

Learning Guide: Calculating Variance Inflation Factor (VIF) in R for Regression Analysis

Authored by
Mohammed looti

November 9, 2025

RECOMMENDED CITATION

Mohammed looti (2025). *Learning Guide: Calculating Variance Inflation Factor (VIF) in R for Regression Analysis*. PSYCHOLOGICAL STATISTICS. Retrieved from <https://statistics.arabpsychology.com/?p=14186>

In the rigorous field of [regression analysis](#), researchers frequently encounter a significant statistical hurdle known as [multicollinearity](#). This challenge arises when two or more [predictor variables](#) within a statistical model exhibit a high degree of linear correlation with one another. When input variables are tightly inter-correlated, they fundamentally fail to contribute unique or independent information to the regression equation. This redundancy makes it exceedingly difficult for the model to accurately isolate the true, individual effect of each predictor on the response variable. High levels of correlation among these inputs can severely compromise the stability and reliability of the resulting [coefficient estimates](#) and statistical inferences derived from the model output.

To accurately diagnose and quantify the severity of this issue, practitioners rely on the [Variance Inflation Factor \(VIF\)](#). The VIF metric serves as the most widely accepted method for detecting multicollinearity. Specifically, the VIF measures the extent to which the variance of an estimated regression coefficient is inflated due to the presence of linear relationships (collinearity) among the predictors. It assesses the strength and degree of correlation between a specific independent variable and all other predictors included in the model simultaneously. Therefore, grasping the interpretation of VIF values is absolutely crucial for ensuring the structural integrity and trustworthiness of any linear regression model.

Understanding VIF Thresholds and Interpretation

The VIF is a ratio that begins at a minimum value of 1, indicating a complete absence of collinearity, and theoretically has no maximum limit. As the VIF value increases above 1, it signals a greater degree of collinearity, suggesting that the variable in question provides information that is increasingly redundant given the presence of the other predictors. Establishing a reliable standard for interpreting these VIF scores is essential for determining when corrective action is necessary to maintain model validity. The following thresholds provide a standardized guide:

A value of 1 indicates perfect orthogonality. This means there is **no correlation** whatsoever between the predictor variable and any other predictor variables within the model.

A value between 1 and 5 suggests a **moderate correlation**. While some degree of correlation exists, it is generally not deemed severe enough to necessitate immediate structural changes or to significantly bias the coefficient estimates.

A value **greater than 5** signals potentially severe correlation between the predictor variable and the others. If a VIF exceeds this widely used threshold (though some highly rigorous models may use 10), the resulting coefficient estimates and [p-values](#) derived from the regression output are likely unreliable, unstable, and statistically inflated.

It is important to acknowledge that the interpretation of VIF values is often context-dependent. While a high VIF universally points to a potential issue in explanatory modeling, specific domains

or specialized models might tolerate marginally higher values, particularly if the primary objective is accurate prediction rather than detailed causal explanation. Nevertheless, in the vast majority of explanatory modeling contexts, high VIF values require careful consideration and subsequent model refinement, such as variable removal or transformation.

Setting Up the R Environment for VIF Calculation

To provide a practical demonstration of calculating the [Variance Inflation Factor \(VIF\)](#), we will leverage the statistical programming language R. Our example utilizes the well-known, built-in *mtcars* dataset, which offers a standardized and accessible platform for statistical instruction. Before diving into model fitting, it is best practice to inspect the structure and initial observations of the data to become familiar with the variables we intend to incorporate into our regression framework.

#view first six lines of *mtcars*

head(mtcars)

```
# mpg cyl disp hp drat wt  qsec vs am gear carb
#Mazda RX4 21.0 6 160 110 3.90 2.620 16.46 0 1 4 4
#Mazda RX4 Wag 21.0 6 160 110 3.90 2.875 17.02 0 1 4 4
#Datsun 710 22.8 4 108 93 3.85 2.320 18.61 1 1 4 1
#Hornet 4 Drive 21.4 6 258 110 3.08 3.215 19.44 1 0 3 1
#Hornet Sportabout 18.7 8 360 175 3.15 3.440 17.02 0 0 3 2
#Valiant 18.1 6 225 105 2.76 3.460 20.22 1 0 3 1
```

Our goal is to construct a linear model designed to predict the fuel efficiency, measured in miles per gallon (*mpg*). We designate *mpg* as the response variable. The primary [predictor variables](#) chosen for this example include *disp* (engine displacement), *hp* (gross horsepower), *wt* (vehicle weight), and *drat* (rear axle ratio). The critical first step is fitting this linear regression model using R's powerful `lm()` function, which lays the necessary foundation before we can proceed to assess the collinearity among these chosen predictors.

