

# Package: sAIC (via r-universe)

September 12, 2024

**Type** Package

**Title** Akaike Information Criterion for Sparse Estimation

**Version** 1.0.1

**Date** 2022-10-18

**Author** Shuichi Kawano [aut, cre]  
(<https://orcid.org/0000-0002-0804-0141>), Yoshiyuki Ninomiya  
[aut]

**Maintainer** Shuichi Kawano <skawano@math.kyushu-u.ac.jp>

**Suggests** MASS, glmnet, glasso

**Description** Computes the Akaike information criterion for the  
generalized linear models (logistic regression, Poisson  
regression, and Gaussian graphical models) estimated by the  
lasso.

**License** GPL (>= 2)

**URL** <https://doi.org/10.1214/16-EJS1179>,  
<https://sites.google.com/site/shuichikawanoen/>

**Repository** CRAN

**NeedsCompilation** no

**Date/Publication** 2022-10-18 23:22:35 UTC

## Contents

sAIC . . . . .	2
<b>Index</b>	<b>4</b>

---

sAIC	<i>Compute the Akaike information criterion for the lasso in generalized linear models</i>
------	--

---

### Description

This function computes the Akaike information criterion for generalized linear models estimated by the lasso.

### Usage

```
sAIC(x, y=NULL, beta, family=c("binomial", "poisson", "ggm"))
```

### Arguments

x	A data matrix.
y	A response vector. If you select family="ggm", you should omit this argument.
beta	An estimated coefficient vector including the intercept. If you select family="ggm", you should use an estimated precision matrix.
family	Response type (binomial, Poisson or Gaussian graphical model).

### Value

AIC	The value of AIC.
-----	-------------------

### Author(s)

Shuichi Kawano  
<skawano@math.kyushu-u.ac.jp>

### References

Ninomiya, Y. and Kawano, S. (2016). *AIC for the Lasso in generalized linear models*. *Electronic Journal of Statistics*, 10, 2537–2560. doi:10.1214/16EJS1179

### Examples

```
library(MASS)
library(glmnet)
library(glasso)

### logistic model
set.seed(3)
n <- 100; np <- 10; beta <- c(rep(0.5,3), rep(0,np-3))
Sigma <- diag( rep(1,np) )
for(i in 1:np) for(j in 1:np) Sigma[i,j] <- 0.5^(abs(i-j))
x <- mvrnorm(n, rep(0, np), Sigma)
y <- rbinom(n,1,1-1/(1+exp(x%*%beta)))
```

```

glmnet.object <- glmnet(x,y,family="binomial",alpha=1)
coef.glmnet <- coef(glmnet.object)
### coefficients
coef.glmnet[ ,10]
### AIC
sAIC(x=x, y=y, beta=coef.glmnet[ ,10], family="binomial")

### Poisson model
set.seed(1)
n <- 100; np <- 10; beta <- c(rep(0.5,3), rep(0,np-3))
Sigma <- diag( rep(1,np) )
for(i in 1:np) for(j in 1:np) Sigma[i,j] <- 0.5^(abs(i-j))
x <- mvrnorm(n, rep(0, np), Sigma)
y <- rpois(n,exp(x%*%beta))
glmnet.object <- glmnet(x,y,family="poisson",alpha=1)
coef.glmnet <- coef(glmnet.object)
### coefficients
coef.glmnet[ ,20]
### AIC
sAIC(x=x, y=y, beta=coef.glmnet[ ,20], family="poisson")

### Gaussian graphical model
set.seed(1)
n <- 100; np <- 10; lambda_list <- 1:100/50
invSigma <- diag( rep(0,np) )
for(i in 1:np)
{
  for(j in 1:np)
  {
    if( i == j ) invSigma[i ,j] <- 1
    if( i == (j-1) || (i-1) == j ) invSigma[i ,j] <- 0.5
  }
}
Sigma <- solve(invSigma)
x <- scale(mvrnorm(n, rep(0, np), Sigma))
glasso.object <- glassopath(var(x), rholist=lambda_list, trace=0)
### AIC
sAIC(x=x, beta=glasso.object$wi[, ,10], family="ggm")

```

# Index

- \* **models**
  - sAIC, 2
- \* **regression**
  - sAIC, 2
- sAIC, 2