

# Package ‘STCYP’

May 7, 2026

**Title** Spatio-Temporal Crop Yield Prediction

**Version** 1.0.0

**Description** Provides crop yield and meteorological data for Ontario, Canada.

Includes functions for fitting and predicting data using spatio-temporal models, as well as tools for visualizing the results. The package builds upon existing R packages, including 'copula' (Hofert et al., 2025) <[doi:10.32614/CRAN.package.copula](https://doi.org/10.32614/CRAN.package.copula)>, and 'bsts' (Scott, 2024) <[doi:10.32614/CRAN.package.bsts](https://doi.org/10.32614/CRAN.package.bsts)>.

**License** MIT + file LICENSE

**Encoding** UTF-8

**RoxygenNote** 7.3.2

**Imports** bsts, copula, ggplot2, grDevices, rootSolve, stats

**Depends** R (>= 4.0.0)

**LazyData** true

**LazyDataCompression** xz

**Suggests** knitr, rmarkdown, testthat (>= 3.0.0),

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**NeedsCompilation** no

**Author** Marie Michaelides [aut],

Mélina Mailhot [aut],

Yongkun Li [aut, cre]

**Maintainer** Yongkun Li <[yongkun.li@concordia.ca](mailto:yongkun.li@concordia.ca)>

**Repository** CRAN

**Date/Publication** 2025-09-09 14:40:23 UTC

## Contents

clayton.theta . . . . .	2
copula_list . . . . .	3
data . . . . .	3

dt . . . . .	4
dynamic.rho . . . . .	5
dynamic.theta.clayton . . . . .	6
dynamic.theta.frank . . . . .	6
dynamic.theta.gumbel . . . . .	7
dynamic.theta.joe . . . . .	7
fit_bsts . . . . .	8
frank.theta . . . . .	8
GH.theta . . . . .	9
init_params_full . . . . .	9
joe.theta . . . . .	10
log_likelihood_noGEV_3d . . . . .	10
medoid_names . . . . .	11
n_test . . . . .	12
n_train . . . . .	12
plot_forecast . . . . .	12
plot_forecast_compare . . . . .	13
simul_fun_noGEV_3d . . . . .	14
time . . . . .	16
time_test . . . . .	16
time_train . . . . .	17
u . . . . .	17
y_test . . . . .	18
y_train . . . . .	18
z_test . . . . .	19
z_train . . . . .	19
<b>Index</b>	<b>20</b>

---

clayton.theta	<i>Compute Clayton Copula Parameter from Kendall's Tau</i>
---------------	--

---

### Description

Computes the Clayton copula dependence parameter based on Kendall's tau.

### Usage

```
clayton.theta(tau)
```

### Arguments

tau                      Numeric, Kendall's tau correlation coefficient.

### Value

Numeric, estimated Clayton copula parameter.

**Examples**

```
clayton.theta(mean(cor(cbind(u[[1]], u[[2]], u[[3]]), method = "kendall")))
```

---

copula_list	<i>Supported copula types</i>
-------------	-------------------------------

---

**Description**

A list containing supported copula types.

**Usage**

```
copula_list
```

**Format**

A list of copula types.

```
copulas "Gaussian" "Clayton" "Frank" "Gumbel" "Joe"
```

---

data	<i>Real crop yield and meteorological data of 24 regions for Ontario, Canada from 1950 to 2022 and anticipated data from 2023 to 2100.</i>
------	--

---

**Description**

Real crop yield and meteorological data of 24 regions for Ontario, Canada from 1950 to 2022 and anticipated data from 2023 to 2100.

**Usage**

```
data
```

**Format**

A data frame with 1752 rows and 27 variables:

**time** chr: year from 1950-2022

**CD** chr: 24 subregions

**lat** num: latitude

**lon** num: longitude

**yield** num: wheat crop yield per census division, in bushel/acre

**cdd** num: Annual maximum number of consecutive days with daily precipitation below 1mm (unit = days)

**cddcold** num: Annual cooling degree days above 18C (unit = degree\_days)

**dlyfrzthw** num: Annual number of days with a diurnal freeze-thaw cycle : tmax > 0 degc and tmin <= -1 degc

**firstfallfrost** num: First day of year with temperature below 0 degc for at least 1 days

**frostdays** num: Annual number of days with minimum daily temperature below 0C

**icedays** num: Annual number of days with maximum daily temperature below 0 degC

**nrcdd** num: The annual number of dry periods of 6 days and more, during which the maximal precipitation on a window of 6 days is under 1.0 mm