#fit the regression model

model <- lm(mpg ~ disp + hp + wt + drat, data = mtcars)

#view the output of the regression model

summary(model)

#Call:

#lm(formula = mpg ~ disp + hp + wt + drat, data = mtcars)

#

```
#Residuals:
# Min 1Q Median 3Q Max
#-3.5077 -1.9052 -0.5057 0.9821 5.6883
#
#Coefficients:
# Estimate Std. Error t value Pr(>|t|)
#(Intercept) 29.148738 6.293588 4.631 8.2e-05 ***
#disp 0.003815 0.010805 0.353 0.72675
#hp -0.034784 0.011597 -2.999 0.00576 **
#wt -3.479668 1.078371 -3.227 0.00327 **
#drat 1.768049 1.319779 1.340 0.19153
#---
#Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#
#Residual standard error: 2.602 on 27 degrees of freedom
#Multiple R-squared: 0.8376, Adjusted R-squared: 0.8136
#F-statistic: 34.82 on 4 and 27 DF, p-value: 2.704e-10
```

Initial Regression Diagnostics and Identifying Multicollinearity

A careful analysis of the initial regression output provides critical preliminary insights into the model's performance. We observe that the Multiple [R-squared](#) value is **0.8376**, which suggests that approximately 83.76% of the total variance observed in MPG is successfully accounted for by the combination of our selected predictors. Furthermore, the robust F-statistic of **34.82**, accompanied by its highly minute [p-value](#) of **2.704e-10**, strongly confirms that the overall regression model is statistically significant and highly effective at explaining variance. However, we note that the statistical significance varies markedly among the individual predictors: *hp* and *wt* are strongly significant ($p < 0.05$), while *disp* and *drat* appear non-significant, raising immediate suspicions about their unique contributions.

To definitively assess whether [multicollinearity](#) is the underlying cause for the non-significance of some predictors--specifically, whether it is inflating the standard errors of the [coefficient estimates](#)--we must calculate the VIF for each predictor. This crucial calculation requires the specialized `vif()` function, which is conveniently packaged within the highly functional **car** library (Companion to Applied Regression). The procedure involves first loading this external library into the R session and then applying the function directly to our previously fitted model object.

```
#load the car library
library(car)
```

```
#calculate the VIF for each predictor variable in the model
vif(model)

# disp hp wt drat
#8.209402 2.894373 5.096601 2.279547
```

The resulting VIF output immediately illuminates the potential structural weaknesses within the model. We observe that the VIFs associated with *disp* (8.21) and *wt* (5.10) both exceed the generally accepted critical threshold of 5. This finding strongly suggests that these variables share a substantial amount of common, redundant information with the other predictors in the model. This redundancy, in turn, leads to inflated variances in their respective [coefficient estimates](#). The variable *disp*, in particular, exhibits a VIF significantly above the threshold (8.21), which serves as a major red flag demanding immediate investigation and potential remedial action.

Visualizing VIF Values for Rapid Assessment

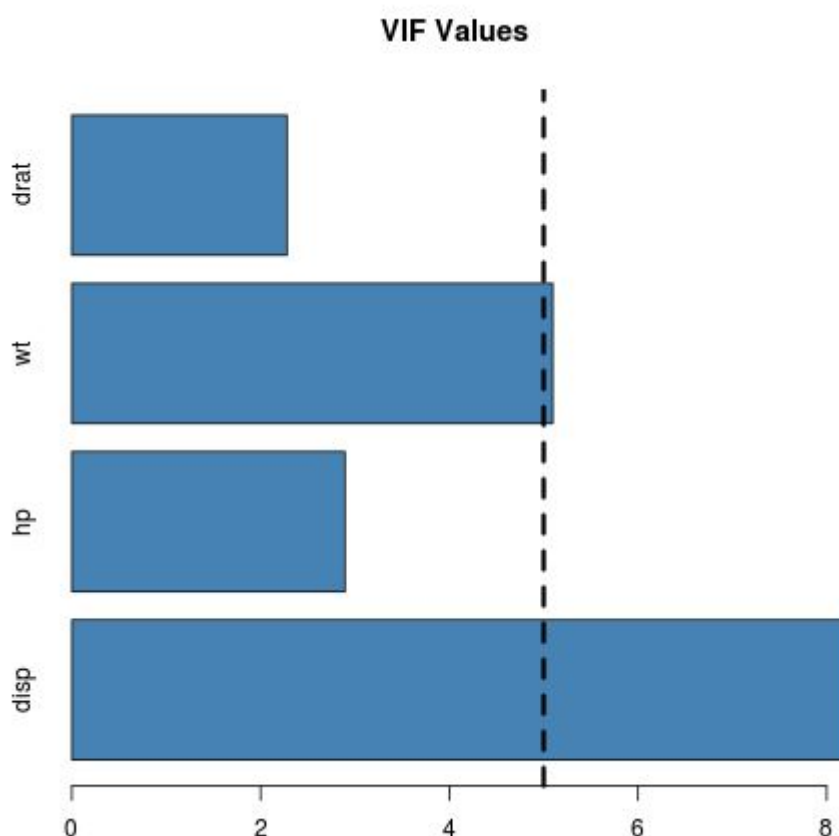
While the numerical output of the VIF calculation is definitive, a visual representation of these values offers an extremely fast and powerful method for assessing which variables pose the greatest threat of collinearity. This technique is particularly invaluable when dealing with models that incorporate a large number of [predictor variables](#), where quickly scanning a list of numbers might be cumbersome. By creating a simple horizontal bar chart, we can effectively display the magnitude of each VIF relative to the critical threshold.

