Learning Guide: Understanding and Calculating AIC for Regression Models in Python

Authored by

Mohammed looti

November 4, 2025

The Akaike information criterion (AIC) stands as a foundational concept in inferential statistics, serving as a powerful tool to rigorously evaluate and compare the relative quality of multiple candidate statistical models, particularly in the domain of regression analysis. Fundamentally, AIC provides an estimate of the information lost when a specific model is deployed to approximate the true process that generated the observed data. Minimizing this estimated information loss is the statistical objective when selecting the optimal model.

The primary utility of AIC lies in its ability to manage the delicate balance between the goodness of fit--how closely the model adheres to the training data--and the inherent complexity of the model itself. By penalizing models that utilize an excessive number of model parameters, AIC guides analysts toward the most parsimonious model. This approach ensures the selection of a robust model that maximizes predictive power while mitigating the severe risks associated with overfitting, thereby enhancing the model's capacity to generalize accurately to new, unseen datasets.

To properly harness the power of AIC for model selection, it is essential to understand the underlying mathematical framework. The calculation integrates measures of the model's performance (its likelihood) and its structural dimensionality (the count of its defining features or predictors). This integration provides a standardized, quantifiable metric upon which objective comparisons between models can be based, regardless of their varying complexity.

The AIC Formula and Components

The calculation of the Akaike information criterion is defined by a concise and elegant mathematical equation that forms the basis of this evaluation metric. This formula effectively quantifies the trade-off between model fit and model complexity, allowing for direct comparison across different structures:

$$AIC = 2K - 2ln(L)$$

A clear comprehension of the variables embedded within this formula is absolutely critical for accurately interpreting the resulting AIC value and making informed model selection decisions. Each component serves a distinct statistical purpose:

K (The Penalty Term): This represents the number of estimated model parameters within the analysis. When dealing with Ordinary Least Squares (OLS) regression, K typically encapsulates the total count of predictor variables augmented by one, which accounts for the crucial intercept term (the constant). For instance, a model constructed using four distinct predictor variables will have a complexity measure of K = 4 + 1 = 5. This term imposes a penalty for every additional parameter introduced.

***ln*(L) (The Goodness-of-Fit Term)**: This value denotes the natural log-likelihood of the model. The log-likelihood is a measure derived from maximum likelihood estimation and quantifies how

probable the observed data are, given the specific parameters estimated by the model. A substantially higher log-likelihood value signifies a model that provides a superior fit to the underlying data structure.

The inherent structure of the AIC formula is engineered to reward models that successfully maximize the log-likelihood (L), thus demonstrating excellent data fit. Simultaneously, it imposes a calculated penalty (represented by the term 2K) whenever the number of parameters (K) is increased. This deliberate penalty mechanism is vital; it actively discourages the selection of overly complex models which, while potentially achieving near-perfect fits on the specific training data, are likely to perform poorly and lack reliability when applied to new or generalized data sets.

Why AIC is Essential for Model Selection

Traditional metrics, such as the standard R-squared value, provide an indication of how well a model explains the variance in the data. However, they suffer from a critical flaw: they never decrease when additional predictor variables are introduced, even if those variables hold no true predictive power. This tendency can spur analysts toward selecting models that are needlessly complex or deeply overfit. The AIC resolves this systemic issue by incorporating the critical penalty term (2K), thereby offering a far more balanced and reliable foundation for comparing models that possess varying degrees of complexity.

When comparing a designated set of candidate models, the rule for applying the Akaike information criterion is remarkably straightforward and universally applied across statistical disciplines: the specific model that yields the **lowest AIC value** is mathematically considered the superior choice. This model is statistically estimated to be the one that most effectively minimizes the estimated loss of information relative to the true data-generating process. Consequently, it represents the most favorable balance point between achieving high data fit and maintaining statistical parsimony.

Within the modern Python data science ecosystem, the process of calculating AIC for linear models has been significantly streamlined and automated. This efficiency is largely attributed to the powerful open-source statsmodels library, which is expertly designed for statistical modeling and econometric analysis. Specifically, the results object generated after fitting a model using the robust **statsmodels.regression.linear_model.OLS()** function inherently includes an easily accessible property named **.aic**. This attribute automatically computes and returns the crucial Akaike information criterion value, eliminating the need for manual formula calculation.

Practical Example Setup: Using the mtcars Dataset

To concretely illustrate the methodology for calculating and interpreting the AIC, we will proceed by fitting and comparing two distinct linear regression models. Both models are designed to predict

the dependent variable, miles per gallon (MPG), utilizing various independent variables drawn from the classic mtcars dataset. This dataset is an industry standard, frequently employed for demonstrating regression concepts due to its manageable size, well-defined variables, and clear relationships between automotive characteristics.