**prcptot** num: Annual total precipitation (unit = mm)

**r1mm** num: Annual number of days with daily precipitation over 1.0 mm/day

**r10mm** num: Annual number of days with daily precipitation over 10.0 mm/day

**r20mm** num: Annual number of days with daily precipitation over 20.0 mm/day

**rx1day** num: Annual maximum 1-day total precipitation (unit = mm)

**rx5day** num: Annual maximum 5-day total precipitation (unit = mm)

**tgmean** num: Annual mean of daily mean temperatures (unit = C degrees)

**tnmean** num: Annual mean of daily minimum temperatures (unit = C degrees)

**tnmin** num: Annual minimum of daily minimum temperatures (unit = C degrees)

**tr18** num: Annual number of tropical nights : defined as days with minimum daily temperature above 18 degc

**txmax** num: Annual minimum of daily maximum temperature (unit = C degrees)

**txmean** num: Annual mean of daily maximum temperature (unit = C degrees)

**txgt25** num: Annual number of days where daily maximum temperature exceeds 25 degC

**txgt27** num: Annual number of days where daily maximum temperature exceeds 27 degC

**txgt29** num: Annual number of days where daily maximum temperature exceeds 29 degC

### Source

ClimateData.ca

---

dt	<i>Selected data from year 1950 to 2022 and covariates including txgt27, tr18, cddcold, txgt29, and tnmean for case study.</i>
----	--

---

### Description

Selected data from year 1950 to 2022 and covariates including txgt27, tr18, cddcold, txgt29, and tnmean for case study.

### Usage

dt

**Format**

A data frame with 1752 rows and 10 variables:

**time** chr: year from 1950-2022

**CD** chr: 24 subregions

**lat** num: latitude

**lon** num: longitude

**yield** num: wheat crop yield per census division, in bushel/acre

**cddcold** num: Annual cooling degree days above 18C (unit = degree\_days)

**tnmean** num: Annual mean of daily minimum temperatures (unit = C degrees)

**tr18** num: Annual number of tropical nights : defined as days with minimum daily temperature above 18 degc

**txgt27** num: Annual number of days where daily maximum temperature exceeds 27 degC

**txgt29** num: Annual number of days where daily maximum temperature exceeds 29 degC

**Source**

ClimateData.ca

---

dynamic.rho

*Compute Dynamic Gaussian Copula Correlation Parameter (rho)*

---

**Description**

Computes the time-varying correlation parameter ( $\rho$ ) for a Gaussian copula.

**Usage**

```
dynamic.rho(params, lagged_rho, X_t)
```

**Arguments**

params	Numeric vector of parameters: omega, alpha, and gamma coefficients.
lagged_rho	Numeric, the previous rho value.
X_t	Numeric vector or matrix of covariates at time t.

**Value**

Numeric, estimated dynamic Gaussian copula correlation.

---

dynamic.theta.clayton *Compute Dynamic Clayton Copula Parameter*

---

**Description**

Computes the Clayton copula parameter dynamically based on lagged values and covariates.

**Usage**

```
dynamic.theta.clayton(params, lagged_theta, X_t)
```

**Arguments**

params	Numeric vector of parameters: omega, alpha, and gamma coefficients.
lagged_theta	Numeric, the previous theta value.
X_t	Numeric vector or matrix of covariates at time t.

**Value**

Numeric, estimated dynamic Clayton copula parameter.

---

dynamic.theta.frank *Compute Dynamic Frank Copula Parameter*

---

**Description**

Computes the Frank copula parameter dynamically based on lagged values and covariates.

**Usage**

```
dynamic.theta.frank(params, lagged_theta, X_t)
```

**Arguments**

params	Numeric vector of parameters: omega, alpha, and gamma coefficients.
lagged_theta	Numeric, the previous theta value.
X_t	Numeric vector or matrix of covariates at time t.

**Value**

Numeric, estimated dynamic Frank copula parameter.

---

dynamic.theta.gumbel    *Compute Dynamic Gumbel Copula Parameter*

---

**Description**

Computes the Gumbel copula parameter dynamically based on lagged values and covariates.

**Usage**

```
dynamic.theta.gumbel(params, lagged_theta, X_t)
```

**Arguments**

params	Numeric vector of parameters: omega, alpha, and gamma coefficients.
lagged_theta	Numeric, the previous theta value.
X_t	Numeric vector or matrix of covariates at time t.

