

# Package ‘autogam’

February 24, 2025

**Title** Automate the Creation of Generalized Additive Models (GAMs)

**Version** 0.1.0

**Language** en-US

**Description** This wrapper package for 'mgcv' makes it easier to create high-performing Generalized Additive Models (GAMs). With its central function `autogam()`, by entering just a dataset and the name of the outcome column as inputs, 'AutoGAM' tries to automate the procedure of configuring a highly accurate GAM which performs at reasonably high speed, even for large datasets.

**License** MIT + file LICENSE

**Config/testthat/edition** 3

**Encoding** UTF-8

**RoxygenNote** 7.3.2

**Depends** R (>= 4.2.0)

**Imports** cli, dplyr, methods, mgcv, purrr, rlang, staccuracy, stats, stringr, univariateML

**Suggests** testthat (>= 3.0.0)

**URL** <https://github.com/tripartio/autogam>,  
<https://tripartio.github.io/autogam/>

**BugReports** <https://github.com/tripartio/autogam/issues>

**NeedsCompilation** no

**Author** Chitu Okoli [aut, cre] (<<https://orcid.org/0000-0001-5574-7572>>)

**Maintainer** Chitu Okoli <Chitu.Okoli@skema.edu>

**Repository** CRAN

**Date/Publication** 2025-02-24 19:20:06 UTC

## Contents

autogam . . . . .	2
autogam generic methods . . . . .	3

plot.autogam . . . . .	4
print.autogam . . . . .	4
smooth_formula_string . . . . .	5
summary.autogam . . . . .	6

<b>Index</b>	<b>7</b>
--------------	----------

---

autogam	<i>Automate the creation of a Generalized Additive Model (GAM)</i>
---------	--

---

## Description

autogam() is a wrapper for 'mgcv::gam()' that makes it easier to create high-performing Generalized Additive Models (GAMs). By entering just a dataset and the name of the outcome column as inputs, autogam() tries to automate the procedure of configuring a highly accurate GAM which performs at reasonably high speed, even for large datasets.

## Usage

```
autogam(data, y_col, ..., bs = "cr")
```

## Arguments

data	dataframe. All the variables in data will be used to predict y_col. To exclude any variables, assign as data only the subset of variables desired.
y_col	character(1). Name of the y outcome variable.
...	Arguments passed on to <code>mgcv::gam()</code> .
bs	character(1). The default basis function for GAM smooths. See <code>?mgcv::smooth.terms</code> for details. Whereas the default bs in mgcv is 'tp', autogam's default is 'cr', which is much faster and comparably accurate.

## Value

Returns an `mgcv::gam` object, the result of predicting y\_col from all other variables in data.

## Examples

```
autogam(mtcars, 'mpg')
```

---

autogam generic methods

*Generic autogam methods passed on to mgcv::gam methods*

---

## Description

An autogam object contains a gam element that is simply an `mgcv::gam` object. So, it supports all `mgcv::gam` methods by, in most cases, simply passing the gam element on to their corresponding `mgcv::gam` methods. Only the following methods have special specifications for autogam (see their dedicated documentation files for details): `print.autogam()`.

## Usage

```
## S3 method for class 'autogam'  
anova(object, ...)
```

```
## S3 method for class 'autogam'  
coef(object, ...)
```

```
## S3 method for class 'autogam'  
cooks.distance(model, ...)
```

```
## S3 method for class 'autogam'  
formula(x, ...)
```

```
## S3 method for class 'autogam'  
influence(model, ...)
```

```
## S3 method for class 'autogam'  
logLik(object, ...)
```

```
## S3 method for class 'autogam'  
model.matrix(object, ...)
```

```
## S3 method for class 'autogam'  
predict(object, ...)
```

```
## S3 method for class 'autogam'  
residuals(object, ...)
```

```
## S3 method for class 'autogam'  
vcov(object, ...)
```

## Arguments

<code>object, model</code>	An object of class autogam.
<code>...</code>	other arguments

x                    formula

### Value

Returns the return object of the corresponding `mgcv::gam` method.

---

plot.autogam                    *Plot Method for autogam Objects*

---

### Description

This function plots an autogam object. It calls the `mgcv::gam` object `mgcv::plot.gam()` method.

### Usage

```
## S3 method for class 'autogam'
plot(x, ...)
```

### Arguments

x                    An object of class autogam.  
 ...                  Additional arguments passed to other methods.

### Value

Same return object as `mgcv::print.gam()`.

---

print.autogam                    *Print Method for autogam Objects*

---

### Description

This function prints an autogam object. It calls the `mgcv::gam` object `print()` method and then adds basic performance metrics from the autogam object:

- For models that predict numeric outcomes, it prints "MAE", the mean absolute error, and "Std. accuracy", the standardized accuracy (staccuracy) of the winsorized MAE relative to the mean absolute deviation.
- For models that predict binary outcomes, it prints "AUC", the area under the ROC curve.

### Usage

```
## S3 method for class 'autogam'
print(x, ...)
```

**Arguments**

x	An object of class <code>autogam</code> .
...	Additional arguments passed to other methods.

**Value**

Invisibly returns the input object `x`.

---

`smooth_formula_string` *Create a character string for a `mgcv::gam` formula*

---

**Description**

Create a character string that wraps appropriate variables in a dataframe with `s()` smooth functions. Based on the datatype of each variable, it determines whether it is a numeric variable to be smoothed:

- Non-numeric: no smoothing.
- Numeric: determine knots based on the number of unique values for that variable:
  - $\leq 4$ : no smoothing
  - 5 to 19 (inclusive): smooth function with knots equal to the floored half of the number of unique values. E.g., 6 unique values receive 3 knots, 7 will receive 3 knots, and 8 will receive 4 knots.
  - $\geq 20$ : smooth function with no specified number of knots, allowing the `gam()` function to detect the appropriate number.

**Usage**

```
smooth_formula_string(
  data,
  y_col,
  smooth_fun = "s",
  bs = "cr",
  expand_parametric = TRUE
)
```

**Arguments**

<code>data</code>	dataframe. All the variables in <code>data</code> except <code>y_col</code> will be listed in the resulting formula string. To exclude any variables, assign as <code>data</code> only the subset of variables desired.
<code>y_col</code>	character(1). Name of the y outcome variable.
<code>smooth_fun</code>	character(1). Function to use for smooth wraps; default is 's' for the <code>s()</code> function.
<code>bs</code>	See documentation for <a href="#">autogam()</a>
<code>expand_parametric</code>	logical(1). If TRUE (default), explicitly list each non-smooth (parametric) term. If FALSE, use <code>.</code> to lump together all non-smooth terms.

**Value**

Returns a single character string that represents a formula with `y_col` on the left and all other variables in data on the right, each formatted with an appropriate `s()` function when applicable.

**Examples**

```
smooth_formula_string(mtcars, 'mpg')
```

---

`summary.autogam`*Summary Method for autogam Objects*

---

**Description**

This function returns a summary of an autogam object. It calls the `mgcv::gam` object `mgcv::summary.gam()` method.

**Usage**

```
## S3 method for class 'autogam'  
summary(object, ...)
```

**Arguments**

<code>object</code>	An object of class autogam.
<code>...</code>	Additional arguments passed to other methods.

**Value**

Same return object as `mgcv::summary.gam()`.

# Index

`anova.autogam`(`autogam` generic methods), 3  
`autogam`, 2  
`autogam` generic methods, 3  
`autogam()`, 5  
  
`coef.autogam`(`autogam` generic methods), 3  
`cooks.distance.autogam`(`autogam` generic methods), 3  
  
`formula.autogam`(`autogam` generic methods), 3  
  
`influence.autogam`(`autogam` generic methods), 3  
  
`logLik.autogam`(`autogam` generic methods), 3  
  
`mgcv::gam()`, 2  
`mgcv::plot.gam()`, 4  
`mgcv::print.gam()`, 4  
`mgcv::summary.gam()`, 6  
`model.matrix.autogam`(`autogam` generic methods), 3  
  
`plot.autogam`, 4  
`predict.autogam`(`autogam` generic methods), 3  
`print.autogam`, 4  
`print.autogam()`, 3  
  
`residuals.autogam`(`autogam` generic methods), 3  
  
`smooth_formula_string`, 5  
`summary.autogam`, 6  
  
`vcov.autogam`(`autogam` generic methods), 3