The visualization is enhanced by adding a distinct vertical reference line at the standard VIF = 5 mark. This addition creates an immediate, intuitive visual reference, allowing the analyst to instantly identify which variables breach the acceptable tolerance level. The R code below demonstrates how to generate this diagnostic plot using the calculated VIF vector and base plotting functions.

```
#create vector of VIF values
vif_values <- vif(model)

#create horizontal bar chart to display each VIF value
barplot(vif_values, main = "VIF Values", horiz = TRUE, col = "steelblue")

#add vertical line at 5
abline(v = 5, lwd = 3, lty = 2)
```



Even within this relatively small model example, the resulting bar chart clearly emphasizes that *disp* and *wt* are the most highly correlated variables, as their bars extend beyond the dashed threshold line. Based on the chosen tolerance level for the [multicollinearity](#) issue, the analyst must then decide on the optimal strategy: whether to retain the variable, apply a transformation (like centering or scaling), or completely remove the problematic predictor. Any modification requires a subsequent re-evaluation of the resulting overall model fit, including metrics like the Multiple [R-squared](#) and the statistical significance of the remaining [predictor variables](#).

Diagnosing the Source: Visualizing Predictor Correlations

To gain a deeper, mechanistic understanding of **why** a particular variable, such as *disp*, exhibits a critically high VIF value, it is highly recommended to examine the underlying linear relationships between the predictors directly. The most effective way to achieve this is by constructing a [correlation matrix](#). This matrix systematically displays the linear correlation coefficients (specifically, Pearson's *r*) for every possible pairwise combination of variables, explicitly highlighting which specific relationships are driving the observed collinearity inflation.

#define the variables we want to include in the correlation matrix

data <- mtcars

```
#create correlation matrix
cor(data)

# disp hp wt drat
#disp 1.0000000 0.7909486 0.8879799 -0.7102139
#hp 0.7909486 1.0000000 0.6587479 -0.4487591
#wt 0.8879799 0.6587479 1.0000000 -0.7124406
#drat -0.7102139 -0.4487591 -0.7124406 1.0000000
```

Our focus returns to the variable *disp*, which registered the highest [VIF](#) score (over 8). By analyzing its corresponding row and column within the [correlation matrix](#), the sources of multicollinearity become starkly clear. The matrix confirms that *disp* is extremely strongly correlated with *wt* (0.888) and maintains a strong positive correlation with *hp* (0.791). It also demonstrates a strong negative correlation with *drat* (-0.710). This comprehensive pattern of strong linear relationships with virtually all other predictors definitively explains the reason for its severely inflated VIF score.

Given the converging evidence--that *disp* suffers from a high VIF value, and concurrently, it failed to achieve statistical significance in the original [regression analysis](#)--it stands out as the prime candidate for removal from the model. Eliminating this redundant variable would almost certainly reduce the existing multicollinearity among the remaining variables, thereby stabilizing their [coefficient estimates](#) and significantly improving the overall interpretability of the model. Both the correlation matrix and the VIF calculation serve parallel, essential diagnostic purposes: identifying instances where one variable is highly redundant due to strong linear relationships with others within the regression framework.

Conclusion: Building Robust and Trustworthy Models

Successfully detecting and effectively managing [multicollinearity](#) represents a foundational and critical step toward constructing robust and scientifically sound linear regression models. The [Variance Inflation Factor \(VIF\)](#) provides an indispensable quantitative measure, enabling analysts working in R to quickly flag problematic [predictor variables](#) within complex datasets. By systematically combining numerical VIF calculations with powerful visualizations and detailed correlation matrices, analysts are empowered to make highly informed decisions regarding feature selection and model structure. Ultimately, this diagnostic process leads directly to the generation of more trustworthy, stable, and statistically sound results for any serious analysis.

Further Reading: