

# cppEDM pyEDM rEDM

## Package Testing and Installation

### Table of Contents

Building cppEDM.....	2
Testing cppEDM.....	2
Building cppEDM/src & etc/Test.cc on Windows.....	4
Building pyEDM.....	5
Testing pyEDM.....	6
Building rEDM.....	8
Testing rEDM.....	9
rEDM CRAN.....	11
rEDM Documentation Utilities.....	13
pyEDM PyPI.....	14

---

## Building cppEDM

---

To build cppEDM

```
cd cppEDM/src
make
```

---

## Testing cppEDM

---

1) Basic numerical checks are done in cppEDM/tests by the programs:

CCMTest.cc DateTimeTest.cc MultiviewTest.cc SimplexTest.cc TestCommon.cc SMapTest.cc

Tests can be built and run with the ./run script. PASS/FAIL is reported on the console:

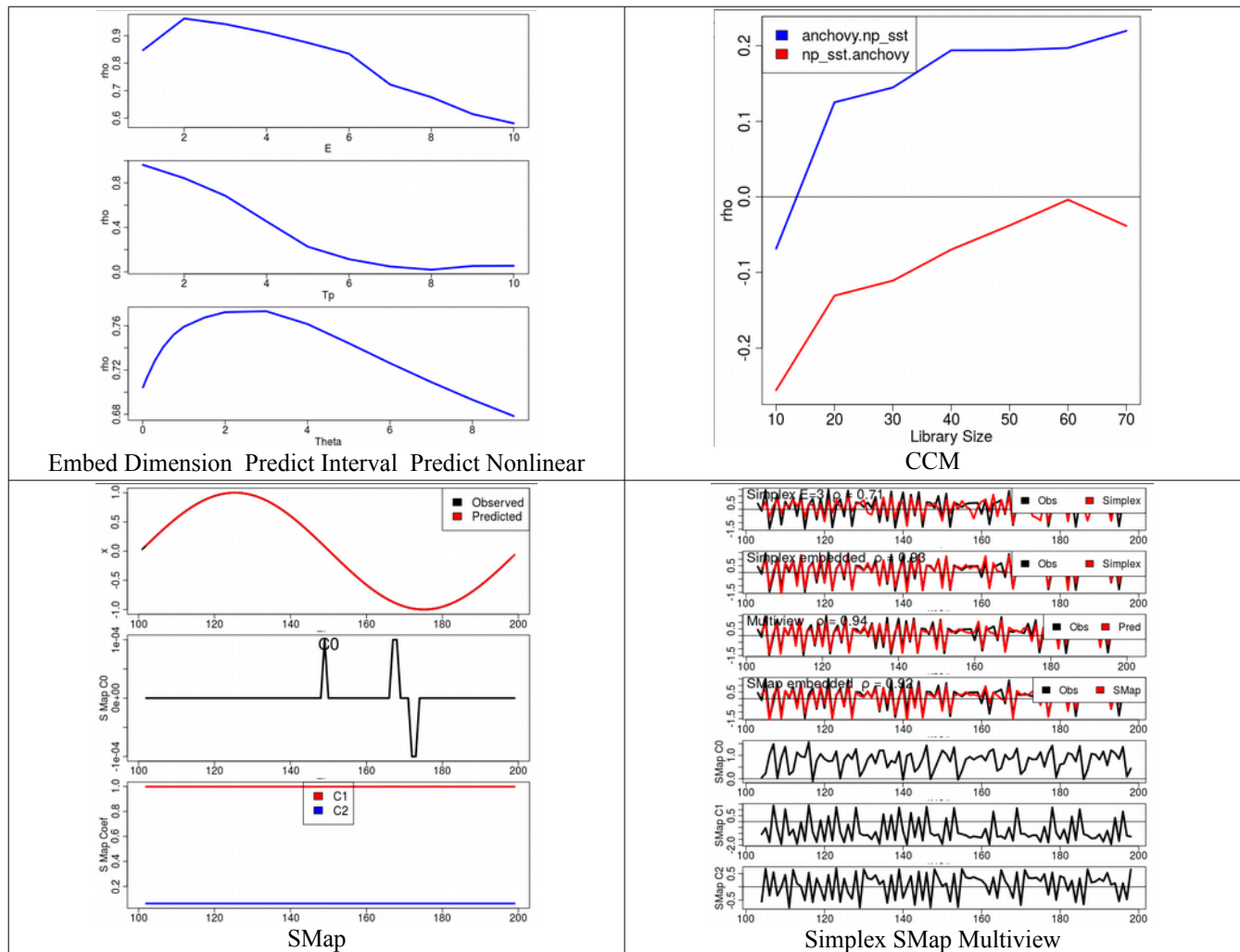
```
tests> ./run
```

```
g++ TestCommon.cc -c -std=c++11 -D
PRINT_DIFFERENCE_IN_RESULTS -lstdc++ -L../lib/
-I../src/ -lEDM -lpthread -llapack
g++ SimplexTest.cc -o SimplexTest -std=c++11 -D
PRINT_DIFFERENCE_IN_RESULTS -lstdc++ -L../lib/
-I../src/ -lEDM -lpthread -llapack TestCommon.o
g++ TestCommonTest.cc -o TestCommonTest -std=c++11 -D
PRINT_DIFFERENCE_IN_RESULTS -lstdc++ -L../lib/
-I../src/ -lEDM -lpthread -llapack TestCommon.o
g++ SMapTest.cc -o SMapTest -std=c++11 -D
PRINT_DIFFERENCE_IN_RESULTS -lstdc++ -L../lib/
-I../src/ -lEDM -lpthread -llapack TestCommon.o
g++ CCMTest.cc -o CCMTest -std=c++11 -D
PRINT_DIFFERENCE_IN_RESULTS -lstdc++ -L../lib/
-I../src/ -lEDM -lpthread -llapack TestCommon.o
g++ MultiviewTest.cc -o MultiviewTest -std=c++11 -D
PRINT_DIFFERENCE_IN_RESULTS -lstdc++ -L../lib/
-I../src/ -lEDM -lpthread -llapack TestCommon.o
g++ DateTimeTest.cc -c -std=c++11 -D
PRINT_DIFFERENCE_IN_RESULTS -lstdc++ -L../lib/
-I../src/ -lEDM -lpthread -llapack
g++ DateTimeTest.cc -o DateTimeTest -std=c++11 -D
PRINT_DIFFERENCE_IN_RESULTS -lstdc++ -L../lib/
-I../src/ -lEDM -lpthread -llapack TestCommon.o
```

```
Test: block_3sp.csv embedded data test
TEST PASSED. All rows same.
Test: block_3sp.csv dynamic embedding test
TEST PASSED. All rows same.
Test: S12CD-S333 ISO datetime
TEST PASSED. All rows same.
Multiview() Set view sample size to 9
Test: Multiview combos test
TEST PASSED. All rows same.
Test: Multiview prediction test
TEST PASSED. All rows same.
Test: circle.csv test
TEST PASSED. All rows same.
Test: block_3sp test
TEST PASSED. All rows same.
Test: CCM sardine anchovy_sst test
TEST PASSED. All rows same.
```