**Value**

Numeric, estimated dynamic Gumbel copula parameter.

---

dynamic.theta.joe    *Compute Dynamic Joe Copula Parameter*

---

**Description**

Computes the Joe copula parameter dynamically based on lagged values and covariates.

**Usage**

```
dynamic.theta.joe(params, lagged_theta, X_t)
```

**Arguments**

params	Numeric vector of parameters: omega, alpha, and gamma coefficients.
lagged_theta	Numeric, the previous theta value.
X_t	Numeric vector or matrix of covariates at time t.

**Value**

Numeric, estimated dynamic Joe copula parameter.

fit\_bsts

*Fit a Bayesian Structural Time Series (BSTS) Model*

---

**Description**

Fits a BSTS model for a time series  $y$ , given a vector or matrix of covariates  $z$ .

**Usage**

```
fit_bsts(y, z, lags = 0, MCMC.iter = 5000)
```

**Arguments**

$y$	A numeric vector (time series response variable).
$z$	A numeric vector or matrix (covariates).
lags	Integer, number of lags for the autoregressive component.
MCMC.iter	Integer, number of MCMC iterations.

**Value**

A fitted BSTS model.

---

frank.theta

*Compute Frank Copula Parameter from Kendall's Tau*

---

**Description**

Computes the Frank copula dependence parameter based on Kendall's tau.

**Usage**

```
frank.theta(tau)
```

**Arguments**

tau	Numeric, Kendall's tau correlation coefficient.
-----	---

**Value**

Numeric, estimated Frank copula parameter.

---

GH.theta	<i>Compute Gumbel Copula Parameter from Kendall's Tau</i>
----------	---

---

**Description**

Computes the Gumbel-Hougaard copula dependence parameter based on Kendall's tau.

**Usage**

```
GH.theta(tau)
```

**Arguments**

tau                    Numeric, Kendall's tau correlation coefficient.

**Value**

Numeric, estimated Gumbel copula parameter.

**Examples**

```
GH.theta(mean(cor(cbind(u[[1]], u[[2]], u[[3]]), method = "kendall")))
```

---

init_params_full	<i>Initial Parameters for 3D Pseudo-Loglikelihood Estimation</i>
------------------	--

---

**Description**

Initial Parameters for 3D Pseudo-Loglikelihood Estimation

**Usage**

```
init_params_full
```

**Format**

A numeric vector of length  $(2 + M)$  where:

**omega** Baseline autoregressive coefficient.

**alpha** Parameter controlling variance.

**gamma1, gamma2, gamma3** Coefficients related to external factors.

---

joe.theta	<i>Compute Joe Copula Parameter from Kendall's Tau</i>
-----------	--

---

**Description**

Computes the Joe copula dependence parameter based on Kendall's tau.

**Usage**

```
joe.theta(tau)
```

**Arguments**

tau	Numeric, Kendall's tau correlation coefficient.
-----	---

**Value**

Numeric, estimated Joe copula parameter.

**Examples**

```
joe.theta(mean(cor(cbind(u[[1]], u[[2]], u[[3]]), method = "kendall")))
```

---

log_likelihood_noGEV_3d	<i>Log-Likelihood Function for 3D Copula Model</i>
-------------------------	--

---

**Description**

Computes the negative log-likelihood of a 3-dimensional copula model with a time-varying copula structure.

**Usage**

```
log_likelihood_noGEV_3d(params, u1, u2, u3, X_t, z1, z2, z3, copula)
```

**Arguments**

params	Numeric vector, model parameters.
u1	Numeric vector (length n_train), pseudo-observations for margin 1.
u2	Numeric vector (length n_train), pseudo-observations for margin 2.
u3	Numeric vector (length n_train), pseudo-observations for margin 3.
X_t	Numeric matrix (n_train x M), risk factors affecting copula parameters.
z1	Numeric matrix (n_train x M), observed data for margin 1.

z2	Numeric matrix (n_train x M), observed data for margin 2.
z3	Numeric matrix (n_train x M), observed data for margin 3.
copula	Character, specifying the copula type: "Clayton", "Frank", "Gumbel", "Joe", or "Gaussian".

**Value**

The negative log-likelihood value for optimization.

