ARS Manual

December 11, 2013

ARSpackage *ARSpackage: an Adaptive Rejection Sampler*

Description

Final project for Statistics 243, an R package that performs adaptive rejection sampling, first proposed by Gilks and Wild in 1992.

Details

Package:	ARSpackage
Type:	Package
Version:	1.0
Date:	2013-12-13
Depends:	methods, numDeriv
Collate:	'adapt_reject.r', 'ars_methods.r'

Author(s)

J. Bladen, L. Felberg, H.W. Tsao, S. Tu

References

Gilks, Wild, 1992. http://faculty.chicagobooth.edu/hedibert.lopes/teaching/ccis2010/1992GilksWild.pdf.

See Also

https://bitbucket.org/lfelberg/stat243_final_proj
https://github.com/paciorek/stat243-fall-2013/tree/master/project

Examples

```
# Testing the normal distribution from -Inf to Inf
n_samples <- ars( 10000, fx = function(x){(1/sqrt(2*pi)*exp((-(x-0)^2)/2))}, bounds=c(-Inf, Inf) )
```

```
# Testing Gamma(2,1) on interval[0.01,Inf]
sample<-ars(10000,function(x){1/2*x*exp(-x)},c(0.01, Inf))
```

Cadapt_reject_sample *The adapt_reject class*

Description

This class contains all the methods used to perform an AR sampling.

Slots

n: Variable of class "numeric", n, containing the number of points to sample, taken as user input.

f_x: Function of class "function", containing the f(x) to sample from, taken as user input.

bounds: Variable of class "numeric", n, containing the bounds of the function, taken as user input.

output: Variable of class "vector", containing sampled points to return to user.

h_at_x: Variable of class "vector", containing computed log(f(x)) values at all x values

hprime_at_x: Variable of class "vector", containing computed derivative of log(f(x)) values at all x values

z: Variable of class "vector", containing abscissae of upper bound function.

samples: Variable of class "vector", containing random numbers generated by s(x) and unif.

x: Variable of class "vector", containing x values used in ARS.

weights: Variable of class "vector", containing sampled points to return to user.

output: Variable of class "numeric", containing sampled points to return to user.

mat_sorted: Variable of class "matrix", containing x values, their corresponding h and h prime values, sorted by increasing x.

guess_of_mode: Variable of class "numeric", containing an optional user input guess of the mode of the distribution, should be within 200 of actual mode.

---

ars *ars: The adapt_reject function*

Description

This calls the class Cadapt_reject_sample and its methods. It returns a vector of samples generated via the Adaptive rejective sampling method.

Usage

```
ars(n_samples, fx, bounds = c(-Inf, Inf),
  guess_of_mode = -999, ...)
```

Arguments

n_samples:	Number of samples desired from distribution
fx:	Function to sample from
bounds:	Bounds of function of interest. The default is an unbounded function

Value

a vector containing n points sampled from the f(x) distribution

---

ev_h *ev_h*

Description

Cadapt_reject_sample method that evaluates the log(f(x)) for a given x and the derivative as well.

Arguments

object	Cadapt_reject_sample object

---

gen_x *gen_x*

Description

Cadapt_reject_sample method for generating first two points. If the distribution is unbounded, then find the function's mode and pick points surrounding it. If it's bounded on one side, we use the bound given and search until we find a point that corresponds to the opposite end of the domain with respect to their derivatives. If bounded on both sides, use given bounds.

Arguments

object	Cadapt_reject_sample object

---

initialize *Cadapt_reject_sample initialization*

Description

A method to intialize the ARS class for sampling. Will store values input from user and will also initialize empty arrays for all other slots.

Arguments

object	Cadapt_reject_sample object
n	numeric determining the number of samples to obtain
f_x	function for distribution to sample from
bounds	vector of distribution bounds
guess_of_mode	numeric optional idea of where distribution is located

lower *lower*

Description

Cadapt_reject_sample method to evaluate the lower bound of x_star.

Arguments

object	Cadapt_reject_sample object

s_x *s(x)*

Description

Cadapt_reject_sample method to normalize the upper bounds of log(f(x)). Multiple objective are performed here. The most important being the calculation of the abcissa vector Z. Additionally, the weights and exact values of the piecewise integration of each interval and the normalization factor for the entire upper bound are calculated and the x's, their evaluations and their derivatives are sorted by x.

Arguments

object	Cadapt_reject_sample object

sampling *sampling*

Description

Method to sample from s_x. The basic algorithm is as follows: 1. Determine an interval to sample from using the weights of integration of the function on each interval, computed in the s_x method. 2. Use inverse CDF method to sample from within the selected interval. Return the object with new sample.

Arguments

object	Cadapt_reject_sample object

update *update*

Description

Cadapt_reject_sample method to determine which ACC/REJ criteria a given sampled value fits into and updates the samples and x values accordingly.

Arguments

object	Cadapt_reject_sample object

upper *upper*

Description

Cadapt_reject_sample method to evaluate the upper bound of x_star.

Arguments

object	Cadapt_reject_sample object

validity_ars *Validity checks for S4* adapt_reject_sample *object*

Description

The main objective of this validity check is to ensure at creation that the number of samples desired is a positive integer

Usage

```
validity_ars(object)
```

Arguments

object	An adapt_reject_sample object

Index