2) A series of graphical tests are run by the cppEDM/etc/Test.cc program, and rendered with the R application PlotTest.R. Carefully check that the graphical output matches the images shown below. The CCM test will not match exactly, but the relative behavior should be the same as shown.

```
cd cppEDM/etc
g++ Test.cc -o Test -std=c++11 -I../src -L../lib -lstdc++ -lEDM -lpthread -llapack -O3
./Test
R
> source('PlotTest.R')
> Run()
> Clean()
```



---

## Building cppEDM on Windows

---

This has been found to work on Windows 10 with MSVC 2019 build tools and mingw.

Build cppEDM/src:

```
nmake /f makefile.windows
```

Compile cppEDM/etc/Test.cc into Test.obj:

```
cl /c Test.cc /EHsc /MD /I../src
```

Download .lib and .dll from Windows for LAPACKE:

<https://icl.cs.utk.edu/lapack-for-windows/lapack/#lapacke>

Copy .dll and .lib from LAPACKE\_examples.zip into ../lapacke

Link Test.obj into Test.exe:

```
link /OUT:Test.exe /LIBPATH:../lib /LIBPATH:../lapacke EDM.lib  
liblapack.lib Test.obj
```

Get missing libraries for LAPACK legacy:

Downloaded libgfortran-3.dll into ../lapacke

<https://www.opendll.com/index.php?file-download=libgfortran-3.dll&arch=32bit>

Downloaded libwinpthread-1.dll into ../lapacke

<https://wikidll.com/mingw-w64/libwinpthread-1-dll>

Set PATH to find the lapacke and mingw dll's:

```
PATh=../lapacke;C:\MINGW\BIN;%PATh%
```

Run Test.exe

---

## Building pyEDM

---

pyEDM can be built locally in two steps:

- 1) Build the cppEDM libEDM.a in pyEDM/cppEDM/src
- 2) Build and install the pyEDM package with python pip