**Examples**

```
test_ll_3d <- log_likelihood_noGEV_3d(init_params_full,
                                     u[[1]],
                                     u[[2]],
                                     u[[3]],
                                     (z_train[[1]] + z_train[[2]] + z_train[[3]])/3,
                                     z_train[[1]],
                                     z_train[[2]],
                                     z_train[[3]],
                                     "Gaussian")
```

---

medoid\_names

*list containing Chatham-Kent, Lambton, and Wellington*

---

**Description**

list containing Chatham-Kent, Lambton, and Wellington

**Usage**

```
medoid_names
```

**Format**

An object of class character of length 3.

---

n_test	19
--------	----

---

**Description**

19

**Usage**

n\_test

**Format**

An object of class integer of length 1.

---

n_train	54
---------	----

---

**Description**

54

**Usage**

n\_train

**Format**

An object of class integer of length 1.

---

plot_forecast	<i>Plot Observed Data and BSTS Forecast</i>
---------------	---

---

**Description**

Creates a plot of observed data, forecasted values, and confidence intervals.

**Usage**

```
plot_forecast(  
  forecast,  
  data_train,  
  data_test,  
  time,  
  quant_high,  
  quant_low,  
  observed_col,  
  forecast_col,  
  title  
)
```

**Arguments**

forecast	A matrix of BSTS forecast samples.
data_train	Numeric vector, training data.
data_test	Numeric vector, test data.
time	Numeric vector, representing time indices.
quant_high	Numeric, upper quantile for confidence interval.
quant_low	Numeric, lower quantile for confidence interval.
observed_col	Character, color for observed data.
forecast_col	Character, color for forecasted data.
title	Character, title of the plot.

**Value**

A ggplot2 object.

---

plot\_forecast\_compare *Compare Forecasts from Two Models*

---

**Description**

Generates a time series plot comparing the forecasts from two models along with observed data.

**Usage**

```
plot_forecast_compare(  
  forecast1,  
  forecast2,  
  data_train,  
  data_test,  
  time,
```

```

    quant_high,
    quant_low,
    col1,
    title
  )

```

### Arguments

forecast1	Numeric matrix, forecasted values from the first model (columns: time points).
forecast2	Numeric matrix, forecasted values from the second model (columns: time points).
data_train	Numeric vector, training data used for modeling.
data_test	Numeric vector, actual test data for evaluation.
time	Numeric vector, representing the time points corresponding to the data.
quant_high	Numeric, upper quantile (e.g., 0.9) for confidence interval.
quant_low	Numeric, lower quantile (e.g., 0.1) for confidence interval.
col1	Character, color for observed data lines.
title	Character, title for the plot.

### Value

A ggplot2 object showing the forecast comparison.

---

simul_fun_noGEV_3d	<i>Function to optimize the full pseudo-loglikelihood and perform new forecasts</i>
--------------------	---

---

### Description

Function to optimize the full pseudo-loglikelihood and perform new forecasts

### Usage

```

simul_fun_noGEV_3d(
  nsim = 100,
  n_train,
  n_test,
  copula,
  init_params,
  fn,
  u1,
  u2,
  u3,
  z1_train,
  z2_train,
  z3_train,

```

```

    z1_test,
    z2_test,
    z3_test,
    X_t,
    y1_test,
    y2_test,
    y3_test,
    BSTS_1,
    BSTS_2,
    BSTS_3
)

```

### Arguments

nsim	Integer, number of simulation replications.
n_train	Integer, number of training observations.
n_test	Integer, number of test observations.
copula	Character, specifying the copula type: "Clayton", "Frank", "Gumbel", "Joe", or "Gaussian".
init_params	Numeric vector, initial parameter values for optimization.
fn	Function, log-likelihood function for parameter estimation.
u1	Numeric vector (n_train), first pseudo-observation for the copula.
u2	Numeric vector (n_train), second pseudo-observation for the copula.
u3	Numeric vector (n_train), third pseudo-observation for the copula.
z1_train	Numeric matrix (n_train x M), observed data for the first margin and sub-feature.
z2_train	Numeric matrix (n_train x M), observed data for the second margin and sub-feature.
z3_train	Numeric matrix (n_train x M), observed data for the third margin and sub-feature.
z1_test	Numeric matrix (n_test x M), true future data for the first margin and sub-feature.
z2_test	Numeric matrix (n_test x M), true future data for the second margin and sub-feature.
z3_test	Numeric matrix (n_test x M), true future data for the third margin and sub-feature.
X_t	Numeric matrix (n_train x M), risk factors for the dynamic copula parameter.
y1_test	Numeric vector (n_test), true future values for the first response variable.
y2_test	Numeric vector (n_test), true future values for the second response variable.
y3_test	Numeric vector (n_test), true future values for the third response variable.
BSTS_1	Fitted BSTS model for the first response variable.
BSTS_2	Fitted BSTS model for the second response variable.
BSTS_3	Fitted BSTS model for the third response variable.

