

# How can prediction from a rank-deficient fit be misleading and cause problems?

Authored by  
**stats writer**

May 12, 2024

## RECOMMENDED CITATION

stats writer (2024). *How can prediction from a rank-deficient fit be misleading and cause problems?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=143988>

When making predictions using a rank-deficient fit, the model may not accurately represent the true relationship between the variables. This can lead to misleading predictions, as the model is not able to fully capture the complexity of the data. Additionally, rank-deficient fits can cause problems such as overfitting, where the model becomes too specific to the data it was trained on and does not generalize well to new data. This can result in incorrect predictions and a lack of reliability in the model. It is important to carefully consider the limitations of a rank-deficient fit when making predictions and to use caution when interpreting the results.

## **Fix: prediction from a rank-deficient fit may be misleading**

**One common warning you may encounter in R is:**

**Warning message:**

**In predict.lm(model, df) :**

**prediction from a rank-deficient fit may be misleading**

**There are two reasons this warning may occur:**

**Reason 1: Two predictor variables are perfectly correlated.**

**Reason 2: You have more model parameters than observations in the dataset.**

**The following examples show how each problem could occur in practice.**

## Reason #1: Two Predictor Variables Are Perfectly Correlated

Suppose we fit the following multiple linear regression model in R and attempt to use it to make predictions:

```
#create data frame
```

```
df <- data.frame(x1=c(1, 2, 3, 4),
```

```
x2=c(2, 4, 6, 8),
```

```
y=c(6, 10, 19, 26))
```

```
#fit multiple linear regression model
```

```
model <- lm(y~x1+x2, data=df)
```

```
#use model to make predictions
```

```
predict(model, df)
```

```
1 2 3 4
```

```
4.9 11.8 18.7 25.6
```

Warning message:

In predict.lm(model, df) :

prediction from a rank-deficient fit may be misleading

We receive a warning message because the predictor variables  $x_1$  and  $x_2$  are perfectly correlated.

Notice that the values of  $x_2$  are simply equal to the

values of  $x_1$  multiplied by two. This is an example of .

This means that  $x_1$  and  $x_2$  do not provide unique or independent information in the regression model, which cause problems when fitting and interpreting the model.

The easiest way to handle this problem is to simply remove one of the predictor variables from the model since having both predictor variables in the model is redundant.

Reason #2: There Are More Model Parameters Than Observations

Suppose we fit the following multiple linear regression model in R and attempt to use it to make predictions:

```
#create data frame
df <- data.frame(x1=c(1, 2, 3, 4),
x2=c(3, 3, 8, 12),
x3=c(4, 6, 3, 11),
y=c(6, 10, 19, 26))
```

```
#fit multiple linear regression model
model <- lm(y~x1*x2*x3, data=df)
```

```
#use model to make predictions
```

```
predict(model, df)
```

```
1 2 3 4
```

```
6 10 19 26
```

Warning message:

```
In predict.lm(model, df) :
```

```
prediction from a rank-deficient fit may be misleading
```

We receive a warning message because we attempted to fit a regression model with seven total model coefficients:

```
x1x2x3x1*x2x1*3x2*x3x1*x2*x3
```

However, we only have four total observations in the dataset.

Since the number of model parameters is greater than the number of observations in the dataset, we refer to this as .

With high dimensional data, it becomes impossible to find a model that can describe the relationship between the predictor variables and the response variable because we don't have enough observations to train the

**model on.**

**The easiest way to resolve this issue is to collect more observations for our dataset or use a simpler model with less coefficients to estimate.**

**The following tutorials explain how to handle other common errors in R:**

ARABPSYCHOLOGY.COM