1) From pyEDM/

```
cd cppEDM/src
make
```

2) From pyEDM/cppEDM/src

```
cd ../..
python -m pip install . --user
```

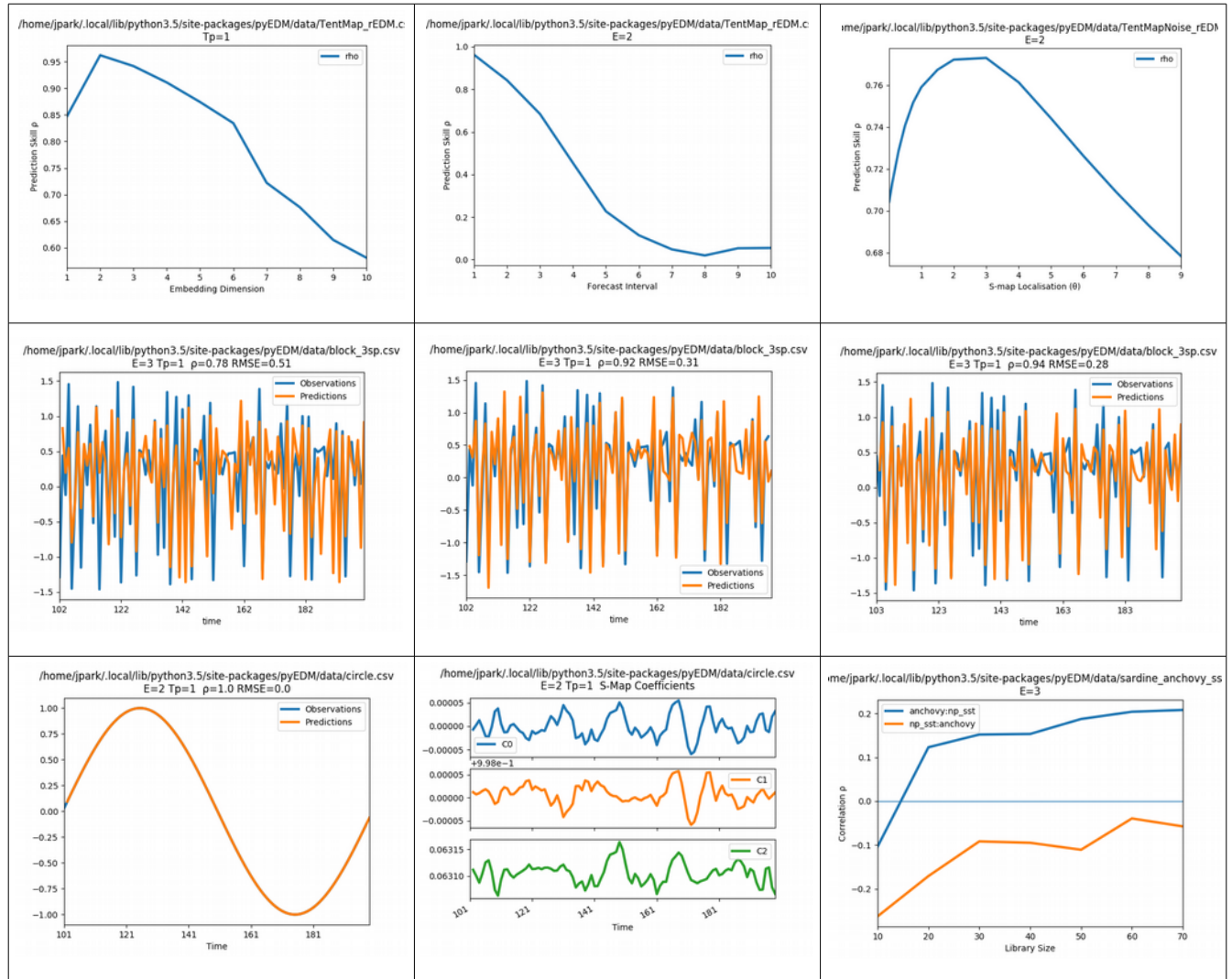
```
Processing pyEDM
Requirement already satisfied: pybind11>=2.3 in /usr/local/lib/python3.5/dist-packages (from pyEDM===-
version-.-.-1.2.1.1-) (2.3.0)
Requirement already satisfied: pandas>=0.20.3 in /usr/local/lib/python3.5/dist-packages (from pyEDM===-
version-.-.-1.2.1.1-) (0.24.2)
Requirement already satisfied: matplotlib>=2.2 in /usr/local/lib/python3.5/dist-packages (from
pyEDM===-version-.-.-1.2.1.1-) (2.2.2)
Requirement already satisfied: numpy>=1.12.0 in /usr/local/lib/python3.5/dist-packages (from
pandas>=0.20.3->pyEDM===-version-.-.-1.2.1.1-) (1.14.5)
Requirement already satisfied: python-dateutil>=2.5.0 in /usr/local/lib/python3.5/dist-packages (from
pandas>=0.20.3->pyEDM===-version-.-.-1.2.1.1-) (2.7.3)
Requirement already satisfied: pytz>=2011k in /usr/local/lib/python3.5/dist-packages (from
pandas>=0.20.3->pyEDM===-version-.-.-1.2.1.1-) (2018.4)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in
/usr/local/lib/python3.5/dist-packages (from matplotlib>=2.2->pyEDM===-version-.-.-1.2.1.1-) (2.2.0)
Requirement already satisfied: six>=1.10 in /usr/local/lib/python3.5/dist-packages (from
matplotlib>=2.2->pyEDM===-version-.-.-1.2.1.1-) (1.11.0)
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.5/dist-packages (from
matplotlib>=2.2->pyEDM===-version-.-.-1.2.1.1-) (1.0.1)
Requirement already satisfied: cyclor>=0.10 in /usr/local/lib/python3.5/dist-packages (from
matplotlib>=2.2->pyEDM===-version-.-.-1.2.1.1-) (0.10.0)
Requirement already satisfied: setuptools in /usr/local/lib/python3.5/dist-packages (from
kiwisolver>=1.0.1->matplotlib>=2.2->pyEDM===-version-.-.-1.2.1.1-) (39.2.0)
```

```
Building wheels for collected packages: pyEDM
  Building wheel for pyEDM (setup.py) ... done
  Stored in directory: /tmp/pip-ephem-wheel-cache-
0npnelyv/wheels/93/f6/91/fe8aef2eef3cef101cf85c751f4d8e3251ed009723a855b6af
Successfully built pyEDM
Installing collected packages: pyEDM
  Found existing installation: pyEDM -version-.-.-1.2.1.1-
  Uninstalling pyEDM--version-.-.-1.2.1.1-:
    Successfully uninstalled pyEDM--version-.-.-1.2.1.1-
Successfully installed pyEDM--version-.-.-1.2.1.1-
```

## Testing pyEDM

The pyEDM/pyEDM/tests.examples.py program runs a series of tests for the python wrapper and interface. The CCM test will not be numerically equivalent, but must have the same behavior.

```
cd pyEDM/tests/
./examples.py
```