**Value**

A list containing:

theta_simulated	Simulated copula parameters across replications.
y1_simulated	Simulated values for the first response variable.
y2_simulated	Simulated values for the second response variable.
y3_simulated	Simulated values for the third response variable.
MSE	Mean squared error for each simulation run.
optim_results	Results from the optimization process.

---

time	1950-2022
------	-----------

---

**Description**

1950-2022

**Usage**

time

**Format**

An object of class character of length 73.

---

time_test	2004-2022
-----------	-----------

---

**Description**

2004-2022

**Usage**

time\_test

**Format**

An object of class character of length 19.

---

time_train	1950-2003
------------	-----------

---

**Description**

1950-2003

**Usage**

time\_train

**Format**

An object of class character of length 54.

---

u	<i>Pseudo-Observations of BSTS Residuals for Crop Yield Forecasting</i>
---	---

---

**Description**

Pseudo-Observations of BSTS Residuals for Crop Yield Forecasting

**Usage**

u

**Format**A matrix with dimensions  $(n_{train}, D)$ :**n\_train** Number of time points used in the training set.**D** Number of regions analyzed (Chatham-Kent, Lambton, Wellington).**Source**

Derived from residuals of BSTS models fitted to crop yield data.

---

y\_test

*Crop Yield Data for Testing in BSTS Models*


---

**Description**

Crop Yield Data for Testing in BSTS Models

**Usage**

y\_test

**Format**

A matrix with dimensions  $(n_{test}, D)$ :

**n\_train** Number of time points used in the test set.

**D** Number of regions analyzed (Chatham-Kent, Lambton, Wellington).

**Source**

Historical crop yield records from ClimateData.ca.

---

y\_train

*Crop Yield Training Matrix*


---

**Description**

Training crop-yield data used for BSTS models.

**Usage**

y\_train

**Format**

A numeric matrix with n\_train rows and D columns:

**rows** (n\_train) Number of time points in the training set.

**columns** (D) Regions analyzed (Chatham-Kent, Lambton, Wellington).

**Source**

ClimateData.ca (processed)

---

z_test	<i>Standardized Covariates (Test)</i>
--------	---------------------------------------

---

**Description**

Standardized climate covariates used to forecast with the BSTS models (test).

**Usage**

z\_test

**Format**

A numeric array with dimensions  $n\_test \times D \times M$ :

$n\_test$  Number of test time points.

$D$  Regions (Chatham-Kent, Lambton, Wellington).

$M$  Number of covariates (cddcold, tr18, txgt27, tnmean, txgt29).

**Source**

ClimateData.ca (processed)

---

z_train	<i>Standardized Covariates (Training)</i>
---------	---

---

**Description**

Standardized climate covariates used to fit the BSTS models (training).

**Usage**

z\_train

**Format**

A numeric array with dimensions  $n\_train \times D \times M$ :

$n\_train$  Number of training time points.

$D$  Regions (Chatham-Kent, Lambton, Wellington).

$M$  Number of covariates (cddcold, tr18, txgt27, tnmean, txgt29).

**Source**

ClimateData.ca (processed)

# Index

## \* datasets

- copula\_list, 3
- data, 3
- dt, 4
- init\_params\_full, 9
- medoid\_names, 11
- n\_test, 12
- n\_train, 12
- time, 16
- time\_test, 16
- time\_train, 17
- u, 17
- y\_test, 18
- y\_train, 18
- z\_test, 19
- z\_train, 19

clayton.theta, 2

copula\_list, 3

data, 3

dt, 4

dynamic.rho, 5

dynamic.theta.clayton, 6

dynamic.theta.frank, 6

dynamic.theta.gumbel, 7

dynamic.theta.joe, 7

fit\_bsts, 8

frank.theta, 8

GH.theta, 9

init\_params\_full, 9

joe.theta, 10

log\_likelihood\_noGEV\_3d, 10

medoid\_names, 11

n\_test, 12

n\_train, 12

plot\_forecast, 12

plot\_forecast\_compare, 13

simul\_fun\_noGEV\_3d, 14

time, 16

time\_test, 16

time\_train, 17

u, 17

y\_test, 18

y\_train, 18

z\_test, 19

z\_train, 19