Package: RcppBlaze (via r-universe)

November 2, 2024

Type Package

Title 'Rcpp' Integration for the 'Blaze' High-Performance 'C++' Math Library

Version 1.0.1

Date 2024-04-27

Maintainer Ching-Chuan Chen <zw12356@gmail.com>

URL https://github.com/Chingchuan-chen/RcppBlaze, https://bitbucket.org/blaze-lib/blaze

BugReports https://github.com/Chingchuan-chen/RcppBlaze/issues

Description Blaze is an open-source, high-performance 'C++' math library for dense and sparse arithmetic. With its state-of-the-art Smart Expression Template implementation Blaze combines the elegance and ease of use of a domain-specific language with HPC-grade performance, making it one of the most intuitive and fastest 'C++' math libraries available. The 'RcppBlaze' package includes the header files from the 'Blaze' library with disabling some functionalities related to link to the thread and system libraries which make 'RcppBlaze' be a header-only library. Therefore, users do not need to install 'Blaze'.

Depends R (>= 4.2.0)

Imports Rcpp (>= 1.0.0), Matrix (>= 1.5-0)

LinkingTo Rcpp

Suggests MatrixExtra, tinytest, microbenchmark

LazyLoad yes

Encoding UTF-8

License BSD_3_clause + file LICENSE

RoxygenNote 7.3.1

Repository https://chingchuan-chen.r-universe.dev

RemoteUrl https://github.com/chingchuan-chen/rcppblaze

RemoteRef HEAD

RemoteSha c52cac4eddc33cda5c3d242d8649616353106dff

Contents

`RcppBlaze-package` *RcppBlaze - 'Rcpp' Integration for the 'Blaze' High-Performance 'C++' Math Library*

Description

RcppBlaze constructs a bridge between **R** and **Blaze**.

Details

Blaze is an open-source, high-performance **C++** math library for dense and sparse arithmetic. With its state-of-the-art Smart Expression Template implementation **Blaze** combines the elegance and ease of use of a domain-specific language with HPC-grade performance, making it one of the most intuitive and fastest **C++** math libraries available. The **RcppBlaze** package includes the header files from the **Blaze** library with disabling some functionalities related to link to the thread and system libraries which make **RcppBlaze** be a header-only library. Therefore, users do not need to install **Blaze**.

Using RcppBlaze

To use **RcppBlaze** in your package, there are some important steps:

1. Include the '`RcppBlaze.h`' header file, which also includes '`blaze/Blaze.h`'.
2. Import `Rcpp`, LinkingTo `Rcpp` and `RcppBlaze` by adding these lines to the '`DESCRIPTION`' file:

```
Imports: Rcpp (>= 1.0.0)
LinkingTo: Rcpp, RcppBlaze
```

3. Link against the `BLAS` and `LAPACK` libraries, by adding following two lines in the '`Makevars`' and '`Makevars.win`' files:

```
PKG_CXXFLAGS=$(SHLIB_OPENMP_CXXFLAGS)
PKG_LIBS = $(LAPACK_LIBS) $(BLAS_LIBS) $(FLIBS) $(SHLIB_OPENMP_CXXFLAGS)
```

4. Since there are conflicted definitions between **R** and **blaze** which is `TRUE` and `FALSE`. You have to write the initializing function for **C/C++** code which the function is named after `R_init_YourPackageName` You can refer to our another package, https://github.com/ChingChuan-Chen/RcppLbfgsBlaze for example.

Notes

1. If you would like to enable Boost threads support, you need to import **BH** package in your DESCRIPTION.
2. `CompressedVector` and `CompressedMatrix` only support `int`, `float` and `double` types.

Author(s)

For RcppBlaze: Ching-Chuan Chen Maintainer: Ching-Chuan Chen <zw12356@gmail.com> For blaze: Klaus Iglberger, Georg Hager, Christian Godenschwager, Tobias Scharpff

References

1. Blaze project: https://bitbucket.org/blaze-lib/blaze.
2. K. Iglberger, G. Hager, J. Treibig, and U. Ruede: Expression Templates Revisited: A Performance Analysis of Current Methodologies. SIAM Journal on Scientific Computing, 34(2): C42–C69, 2012, doi:10.1137/110830125.
3. K. Iglberger, G. Hager, J. Treibig, and U. Ruede, High Performance Smart Expression Template Math Libraries. Proceedings of the 2nd International Workshop on New Algorithms and Programming Models for the Manycore Era (APMM 2012) at HPCS 2012, doi:10.1109/HPCSim.2012.6266939.

See Also

Useful links:

- https://github.com/Chingchuan-chen/RcppBlaze
- https://bitbucket.org/blaze-lib/blaze
- Report bugs at https://github.com/Chingchuan-chen/RcppBlaze/issues

---

`blaze_set_num_threads` *Set/Get the Number of Threads used in blaze*

Description

Set/Get the Number of Threads used in blaze

Usage

```
blaze_set_num_threads(n)

blaze_get_num_threads()
```

Arguments

`n`	The number of threads to set in blaze.

Value

`blaze_get_threads` returns an integer and `blaze_set_threads` returns nothing.

See Also

blaze wiki: https://bitbucket.org/blaze-lib/blaze/wiki/Shared%20Memory%20Parallelization.

`blaze_set_seed` *Set/Get the random number generator for blaze with given seed*

Description

Set/Get the random number generator for blaze with given seed

Usage

```
blaze_set_seed(seed)

blaze_get_seed()
```

Arguments

`seed`	A positive integer to specify the seed value for the random number generator.

Value

No return value.

`blaze_version` *The version of Blaze used in RcppBlaze*

Description

To return the version of Blaze used in RcppBlaze.

Usage

```
blaze_version(single)
```

Arguments

`single`	A logical value indicates which type to return. If TRUE, it returns an integer. If FALSE, it returns a named vector.

Value

A number or a named vector to represent the version of `blaze` depending on the input, `single`.

See Also

Blaze header file `blaze/system/Version.h`.

Examples

```
blaze_version(FALSE)
```

`fastLmPure` *linear model fitting function based on RcppBlaze*

Description

`fastLmPure` provides the estimates of the linear model based on **RcppBlaze**.

Usage

```
fastLmPure(X, y, type)
```

Arguments

`X`	A model matrix.
`y`	A response vector.
`type`	A integer. 0 is QR solver, 1 is LDLT solver, 2 is LLT sovler and 3 is LU solver.

Details

`fastLm` estimates the linear model using the `solve`.

Value

A list containing coefficients, standard errors, rank of model matrix, degree of freedom of residuals, residuals, the standard deviation of random errors and fitted values.

Examples

```
# according to fastLm example in RcppArmadillo
data(trees, package="datasets")
flm <- fastLmPure(cbind(1, log(trees$Girth)), log(trees$Volume), 0)
print(flm)
```

Index