PyEDM python unittests are run from pyEDM/tests/ with:

```
python -m unittest discover
```

```
--- CCM ---
Parameters::Validate(): Set knn = 4 (E+1) for Simplex.
cppEDM Version 1.2.1 2020-02-05
CrossMap(): Simplex cross mapping from anchovy to np_sst  E=3  knn=4  Library range: [10 75 5]
10 15 20 25 30 35 40 45 50 55 60 65 70 75

cppEDM Version 1.2.1 2020-02-05
CrossMap(): Simplex cross mapping from np_sst to anchovy  E=3  knn=4  Library range: [10 75 5]
10 15 20 25 30 35 40 45 50 55 60 65 70 75

cppEDM Version 1.2.1 2020-02-05
.--- Multiview ---
Multiview() Set view sample size to 9
.--- Simplex embedded = False ---
.--- Simplex embedded = True ---
.--- S-map circle embedded = True ---
.--- S-map block_3sp embedded = True ---
.
-----
Ran 6 tests in 0.170s

OK
```

```
tests> rm -rf __pycache__/
```

---

## Building rEDM

---

rEDM can be locally built with R CMD in rEDM/.

First, you may wish to cleanup a previous build:

```
cd rEDM
rm -rf src/*.o src/rEDM.so src/cppEDM/lib/libEDM.a

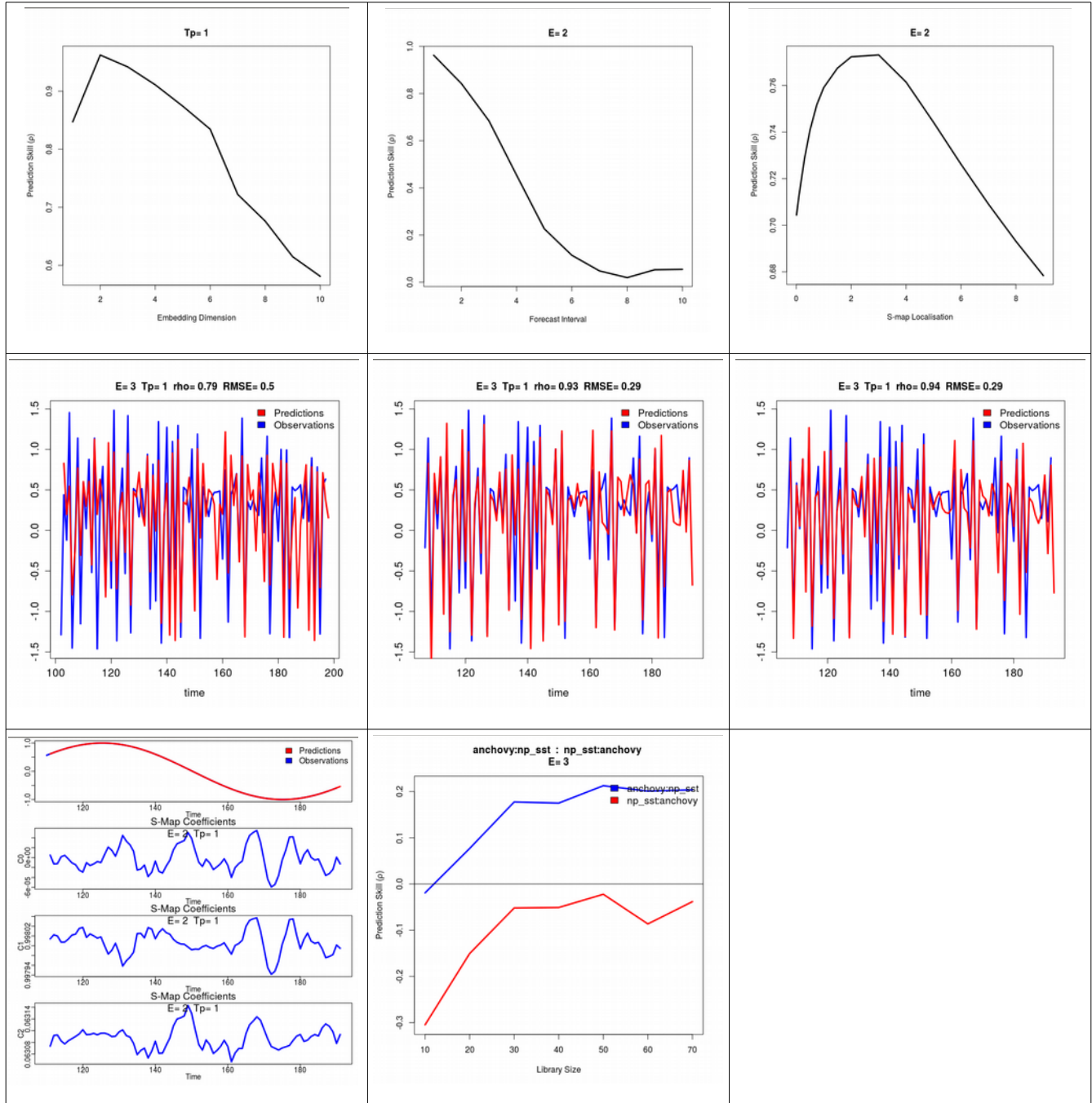
R CMD INSTALL .
* installing to library '/usr/local/lib/R/site-library'
* installing *source* package 'rEDM' ...
** libs
g++ -std=gnu++11 -I/usr/share/R/include -DNDEBUG -I ./cppEDM/src/ -I"/usr/local/lib/R/site-library/Rcpp/include" -fpic -g -O2 -fstack-protector-strong -Wformat -Werror=format-security -Wdate-time -D_FORTIFY_SOURCE=2 -g -c CCM.cpp -o CCM.o
...
(cd ./cppEDM/src/; make; make clean)
make[1]: Entering directory 'rEDM/src/cppEDM/src'
