

MVLU COLLEGE

7 Performing one-way ANOVA using aov() (R).

The screenshot shows the RStudio interface with the following details:

- File Menu:** File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help.
- Source Editor:** Shows R code for performing one-way ANOVA. The code reads a dataset, calculates group means, and then uses the aov function to perform the analysis.
- Environment Tab:** Displays variables like data3, df_between, df_within, F_value, grand_mean, group_sizes, MSB, MSW, and SSB.
- Files Tab:** Shows the project structure with files like RData, .Rhistory, and various PDFs.

```
R + R 4.5.2 · ~/ ~
> data1 <- read.csv("C:/Users/Admin/Desktop/ANITA TIWARI (data science R)/datasets/titanic.csv")
> data1$Pclass <- as.factor(data1$Pclass)
> data1$Sex <- as.factor(data1$Sex)
> grand_mean <- mean(data1$Fare)
> tm_means <- tapply(data1$Fare, data1$Pclass, mean)
> tm_sizes <- table(data1$Pclass)
> SS_TM <- sum(tm_sizes * (tm_means - grand_mean)^2)
> gender_means <- tapply(data1$Fare, data1$Sex, mean)
> gender_sizes <- table(data1$Sex)
> SS_G <- sum(gender_sizes * (gender_means - grand_mean)^2)
> SS_total <- sum((data1$Fare - grand_mean)^2)
> SS_error <- SS_total - SS_TM - SS_G
> df_TM <- length(tm_means) - 1
> df_G <- length(gender_means) - 1
> df_error <- nrow(data1) - length(tm_means) - length(gender_means) + 1
> MS_TM <- SS_TM / df_TM
> MS_G <- SS_G / df_G
> MS_error <- SS_error / df_error
> F_TM <- MS_TM / MS_error
> F_G <- MS_G / MS_error
>
> F_TM
[1] 255.1855
> F_G
[1] 48.05352
>
```

8 Performing two-way ANOVA using aov() (R).

The screenshot shows the RStudio interface with the following details:

- File Menu:** File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help.
- Source Editor:** Shows R code for performing two-way ANOVA on the titanic dataset. The code reads the CSV file, converts Pclass and Sex to factors, calculates means and sizes for each group, and then performs the two-way ANOVA analysis.
- Environment Tab:** Displays variables like SS_G, SS_TM, SS_tot, SSB, SSW, table, tm_mean, tm_size, and tm_size.
- Files Tab:** Shows the project structure with files like RData, .Rhistory, and various PDFs.

```
R + R 4.5.2 · ~/ ~
> data2 <- read.csv("C:/Users/Admin/Desktop/ANITA TIWARI (data science R)/datasets/titanic.csv")
> data2$Pclass <- as.factor(data2$Pclass)
> data2$Sex <- as.factor(data2$Sex)
> grand_mean <- mean(data2$Fare)
> tm_means <- tapply(data2$Fare, data2$Pclass, mean)
> tm_sizes <- table(data2$Pclass)
> SS_TM <- sum(tm_sizes * (tm_means - grand_mean)^2)
> gender_means <- tapply(data2$Fare, data2$Sex, mean)
> gender_sizes <- table(data2$Sex)
> SS_G <- sum(gender_sizes * (gender_means - grand_mean)^2)
> SS_total <- sum((data2$Fare - grand_mean)^2)
> SS_error <- SS_total - SS_TM - SS_G
> df_TM <- length(tm_means) - 1
> df_G <- length(gender_means) - 1
> df_error <- nrow(data2) - length(tm_means) - length(gender_means) + 1
>
> MS_TM <- SS_TM / df_TM
> MS_G <- SS_G / df_G
> MS_error <- SS_error / df_error
>
> F_TM <- MS_TM / MS_error
> F_G <- MS_G / MS_error
>
> F_TM
[1] 255.1855
> F_G
[1] 48.05352
>
```

MVLU COLLEGE

9 Conducting Chi-square tests using chisq.test() (R)

Chi-Square Test of Independence

The screenshot shows the RStudio interface with the following code in the script editor:

```
1 # Load dataset
2 data <- read.csv("C:/Users/Admin/Desktop/ANKITA TIWARI (data science R)/datasets/Suicides_in_India_random_500 (1).csv")
3
4 # Create contingency table
5 observed <- table(data$Gender, data$Age_group)
6
7 # Row totals, column totals, grand total
8 row_totals <- rowSums(observed)
9 col_totals <- colSums(observed)
10 grand_total <- sum(observed)
11
12 # Expected frequencies
```

Output in the console:

```
> # Chi-square statistic
> chisq.test(observed)
```

Output in the environment pane:

	0-100+	0-14	15-29	30-44	45-59	60+
Female	13	53	48	35	48	55
Male	9	46	63	39	45	45

```
> expected
```

	0-100+	0-14	15-29	30-44	45-59	60+
Female	11.11022	49.99599	56.05611	37.37074	46.96593	50.501
Male	10.88978	49.00401	54.94389	36.62926	46.03407	49.499

```
> chisq.test(observed)
```

```
[1] 4.512593
```

```
> df
```

```
[1] 5
```

```
>
```

Chi-Square Test of Goodness of Fit

The screenshot shows the RStudio interface with the following code in the script editor:

```
1 data <- read.csv("C:/Users/Admin/Desktop/ANKITA TIWARI (data science R)/datasets/Suicides_in_India_random_500 (1).csv")
2
3 observed <- tapply(data$Total, data$Age_group, sum)
4
5 k <- length(observed)
6
7 expected <- rep(sum(observed) / k, k)
8
9 chisq.test(observed - expected)^2 / expected
```

Output in the console:

```
> observed <- tapply(data$Total, data$Age_group, sum)
```

Output in the environment pane:

	0-100+	0-14	15-29	30-44	45-59	60+
observed	2664	368	3268	1864	1544	813

```
> expected
```

	0-100+	0-14	15-29	30-44	45-59	60+
expected	1753.5	1753.5	1753.5	1753.5	1753.5	1753.5

```
> chisq.test(observed - expected)
```

```
[1] 3412.017
```

```
> df
```

```
[1] 5
```

```
>
```