How to Troubleshoot “Fix in R” Errors Caused by Singularities

Authored by

stats writer

December 3, 2025

The term "Fix in R" often refers to resolving numerical or statistical issues encountered during modeling or calculation. When dealing with complex statistical models, especially those involving iteration for parameter estimation, researchers sometimes face an ambiguous error: "not defined because of singularities." This error is fundamentally rooted in mathematical instability, where the process of estimating model parameters breaks down.

In a mathematical sense, singularities are points where a function or mapping behaves pathologically--often where a derivative is zero or undefined, making standard analytical methods impossible. In the context of computational statistics, particularly when using iterative algorithms like those within the glm() function for fitting Generalized Linear Models (GLMs), a singularity typically indicates a breakdown in the estimation process because the matrix required for calculation (the information matrix or Hessian matrix) cannot be inverted. This non-invertibility is almost always caused by perfect linear dependence among the predictor variables.

Understanding this mathematical foundation is crucial, as it explains why R cannot proceed with finding unique coefficients. When predictor variables exhibit perfect multicollinearity, the system of equations defining the regression coefficients has infinitely many solutions, preventing the algorithm from converging to a single, stable estimate. This tutorial provides a comprehensive guide on diagnosing and resolving this common, yet often confusing, statistical error in R.

The Specific R Error: "not defined because of singularities"

One of the most concerning error messages a user may encounter in R, particularly after fitting a regression model using the glm() function or similar statistical routines, is the warning regarding singularities. This message does not immediately halt execution but profoundly impacts the validity of the resulting coefficients.

Coefficients: (1 not defined because of singularities)

This specific output signifies a state of complete multicollinearity within the model's design matrix. When two or more predictor variables share an exact linear relationship--meaning one variable can be perfectly predicted from a linear combination of the others--the statistical estimation process fails for at least one of those dependent variables. This condition creates redundancy, preventing the model from isolating the unique contribution of each variable to the outcome.

The presence of a singularity means that the design matrix is not of full rank. Statistically, this rank deficiency implies that the inverse of the matrix, necessary for calculating the least squares or maximum likelihood estimates, does not exist. Consequently, the standard errors and coefficient estimates for the redundant predictor cannot be determined, resulting in the dreaded `NA` values in the model summary output.

Understanding the Role of Multicollinearity

While multicollinearity generally refers to high correlation among predictors, a singularity error in R indicates a case of **perfect multicollinearity**. This is not simply a high correlation (like 0.95), but a correlation of exactly 1 or -1, or a perfect linear combination involving three or more variables. This perfect dependence violates a fundamental assumption of most regression techniques: that predictors are linearly independent.

Perfect linear dependence often arises from data processing errors, such as including a variable and a transformation of that variable (e.g., age and age in months), or by inadvertently including a variable that is the exact sum of other predictors. For example, if a dataset contains columns for "total income," "spouse's income," and "personal income," and the two sub-categories sum perfectly to the total, then perfect linear dependence exists, leading to a singularity when all three are included in the model.

The solution hinges on identifying and removing the redundant information. By simplifying the model structure to include only variables that provide unique predictive power, the design matrix regains full rank, the singularity is resolved, and the iterative estimation process (e.g., Fisher Scoring used by glm() function) can successfully invert the Hessian matrix and converge to a stable set of unique coefficient estimates.

Reproducing the Error in R using glm() function

To properly illustrate the problem, we will attempt to fit a logistic regression model using the glm() function in R to a sample data frame where one predictor variable is perfectly collinear with another. This demonstration highlights how the error manifests in the model summary.

We define a data frame `df` where the predictor `x2` is simply double the value of `x1` across all observations (i.e., `x2 = 2 * x1`). This relationship constitutes perfect linear dependence. When we attempt to fit the model including both `x1` and `x2`, the computational failure becomes evident.

```
#define data
df <- data.frame(y = c(0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1),
x1 = c(3, 3, 4, 4, 3, 2, 5, 8, 9, 9, 9, 8, 9, 9, 9),
x2 = c(6, 6, 8, 8, 6, 4, 10, 16, 18, 18, 18, 16, 18, 18, 18),
x3 = c(4, 7, 7, 3, 8, 9, 9, 8, 7, 8, 9, 4, 9, 10, 13))

#fit logistic regression model
model <- glm(y~x1+x2+x3, data=df, family=binomial)

#view model summary
```

summary(model)

Call:
glm(formula = y ~ x1 + x2 + x3, family = binomial, data = df)

Deviance Residuals:
Min 1Q Median 3Q Max
-1.372e-05 -2.110e-08 2.110e-08 2.110e-08 1.575e-05

Coefficients: (1 not defined because of singularities)
Estimate Std. Error z value Pr(>|z|)
(Intercept) -75.496 176487.031 0.000 1
x1 14.546 24314.459 0.001 1
x2 NA NA NA NA
x3 -2.258 20119.863 0.000 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 2.0728e+01 on 14 degrees of freedom
Residual deviance: 5.1523e-10 on 12 degrees of freedom
AIC: 6

Number of Fisher Scoring iterations: 24

Observing the output immediately reveals the issue. Right before the coefficient table, R provides the crucial warning:

Coefficients: (1 not defined because of singularities)

Furthermore, the coefficient row for predictor `x2` shows `NA` for the Estimate, Standard Error, z value, and p-value. This definitively confirms that the estimation procedure could not calculate a unique coefficient for `x2` because it is perfectly redundant given the presence of `x1`. The singularity has effectively collapsed the dimension of the model space, making parameter estimation impossible for one of the linearly dependent variables.

