What does the term “Fix in R” mean and why is the type (list) invalid for the variable?

Authored by

stats writer

June 28, 2024

The term "Fix in R" refers to a common programming practice in the R language where a variable or object is modified and its value is permanently changed. This is typically done using the assignment operator, "=" or the "

Fix in R: invalid type (list) for variable

One error you may encounter in R is:

Error in model.frame.default(formula = y ~ x, drop.unused.levels = TRUE) : invalid type (list) for variable 'x'

This error usually occurs when you attempt to fit a regression model or an ANOVA model in R and use a list for one of the variables instead of a vector.

This tutorial shares how to fix this error in practice.

How to Reproduce the Error

Suppose I attempt to fit a in R:

```
#define variables
x <- list(1, 4, 4, 5, 7, 8, 9, 10, 13, 14)
y <- c(10, 13, 13, 14, 18, 20, 22, 24, 29, 31)

#attempt to fit regression model
```

```
model <- lm(y ~ x)
```

```
Error in model.frame.default(formula = y ~ x,
drop.unused.levels = TRUE) :
invalid type (list) for variable 'x'
```

I receive an error because the lm() function can only take vectors as input and the x variable is currently a list.

How to Avoid the Error

The easiest way to avoid this error is to simply use the function to convert the list variable to a vector:

```
#define variables
x <- list(1, 4, 4, 5, 7, 8, 9, 10, 13, 14)
y <- c(10, 13, 13, 14, 18, 20, 22, 24, 29, 31)

#attempt to fit regression model
model <- lm(y ~ unlist(x))

#view the model output
summary(model)
```

Call:

```
lm(formula = y ~ unlist(x))

Residuals:
Min 1Q Median 3Q Max
-1.1282 -0.4194 -0.1087 0.2966 1.7068

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) 6.58447 0.55413 11.88 2.31e-06 ***
unlist(x) 1.70874 0.06544 26.11 4.97e-09 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.8134 on 8 degrees of freedom
Multiple R-squared: 0.9884, Adjusted R-squared: 0.987
F-statistic: 681.8 on 1 and 8 DF, p-value: 4.97e-09
```

Notice that we're able to fit the simple linear regression model without any errors this time because we used unlist() to convert variable x to a vector.

Note that if you're fitting a and you have multiple predictor variables that are currently list objects, you can use unlist() to convert each of them to vectors

before fitting the regression model:

```
#define variables
x1 <- list(1, 4, 4, 5, 7, 8, 9, 10, 13, 14)
x2 <- list(20, 16, 16, 15, 16, 12, 10, 8, 8, 4)
y <- c(10, 13, 13, 14, 18, 20, 22, 24, 29, 31)

#fit multiple linear regression model
model <- lm(y ~ unlist(x1) + unlist(x2))

#view the model output
summary(model)

Call:
lm(formula = y ~ unlist(x1) + unlist(x2))

Residuals:
Min 1Q Median 3Q Max
-1.1579 -0.4211 -0.1386 0.3108 1.7130

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) 8.34282 4.44971 1.875 0.102932
unlist(x1) 1.61339 0.24899 6.480 0.000341 ***
unlist(x2) -0.08346 0.20937 -0.399 0.702044
---
```

Signif. codes: 0 '*' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1**

Residual standard error: 0.8599 on 7 degrees of freedom
Multiple R-squared: 0.9887, Adjusted R-squared: 0.9854
F-statistic: 305.1 on 2 and 7 DF, p-value: 1.553e-07

Once again we don't receive any errors since we converted each of the list objects to vectors.

Additional Resources