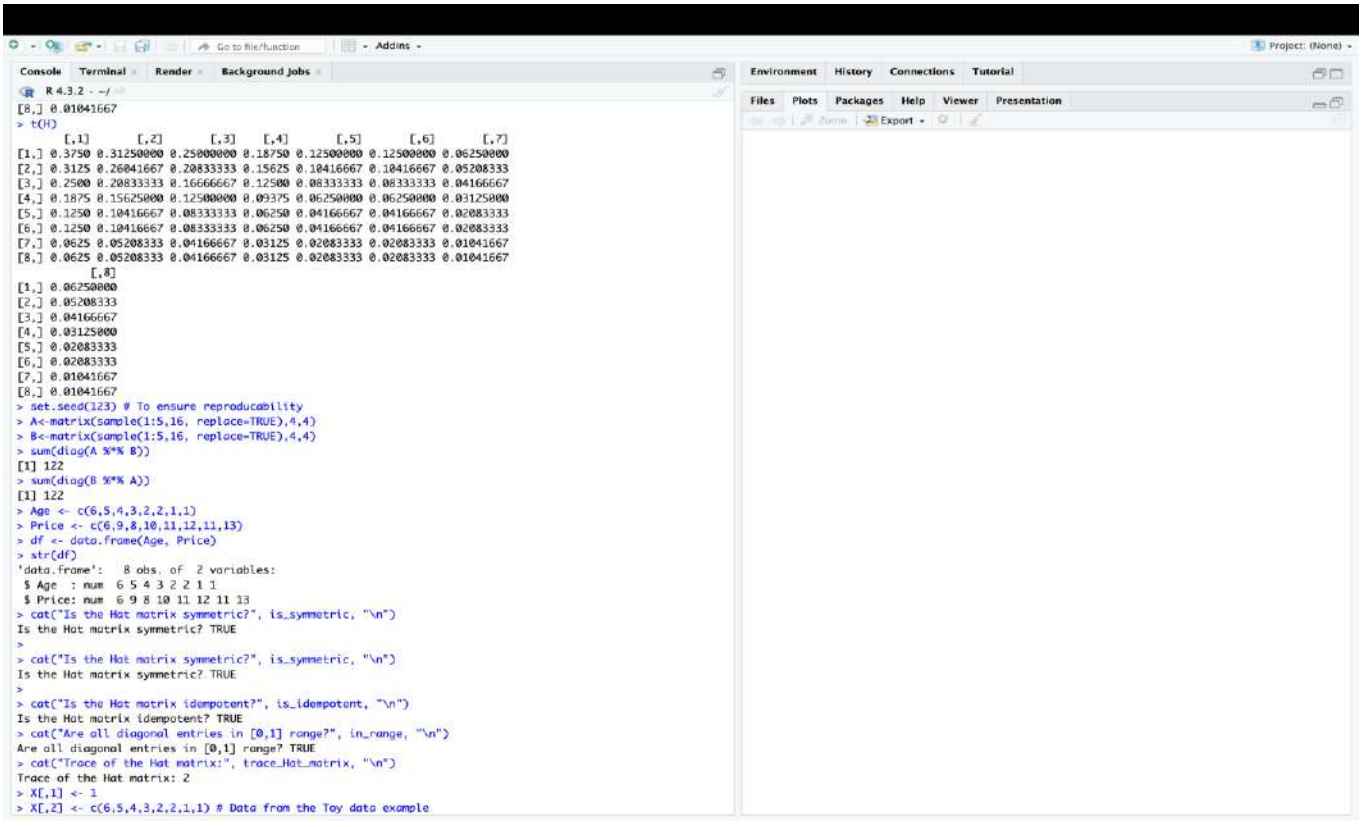
```
> X[,1] <- 1
> X[,2] <- c(6,5,4,3,2,2,1,1) # Data from the Toy data example
> Y[,1] <- matrix(c(6,9,8,10,11,12,11,13),8,1) # Data from the Toy data example
>
> Hat_matrix <- X %*% solve(t(X) %*% X) %*% t(X) # To calculate Hat Matrix
>
> y_Hat <- Hat_matrix %*% Y # To verify the hat matrix
>
> all.equal(Hat_matrix,t(Hat_matrix)) # To verify the Hat matrix is symmerical or not
[1] TRUE
>
> Hat_Idp <- Hat_matrix %*% Hat_matrix
> all.equal(Hat_matrix, Hat_Idp)
[1] TRUE
>
> range(diag(Hat_matrix))  # To check the range
[1] 0.125 0.500
>
> sum(diag(Hat_matrix)) # To find the trace
[1] 2
>
> for (i in 1:3){
+     A<-matrix(sample(1:5,16, replace=TRUE),4,4);
+     B<-matrix(sample(1:5,16, replace=TRUE),4,4);
+     AB_trace <- sum(diag(A %*% B))
+     BA_trace <- sum(diag(B %*% A))
+     print(paste("Round no = ",i))
+     print(paste("Trace of matrix AB = ", AB_trace))
+     print(paste("Trace of matrix BA = ", BA_trace))
+     print(paste("Trace AB is equal to Trace BA is",all.equal(AB_trace, BA_trace)))
+ }
[1] "Round no =  1"
[1] "Trace of matrix AB =  150"
[1] "Trace of matrix BA =  150"
[1] "Trace AB is equal to Trace BA is TRUE"
[1] "Round no =  2"
[1] "Trace of matrix AB =  126"
[1] "Trace of matrix BA =  126"
[1] "Trace AB is equal to Trace BA is TRUE"
[1] "Round no =  3"
[1] "Trace of matrix AB =  138"
[1] "Trace of matrix BA =  138"
[1] "Trace AB is equal to Trace BA is TRUE"
> model <- lm(Price~Age, df)
> summary(model)

Call:
```



```
> model <- lm(Price~Age, df)
> summary(model)

Call:
lm(formula = Price ~ Age, data = df)

Residuals:
    Min      1Q  Median      3Q     Max
-1.2500 -0.6875 -0.0625  0.7812  1.2500

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  13.3750     0.6847  19.535 1.17e-06 ***
Age          -1.1250     0.1976  -5.692  0.00127 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9682 on 6 degrees of freedom
Multiple R-squared:  0.8437,    Adjusted R-squared:  0.8177
F-statistic:  32.4 on 1 and 6 DF,  p-value: 0.001269

> sum(lm.influence(model)$hat)
[1] 2
> mod <- lm(Price~ 0 + Age, df)
> summary(mod)

Call:
lm(formula = Price ~ 0 + Age, data = df)

Residuals:
   Min     1Q Median     3Q    Max
-7.312 -1.180  4.953  7.867 10.781

Coefficients:
    Estimate Std. Error t value Pr(>|t|)