g++ -c Common.cc -std=c++11 -DCCM_THREADED -DMULTIVIEW_VALUES_OVERLOAD -O3 -fpic
g++ -c AuxFunc.cc -std=c++11 -DCCM_THREADED -DMULTIVIEW_VALUES_OVERLOAD -O3 -fpic
...
ar -rcs libEDM.a Common.o AuxFunc.o DateTimeUtil.o Parameter.o Embed.o Interface.o Neighbors.o Simplex.o Eval.o CCM.o Multiview.o SMap.o
cp libEDM.a ../lib/
make[1]: Leaving directory 'rEDM/src/cppEDM/src'
make[1]: Entering directory 'rEDM/src/cppEDM/src'
rm -f Common.o AuxFunc.o DateTimeUtil.o Parameter.o Embed.o Interface.o Neighbors.o Simplex.o Eval.o CCM.o Multiview.o SMap.o libEDM.a
make[1]: Leaving directory 'rEDM/src/cppEDM/src'
g++ -std=gnu++11 -shared -L/usr/lib/R/lib -Wl,-Bsymbolic-functions -Wl,-z,relro -o rEDM.so CCM.o ComputeError.o DataFrame.o Embed.o EmbedDim.o Multiview.o PredictInterval.o PredictNL.o RcppEDMCommon.o RcppExports.o SMap.o Simplex.o -L ./cppEDM/lib/ -lEDM -llapack -L/usr/lib/R/lib -lR
installing to /usr/local/lib/R/site-library/rEDM/libs
** R
** data
*** moving datasets to lazyload DB
** inst
** preparing package for lazy loading
** help
*** installing help indices
*** copying figures
** building package indices
** installing vignettes
  'rEDM-tutorial.Rmd' using 'UTF-8'
** testing if installed package can be loaded
* DONE (rEDM)
```



## Testing rEDM

The rEDM/R/Examples.R program executes Rcpp wrapper graphical tests. The CCM test will not be numerically equivalent, but must have the same behavior.

```
cd R/
R
> source("Examples.R")
> Examples()
```