Our initial step involves preparing the environment by loading the necessary Python libraries. We require pandas for efficient data manipulation and structuring, alongside statsmodels for performing the actual statistical regression analysis. Following the import, we will load the dataset directly into a pandas DataFrame from a reliable remote URL source.

```
from sklearn.linear_model import LinearRegression
import statsmodels.api as sm
import pandas as pd

#define URL where dataset is located
url = "https://raw.githubusercontent.com/Statology/Python-Guides/main/mtcars.csv"

#read in data
data = pd.read_csv(url)

#view head of data
data.head()
```

model mpg cyl disp hp drat wt qsec vs am gear carb
0 Mazda RX4 21.0 6 160.0 110 3.90 2.620 16.46 0 1 4 4
1 Mazda RX4 Wag 21.0 6 160.0 110 3.90 2.875 17.02 0 1 4 4
2 Datsun 710 22.8 4 108.0 93 3.85 2.320 18.61 1 1 4 1
3 Hornet 4 Drive 21.4 6 258.0 110 3.08 3.215 19.44 1 0 3 1
4 Hornet Sportabout 18.7 8 360.0 175 3.15 3.440 17.02 0 0 3 2

With the data successfully loaded, we now proceed to formally define the two separate models that we intend to compare using the AIC metric. It is important to note that the response variable, **mpg** (miles per gallon), will remain constant across both model specifications. The models differ only in their selected set of predictor variables, thereby altering their complexity (K).

Predictor variables in Model 1 (The Complex Model): This model includes four predictors: **disp** (engine displacement), **hp** (gross horsepower), **wt** (vehicle weight), and **qsec** (quarter-mile time). Given the intercept, K = 5.
Predictor variables in Model 2 (The Simpler Model): This model is restricted to only two predictors: **disp** (engine displacement) and **qsec** (quarter-mile time). Given the intercept, K = 3.

Calculating AIC for Model 1 (Complex Model)

Model 1 is designated as our more complex model, leveraging four distinct predictor variables to explain the variability in MPG. When fitting an Ordinary Least Squares (OLS) model using the statsmodels library, a necessary preparatory step is utilizing the **sm.add_constant()** function. This function modifies the predictor matrix to explicitly ensure the inclusion of the intercept term, which is crucial for a properly specified regression equation and essential for accurately calculating the parameter count (K).

The following sequence of Python code defines our response variable and the four complex predictor variables, formally fits the OLS regression model to the data, and subsequently accesses the computed **aic** attribute directly from the resulting model summary object, providing the first comparative value.

```
#define response variable
y = data

#define predictor variables
x = data]

#add constant to predictor variables
x = sm.add_constant(x)

#fit regression model
model = sm.OLS(y, x).fit()

#view AIC of model
print(model.aic)

157.06960941462438
```

The calculated AIC value for Model 1, our complex specification, is approximately **157.07**. This numeric baseline will now be held in reserve as we proceed to calculate the corresponding AIC for the structurally simpler Model 2, enabling a direct and quantitative comparison of model quality.

Calculating AIC for Model 2 (Simpler Model)

Model 2 represents the parsimonious alternative, utilizing only two predictor variables: engine displacement (disp) and quarter-mile time (qsec). While this approach reduces the model's complexity (K=3), leading to a greater penalty avoidance, there is an inherent statistical risk that the reduction in predictive variables might compromise the model's overall fit to the data. The

subsequent AIC calculation will definitively reveal whether the benefits of increased simplicity outweigh any potential sacrifice in explanatory accuracy.

We meticulously repeat the model fitting procedure, ensuring that we accurately redefine the predictor matrix **x** to exclusively include the two chosen variables relevant for this simpler model. As before, the **sm.add_constant()** function is used to correctly incorporate the intercept term necessary for the OLS fitting process.

#define response variable
y = data

#define predictor variables
x = data]

#add constant to predictor variables
x = sm.add_constant(x)

#fit regression model
model = sm.OLS(y, x).fit()

#view AIC of model
print(model.aic)

169.84184864154588

The AIC for Model 2 is computed as approximately **169.84**. With both key metrics now calculated, we are equipped to conduct the final interpretive comparison essential for model selection.

Interpreting the Results and Next Steps

The conclusive step in utilizing the AIC involves a direct comparison of the values derived from the two competing model specifications:

AIC of Model 1 (Four Predictors, K=5): 157.07
AIC of Model 2 (Two Predictors, K=3): 169.84

Based on the established selection criterion--that the lowest AIC value indicates the superior model--we can confidently conclude that **Model 1 is the better-fitting and more statistically robust model**. The difference in AIC values (169.84 - 157.07 = 12.77) is substantial, suggesting a significant improvement in explanatory power. This result clearly demonstrates that the enhanced predictive accuracy gained by incorporating the additional variables (horsepower and weight) was sufficient to dramatically overcome the statistical penalty imposed by the increased complexity (K),

making Model 1 the statistically preferred choice for generalized application.

Once the most appropriate model has been successfully identified using the AIC, the data scientist proceeds with a detailed, in-depth analysis of its statistical results. This subsequent phase includes a thorough examination of the R-squared value to quantify the proportion of variance explained, followed by a meticulous scrutiny of the beta coefficients. Analyzing these coefficients reveals the magnitude, direction, and statistical significance of the exact relationship between the chosen set of predictor variables and the response variable (MPG). Further, crucial steps in validating the selected model typically involve detailed residual analysis, testing for multicollinearity, and external validation to guarantee that the model reliably satisfies all underlying assumptions inherent to linear regression methodologies.

Additional Resources for Statistical Modeling

For those aspiring to deepen their understanding of sophisticated model selection criteria and techniques, it is highly recommended to explore related statistical metrics and methodologies. Specifically, investigating the Bayesian Information Criterion (BIC), which applies a stronger penalty for model complexity than AIC, or utilizing robust cross-validation techniques, provides valuable alternative perspectives. These advanced tools offer different, yet equally powerful, approaches to objectively balancing model complexity against ultimate predictive performance, contributing significantly to a comprehensive statistical toolkit.