Age   2.2188     0.7354   3.017   0.0195 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.205 on 7 degrees of freedom
Multiple R-squared:  0.5653,    Adjusted R-squared:  0.5032
F-statistic: 9.103 on 1 and 7 DF,  p-value: 0.01946

> sum(lm.influence(mod)$hat)
[1] 1
> x <- df$Age
> H <- x %*% solve(t(x) %*% x) %*% t(x) # Generate the Hat Matrix
> sum(diag(H))
```



```
> sum(diag(H))
[1] 1
> H
      [,1]       [,2]       [,3]    [,4]       [,5]       [,6]       [,7]
[1,] 0.3750 0.31250000 0.25000000 0.18750 0.12500000 0.12500000 0.06250000
[2,] 0.3125 0.26041667 0.20833333 0.15625 0.10416667 0.10416667 0.05208333
[3,] 0.2500 0.20833333 0.16666667 0.12500 0.08333333 0.08333333 0.04166667
[4,] 0.1875 0.15625000 0.12500000 0.09375 0.06250000 0.06250000 0.03125000
[5,] 0.1250 0.10416667 0.08333333 0.06250 0.04166667 0.04166667 0.02083333
[6,] 0.1250 0.10416667 0.08333333 0.06250 0.04166667 0.04166667 0.02083333
[7,] 0.0625 0.05208333 0.04166667 0.03125 0.02083333 0.02083333 0.01041667
[8,] 0.0625 0.05208333 0.04166667 0.03125 0.02083333 0.02083333 0.01041667
           [,8]
[1,] 0.06250000
[2,] 0.05208333
[3,] 0.04166667
[4,] 0.03125000
[5,] 0.02083333
[6,] 0.02083333
[7,] 0.01041667
[8,] 0.01041667
> H %*% H # Idempotent check
      [,1]       [,2]       [,3]    [,4]       [,5]       [,6]       [,7]
[1,] 0.3750 0.31250000 0.25000000 0.18750 0.12500000 0.12500000 0.06250000
[2,] 0.3125 0.26041667 0.20833333 0.15625 0.10416667 0.10416667 0.05208333
[3,] 0.2500 0.20833333 0.16666667 0.12500 0.08333333 0.08333333 0.04166667
[4,] 0.1875 0.15625000 0.12500000 0.09375 0.06250000 0.06250000 0.03125000
[5,] 0.1250 0.10416667 0.08333333 0.06250 0.04166667 0.04166667 0.02083333
[6,] 0.1250 0.10416667 0.08333333 0.06250 0.04166667 0.04166667 0.02083333
[7,] 0.0625 0.05208333 0.04166667 0.03125 0.02083333 0.02083333 0.01041667
[8,] 0.0625 0.05208333 0.04166667 0.03125 0.02083333 0.02083333 0.01041667
           [,8]
[1,] 0.06250000
[2,] 0.05208333
[3,] 0.04166667
[4,] 0.03125000
[5,] 0.02083333
[6,] 0.02083333
[7,] 0.01041667
[8,] 0.01041667
> t(H)
      [,1]       [,2]       [,3]    [,4]       [,5]       [,6]       [,7]
[1,] 0.3750 0.31250000 0.25000000 0.18750 0.12500000 0.12500000 0.06250000
[2,] 0.3125 0.26041667 0.20833333 0.15625 0.10416667 0.10416667 0.05208333
