

# Package: RcppDist (via r-universe)

September 5, 2024

**Title** 'Rcpp' Integration of Additional Probability Distributions

**Version** 0.1.1.9000

**Description** The 'Rcpp' package provides a C++ library to make it easier to use C++ with R. R and 'Rcpp' provide functions for a variety of statistical distributions. Several R packages make functions available to R for additional statistical distributions. However, to access these functions from C++ code, a costly call to the R functions must be made. 'RcppDist' provides a header-only C++ library with functions for additional statistical distributions that can be called from C++ when writing code using 'Rcpp' or 'RcppArmadillo'. Functions are available that return a 'NumericVector' as well as doubles, and for multivariate or matrix distributions, 'Armadillo' vectors and matrices. 'RcppDist' provides functions for the following distributions: the four parameter beta distribution; the Laplace distribution; the location-scale t distribution; the truncated normal distribution; the truncated t distribution; a truncated location-scale t distribution; the triangle distribution; the multivariate normal distribution\*; the multivariate t distribution\*; the Wishart distribution\*; and the inverse Wishart distribution\*. Distributions marked with an asterisk rely on 'RcppArmadillo'.

**License** GPL (>= 2.0)

**URL** <https://github.com/duckmayr/RcppDist>

**BugReports** <https://github.com/duckmayr/RcppDist/issues>

**Depends** R (>= 3.0.0)

**Encoding** UTF-8

**LazyData** true

**LinkingTo** Rcpp, RcppArmadillo

**Imports** Rcpp

**RoxygenNote** 6.1.0

**Suggests** testthat, knitr, rmarkdown

**VignetteBuilder** knitr

**Repository** <https://duckmayr.r-universe.dev>

**RemoteUrl** <https://github.com/duckmayr/rcppdist>

**RemoteRef** HEAD

**RemoteSha** e2859cde3bfab8fffbba4fee4989bc1467e35c48

## Contents

bayeslm . . . . .	2
RcppDist . . . . .	4

<b>Index</b>	<b>5</b>
--------------	----------

---

bayeslm	<i>bayeslm</i>
---------	----------------

---

## Description

Demonstrates the use of RcppDist in C++ with Bayesian linear regression

## Usage

```
bayeslm(y, x, iters = 1000L)
```

## Arguments

<code>y</code>	A numeric vector – the response
<code>x</code>	A numeric matrix – the explanatory variables; note this assumes you have included a column of ones if you intend there to be an intercept.
<code>iters</code>	An integer vector of length one, the number of posterior draws desired; the default is 1000.

## Details

To see an example of using RcppDist C++ functions in C++ code, we can code up a Bayesian linear regression with completely uninformative priors (such that estimates should be equivalent to classical estimates). The code to do so is as follows:

```
#include <RcppDist.h>
// or, alternatively,
// #include <RcppArmadillo.h>
// #include <mvnorm.h>

// [[Rcpp::depends(RcppArmadillo, RcppDist)]]

// [[Rcpp::export]]
```

```

Rcpp::List bayeslm(const arma::vec& y, const arma::mat& x,
                  const int iters = 1000) {
  int n = x.n_rows;
  int p = x.n_cols;
  double a = (n - p) / 2.0; // Shape for sigma^2's gamma distribution
  arma::mat xtx = x.t() * x; // X'X
  arma::mat xtxinv = xtx.i(); // (X'X)^-1
  arma::vec mu = xtxinv * x.t() * y; // Mean of beta draws
  arma::mat px = x * xtxinv * x.t(); // Projector matrix of X
  double ssq = arma::as_scalar(y.t() * (arma::eye(n, n) - px) * y); // s^2
  ssq *= (1.0 / (n - p));
  double b = 1.0 / (a * ssq); // Scale for sigma^2's gamma distribution
  arma::mat beta_draws(iters, p); // Object to store beta draws in
  Rcpp::NumericVector sigmasq_draws(iters); // Stores sigma^2 draws
  for ( int iter = 0; iter < iters; ++iter ) {
    double sigmasq = 1.0 / R::rgamma(a, b); // Draw a sigma^2 value
    sigmasq_draws[iter] = sigmasq;
    // Here we can just use RcppDist's multivariate normal generator
    // to draw a beta value given our last sigma^2 draw
    beta_draws.row(iter) = rmvnorm(1, mu, xtxinv * sigmasq);
  }
  return Rcpp::List::create(Rcpp::_["beta_draws"] = beta_draws,
                           Rcpp::_["sigmasq_draws"] = sigmasq_draws);
}

```

### Value

A list of length two; the first element is a numeric matrix of the beta draws and the second element is a numeric vector of the sigma squared draws

### Examples

```

# Simulate data
set.seed(123)
n <- 30
x <- cbind(1, matrix(rnorm(n*3), ncol = 3))
beta <- matrix(c(10, -2, -1, 3), nrow = 4)
sigma <- 1.5
y <- x %*% beta + rnorm(n, mean = 0, sd = sigma)
# Run the models
freqmod <- lm(y ~ x[, -1])
bayesmod <- bayeslm(y, x)
# Check the estimates
estimates <- data.frame(
  truth = c(beta, sigma^2),
  freqmod = c(coef(freqmod), sigma(freqmod)^2),
  bayesmod = c(apply(bayesmod$beta_draws, 2, mean),
              mean(bayesmod$sigmasq_draws))
)
rownames(estimates) <- c("beta_0", "beta_1", "beta_2", "beta_3", "sigmasq")
print(estimates, digits = 2)

```

---

RcppDist

*RcppDist*

---

## Description

'Rcpp' Integration of Additional Probability Distributions

## Details

The 'Rcpp' package provides a C++ library to make it easier to use C++ with R. R and 'Rcpp' provide functions for a variety of statistical distributions. Several R packages make functions available to R for additional statistical distributions. However, to access these functions from C++ code, a costly call to the R functions must be made.

'RcppDist' provides a header-only C++ library with functions for additional statistical distributions that can be called from C++ when writing code using 'Rcpp' or 'RcppArmadillo'. Functions are available that return a 'NumericVector' as well as doubles, and for multivariate or matrix distributions, 'Armadillo' vectors and matrices. RcppDist provides functions for the following distributions:

- The four parameter beta distribution
- The Laplace distribution
- The location-scale t distribution
- The truncated normal distribution
- The truncated t distribution
- A truncated location-scale t distribution
- The triangle distribution
- The multivariate normal distribution\*
- The multivariate t distribution\*
- The Wishart distribution\*
- And the inverse Wishart distribution\*.

Distributions marked with an asterisk rely also on RcppArmadillo.

For more information on using 'RcppDist' functions in your C++ code, please consult the vignette via `vignette("RcppDist")`; the vignette explains how to link to the package and include the headers, which header files provide which functions, and also provides all function declarations (so that you can see the function and argument names and return/argument types; the arguments are also described in reasonable detail). You can also see an example of using the multivariate normal generator provided by 'RcppDist' in the function [bayeslm](#).

## Author(s)

JB Duck-Mayr

# Index

bayeslm, [2](#), [4](#)

RcppDist, [4](#)

RcppDist-package (RcppDist), [4](#)