rEDM unittest can be run from rEDM/tests

```
cd tests
R
> source('testthat.R')

[1] "Error: ColumnsInDataFrame(): dataFrame is not valid."
[1] "Error: ColumnsInDataFrame(): Target not found."
[1] "Error: ColumnsInDataFrame(): Target None not found."
[1] "Error: ColumnsInDataFrame(): dataFrame is not valid."
[1] "Error: ColumnsInDataFrame(): Target None not found."
[1] "Error: ColumnsInDataFrame(): dataFrame is not valid."
[1] "Error: ColumnsInDataFrame(): Target not found."
Multiview() Set view sample size to 9
[1] "Error: ColumnsInDataFrame(): dataFrame is not valid."
[1] "Error: ColumnsInDataFrame(): Target None not found."
[1] "Error: ColumnsInDataFrame(): dataFrame is not valid."
[1] "Error: ColumnsInDataFrame(): Target None not found."
[1] "Error: ColumnsInDataFrame(): dataFrame is not valid."
[1] "Error: ColumnsInDataFrame(): dataFrame is not valid."
===== testthat results =====
[ OK: 64 | SKIPPED: 0 | WARNINGS: 0 | FAILED: 0 ]
```

---

## rEDM CRAN

---

To test and prepare rEDM for CRAN, use the devtools package.

1) CRAN build check on \*nix :

```
R
> devtools::check()

— Building ————— rEDM —
Setting env vars:
● CFLAGS      : -Wall -pedantic -fdiagnostics-color=always
● CXXFLAGS    : -Wall -pedantic -fdiagnostics-color=always
● CXX11FLAGS  : -Wall -pedantic -fdiagnostics-color=always
—
✓ checking for file 'rEDM.build/DESCRIPTION' ...
— preparing 'rEDM':
✓ checking DESCRIPTION meta-information ...
— cleaning src
— installing the package to build vignettes
✓ creating vignettes (1m 16.9s)
— building 'rEDM_1.2.2.tar.gz'

— Checking ————— rEDM —
Setting env vars:
● _R_CHECK_CRAN_INCOMING_USE_ASPELL_ : TRUE
● _R_CHECK_CRAN_INCOMING_REMOTE_    : FALSE
● _R_CHECK_CRAN_INCOMING_           : FALSE
● _R_CHECK_FORCE_SUGGESTS_          : FALSE
— R CMD check —————
— using R version 3.4.4 (2018-03-15)
— using platform: x86_64-pc-linux-gnu (64-bit)
— using session charset: UTF-8
— using options '--no-manual --as-cran'
✓ checking for file 'rEDM/DESCRIPTION'
— checking extension type ... Package
— this is package 'rEDM' version '1.2.2'
✓ checking package namespace information
✓ checking package dependencies (2.3s)
✓ checking dependencies in R code ...
✓ checking compilation flags in Makevars ...
✓ checking compiled code ...
✓ checking sizes of PDF files under 'inst/doc' (849ms)
✓ checking installed files from 'inst/doc' ...
✓ checking files in 'vignettes'
✓ checking examples (1.1s)
✓ checking for unstated dependencies in vignettes ...
✓ checking package vignettes in 'inst/doc' ...
✓ checking re-building of vignette outputs (13.6s)
See
  '/tmp/RtmPgTq4pn/rEDM.Rcheck/00check.log'
for details.

— R CMD check results ————— rEDM 1.2.2 —
Duration: 1m 32.4s

> checking installed package size ... NOTE
  installed size is 8.7Mb
  sub-directories of 1Mb or more:
    libs 7.7Mb

0 errors ✓ | 0 warnings ✓ | 1 note ✖
```

2) CRAN build check on Windows using cloud servers. This will email the build results to the package maintainer address..