[3,] 0.2500 0.20833333 0.16666667 0.12500 0.08333333 0.08333333 0.04166667
[4,] 0.1875 0.15625000 0.12500000 0.09375 0.06250000 0.06250000 0.03125000
[5,] 0.1250 0.10416667 0.08333333 0.06250 0.04166667 0.04166667 0.02083333
[6,] 0.1250 0.10416667 0.08333333 0.06250 0.04166667 0.04166667 0.02083333
[7,] 0.0625 0.05208333 0.04166667 0.03125 0.02083333 0.02083333 0.01041667
[8,] 0.0625 0.05208333 0.04166667 0.03125 0.02083333 0.02083333 0.01041667
```



```
[8,] 0.01041667
> t(H)
      [,1]       [,2]       [,3]    [,4]       [,5]       [,6]       [,7]
[1,] 0.3750 0.31250000 0.25000000 0.18750 0.12500000 0.12500000 0.06250000
[2,] 0.3125 0.26041667 0.20833333 0.15625 0.10416667 0.10416667 0.05208333
[3,] 0.2500 0.20833333 0.16666667 0.12500 0.08333333 0.08333333 0.04166667
[4,] 0.1875 0.15625000 0.12500000 0.09375 0.06250000 0.06250000 0.03125000
[5,] 0.1250 0.10416667 0.08333333 0.06250 0.04166667 0.04166667 0.02083333
[6,] 0.1250 0.10416667 0.08333333 0.06250 0.04166667 0.04166667 0.02083333
[7,] 0.0625 0.05208333 0.04166667 0.03125 0.02083333 0.02083333 0.01041667
[8,] 0.0625 0.05208333 0.04166667 0.03125 0.02083333 0.02083333 0.01041667
           [,8]
[1,] 0.06250000
[2,] 0.05208333
[3,] 0.04166667
[4,] 0.03125000
[5,] 0.02083333
[6,] 0.02083333
[7,] 0.01041667
[8,] 0.01041667
> set.seed(123) # To ensure reproducability
> A<-matrix(sample(1:5,16, replace=TRUE),4,4)
> B<-matrix(sample(1:5,16, replace=TRUE),4,4)
> sum(diag(A %*% B))
[1] 122
> sum(diag(B %*% A))
[1] 122
> Age <- c(6,5,4,3,2,2,1,1)
> Price <- c(6,9,8,10,11,12,11,13)
> df <- data.frame(Age, Price)
> str(df)
'data.frame':   8 obs. of  2 variables:
 $ Age  : num  6 5 4 3 2 2 1 1
 $ Price: num  6 9 8 10 11 12 11 13
> cat("Is the Hat matrix symmetric?", is_symmetric, "\n")
Is the Hat matrix symmetric? TRUE
>
> cat("Is the Hat matrix symmetric?", is_symmetric, "\n")
Is the Hat matrix symmetric? TRUE
>
> cat("Is the Hat matrix idempotent?", is_idempotent, "\n")
Is the Hat matrix idempotent? TRUE
> cat("Are all diagonal entries in [0,1] range?", in_range, "\n")
Are all diagonal entries in [0,1] range? TRUE
> cat("Trace of the Hat matrix:", trace_Hat_matrix, "\n")
Trace of the Hat matrix: 2
> X[,1] <- 1
> X[,2] <- c(6,5,4,3,2,2,1,1) # Data from the Toy data example
```