Diagnosing the Cause: Utilizing the cor() function

To precisely identify which predictor variables are responsible for the singularity and the subsequent rank deficiency, the most straightforward approach is to calculate the correlation structure of the data using the built-in cor() function. This function generates a correlation matrix, which allows for rapid visual inspection of perfect linear relationships (correlations equal to exactly

1.0 or -1.0).

The correlation matrix is symmetric, displaying the pairwise correlation coefficients between all variables in the data frame. While high correlation (e.g., 0.90) indicates severe multicollinearity that might inflate standard errors, only a perfect correlation (1.0) guarantees a singularity error that causes coefficients to be undefined.

Executing the cor() function on our example data frame `df` immediately reveals the culprit variables:

#create correlation matrix
cor(df)

y x1 x2 x3
y 1.0000000 0.9675325 0.9675325 0.3610320
x1 0.9675325 1.0000000 1.0000000 0.3872889
x2 0.9675325 1.0000000 1.0000000 0.3872889
x3 0.3610320 0.3872889 0.3872889 1.0000000

Inspection of the resulting matrix clearly shows that the correlation coefficient between `x1` and `x2` is exactly **1.0000000**. This confirms the perfect linear relationship we suspected. Since these two variables carry identical information regarding the relationship with the response variable `y`, including both in the regression model is unnecessary and mathematically problematic.

The Resolution: Dropping the Redundant Predictor

The standard and most effective method for resolving a singularity caused by perfect multicollinearity is to simply remove one of the perfectly correlated predictor variables from the model specification. Since the variables provide the same information, dropping one does not lead to any loss of predictive power or model fit. The core challenge is computational, not statistical.

In our example, we can choose to drop either `x1` or `x2`. For simplicity, we will remove `x2` from the model formula. By doing so, the model's design matrix becomes full rank again, allowing the estimation procedure to successfully invert the necessary matrices and solve for the unique coefficients.

It is important to note that the decision of which variable to drop--`x1` or `x2`--is arbitrary regarding the model's overall fit (Null and Residual Deviance, AIC). However, the remaining coefficient will capture the combined effect that the redundant pair previously represented. If one variable is inherently more interpretable than the other, keeping the more meaningful variable is generally recommended for ease of communication and reporting.

Interpreting the Fixed Model Output

After refitting the logistic regression model using the reduced formula (`y~x1+x3`), we observe that the singularity error is absent, and all coefficients are properly estimated. The model summary now provides reliable estimates and standard errors for the remaining predictors.

#fit logistic regression model
model <- glm(y~x1+x3, data=df, family=binomial)

#view model summary
summary(model)

Call:
glm(formula = y ~ x1 + x3, family = binomial, data = df)

Deviance Residuals:
Min 1Q Median 3Q Max
-1.372e-05 -2.110e-08 2.110e-08 2.110e-08 1.575e-05

Coefficients:
Estimate Std. Error z value Pr(>|z|)
(Intercept) -75.496 176487.031 0.000 1
x1 14.546 24314.459 0.001 1
x3 -2.258 20119.863 0.000 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 2.0728e+01 on 14 degrees of freedom
Residual deviance: 5.1523e-10 on 12 degrees of freedom
AIC: 6

Number of Fisher Scoring iterations: 24

Crucially, the warning message "not defined because of singularities" is gone. All coefficients now have calculated estimates, standard errors, and corresponding p-values. Furthermore, notice that metrics like Null Deviance, Residual Deviance, and AIC remain unchanged compared to the original, failed model attempt. This confirms that the model's explanatory power was not diminished by removing the redundant variable `x2`.

Important Note: When variables are perfectly correlated, the coefficient estimated for the retained variable (here, `x1`) effectively represents the relationship previously split between the two collinear variables. If the relationship was `x2 = 2 * x1`, and we drop `x2`, the coefficient for `x1` will be

numerically correct for the simplified model. Since the singularity issue is purely due to linear dependence, removing one term ensures that the remaining model structure is mathematically sound for estimation purposes.

Best Practices for Data Preparation

Avoiding the singularity error starts with meticulous data preparation. While the cor() function is excellent for diagnosis, implementing preventative steps can save significant time, especially when working with large datasets or many predictors.

A primary cause of perfect linear dependence involves structural issues in the dataset itself. Analysts should proactively review their variables before fitting models, looking for common pitfalls such as:

Including both a count and its derived rate or percentage (e.g., population and population density).
Including all dummy variables for a categorical factor (which leads to the Dummy Variable Trap, or perfect collinearity with the intercept term).
Including variables that are definitions of one another (like our `x1` and `x2` example, where one is a direct scalar multiple of the other).

For complex cases, especially when dealing with high-dimensional data, calculating the variance inflation factor (VIF) can help diagnose non-perfect but severe multicollinearity. However, for the specific "singularity" error, the exact correlation check remains the definitive diagnostic tool. Resolving this error by dropping the redundant predictor leads to a mathematically well-posed regression model capable of producing stable and interpretable results.