R

```
> devtools::check_win_release()
```

```
Building windows version of rEDM (1.2.2) for R-release with win-builder.r-project.org.  
Email results to JosephPark@IEEE.org?
```

```
1: I forget  
2: Of course  
3: No way
```

```
Selection: 2
```

```
✓ checking for file 'rEDM.build/DESCRIPTION' ...  
- preparing 'rEDM':  
✓ checking DESCRIPTION meta-information ...  
- cleaning src  
- installing the package to build vignettes  
✓ creating vignettes (lm 20.6s)  
- building 'rEDM_1.2.2.tar.gz'
```

```
[02:55 PM (2020-02-18)] Check JosephPark@IEEE.org for a link to the built package in 15-30 mins.
```

```
win-builder.r-project.org - /PYKlZmqsszt1/  
18.02.2020 16:10 4500 00check.log  
18.02.2020 16:10 14129 00install.out  
18.02.2020 16:10 <dir> examples_and_tests  
18.02.2020 16:10 2067513 rEDM_1.2.2.zip
```

```
* installing *source* package 'rEDM' ...  
** using staged installation  
** libs
```

```
*** arch - i386
```

```
d:/Compiler/gcc-4.9.3/mingw_32/bin/g++ -std=gnu++11 -I"D:/RCompile/recent/R-3.6.2/include" -DNDEBUG  
-I./cppEDM/src -I../ -I"d:/RCompile/CRANpkg/lib/3.6/Rcpp/include"  
-I"d:/RCompile/CRANpkg/lib/3.6/RcppThread/include" -I"d:/Compiler/gcc-4.9.3/local330/include" -O2  
-Wall -mtune=core2 -c CCM.cpp -o CCM.o
```

```
...
```

```
installing to d:/RCompile/CRANquest/R-release/lib/00LOCK-rEDM/00new/rEDM/libs/i386
```

```
*** arch - x64
```

```
d:/Compiler/gcc-4.9.3/mingw_64/bin/g++ -m64 -std=gnu++11 -I"D:/RCompile/recent/R-3.6.2/include"  
-DNDEBUG -I./cppEDM/src -I../ -I"d:/RCompile/CRANpkg/lib/3.6/Rcpp/include"  
-I"d:/RCompile/CRANpkg/lib/3.6/RcppThread/include" -I"d:/Compiler/gcc-4.9.3/local330/include" -O2  
-Wall -mtune=core2 -c CCM.cpp -o CCM.o  
d:/Compiler/gcc-4.9.3/mingw_64/bin/g++ -m64 -shared -s -static-libgcc -o rEDM.dll tmp.def CCM.o  
ComputeError.o DataFrame.o Embed.o EmbedDim.o Multiview.o PredictInterval.o PredictNL.o RcppEDMCommon.o  
RcppExports.o SMap.o Simplex.o -L./cppEDM/src/ -lEDM -LD:/RCompile/recent/R-3.6.2/bin/x64 -lRlapack  
-Ld:/Compiler/gcc-4.9.3/local330/lib/x64 -Ld:/Compiler/gcc-4.9.3/local330/lib -LD:/RCompile/recent/R-  
3.6.2/bin/x64 -lR
```

```
installing to d:/RCompile/CRANquest/R-release/lib/00LOCK-rEDM/00new/rEDM/libs/x64
```

```
** R
```

```
** data
```

```
*** moving datasets to lazyload DB
```

```
** byte-compile and prepare package for lazy loading
```

```
** help
```

```
** installing vignettes
```

```
** testing if installed package can be loaded from temporary location
```

```
*** arch - i386
```

```
*** arch - x64
```

```
** testing if installed package can be loaded from final location
```

```
*** arch - i386
```

```
*** arch - x64
```

```
** testing if installed package keeps a record of temporary installation path
```

```
* MD5 sums
```

```
packaged installation of 'rEDM' as rEDM_1.2.2.zip
```

```
* DONE (rEDM)
```

3) Build the CRAN release file to upload to CRAN

```
> devtools::build()
```

---

## rEDM Documentation Utilities

---

Useful commands and rmarkdown package commands to build and convert documentation.

```
rmarkdown::render("rEDM-tutorial.Rmd", "pdf_document")  
rmarkdown::render("rEDM-tutorial.Rmd", "html_document")
```

```
R CMD Rd2pdf rEDM
```

```
R CMD Rdconv -t html ./rEDM/man/rEDM.Rd > rEDM.html
```

---

## pyEDM PyPI

---

The pyEDM package is distributed on the PyPI archives: <https://pypi.org/project/pyEDM/>

The version string must be different from the previously published one.

Increment the 4th element of `__version__ = "1.2.1.1"` in `pyEDM/pyEDM/__init.py`

Build the wheels

```
python setup.py bdist_wheel
```

Upload to PyPI using twine:

```
twine upload [wheel output location]
```