Package ‘trending’

April 3, 2023

**Title**  Model Temporal Trends

**Version**  0.1.0

**Description**  Provides a coherent interface to multiple modelling tools for fitting trends along with a standardised approach for generating confidence and prediction intervals.

**URL**  https://github.com/reconverse/trending

**BugReports**  https://github.com/reconverse/trending/issues

**License**  MIT + file LICENSE

**Encoding**  UTF-8

**RoxygenNote**  7.2.3

**Imports**  ciTools, MASS, stats, tibble, vctrs, pillar

**Suggests**  knitr, brms, BH, ReppEigen, rmarkdown, dplyr, outbreaks, testthat (>= 3.0.0), ggplot2, patchwork, covr

**VignetteBuilder**  knitr

**Config/testthat/edition**  3

**NeedsCompilation**  no

**Author**  Tim Taylor [aut] (<https://orcid.org/0000-0002-8587-7113>), Dirk Schumacher [ctb], Thibaut Jombart [ctb, cre]

**Maintainer**  Thibaut Jombart <thibautjombart@gmail.com>

**Repository**  CRAN

**Date/Publication**  2023-04-03 19:00:02 UTC

**R topics documented:**

accessors ................................................................. 2
fit ................................................................. 6
fit.list .......................................................... 7
fit.trending_model .................................................. 8
predict.list ...................................................... 9
accessors

| predict.trending_fit          | 11 |
| predict.trending_fit_tbl      | 12 |
| predict.trending_model        | 14 |
| trending_model               | 16 |

Index

<table>
<thead>
<tr>
<th>accessor</th>
<th>Description</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>get_result(x, ...)</td>
<td>Description for accessing model information.</td>
<td>get_result(x, ...)</td>
</tr>
<tr>
<td>get_warnings(x, ...)</td>
<td>Description for accessing model information.</td>
<td>get_warnings(x, ...)</td>
</tr>
</tbody>
</table>

Usage

```r
get_result(x, ...)

## Default S3 method:
get_result(x, ...)

## S3 method for class 'trending_fit'
get_result(x, ...)

## S3 method for class 'trending_predict'
get_result(x, ...)

## S3 method for class 'trending_fit_tbl'
get_result(x, ...)

## S3 method for class 'trending_predict_tbl'
get_result(x, ...)

get_warnings(x, ...)

## Default S3 method:
get_warnings(x, ...)

## S3 method for class 'trending_fit'
get_warnings(x, ...)

## S3 method for class 'trending_predict'
get_warnings(x, ...)

## S3 method for class 'trending_fit_tbl'
get_warnings(x, ...)

## S3 method for class 'trending_predict_tbl'
```
get_warnings(x, ...)

get_errors(x, ...)

## Default S3 method:
get_errors(x, ...)

## S3 method for class 'trending_fit'
get_errors(x, ...)

## S3 method for class 'trending_predict'
get_errors(x, ...)

## S3 method for class 'trending_fit_tbl'
get_errors(x, ...)

## S3 method for class 'trending_predict_tbl'
get_errors(x, ...)

get_fitted_model(x, ...)

## Default S3 method:
get_fitted_model(x, ...)

## S3 method for class 'trending_fit'
get_fitted_model(x, ...)

## S3 method for class 'trending_fit_tbl'
get_fitted_model(x, ...)

get_fitted_data(x, ...)

## Default S3 method:
get_fitted_data(x, ...)

## S3 method for class 'trending_fit'
get_fitted_data(x, ...)

## S3 method for class 'trending_fit_tbl'
get_fitted_data(x, ...)

get_formula(x, ...)

## Default S3 method:
get_formula(x, ...)

## S3 method for class 'trending_model'
get_formula(x, ...)
## S3 method for class 'trending_fit'
get_formula(x, ...)

## S3 method for class 'trending_fit.tbl'
get_formula(x, ...)

get_response(x, ...)

## Default S3 method:
get_response(x, ...)

## S3 method for class 'trending_model'
get_response(x, ...)

## S3 method for class 'trending_fit'
get_response(x, ...)

## S3 method for class 'trending_fit.tbl'
get_response(x, ...)

## S3 method for class 'trending_prediction'
get_response(x, ...)

## S3 method for class 'trending_predict'
get_response(x, ...)

## S3 method for class 'trending_predict.tbl'
get_response(x, ...)

get_predictors(x, ...)

## Default S3 method:
get_predictors(x, ...)

## S3 method for class 'trending_model'
get_predictors(x, ...)

## S3 method for class 'trending_fit'
get_predictors(x, ...)

## S3 method for class 'trending_fit.tbl'
get_predictors(x, ...)

## S3 method for class 'trending_prediction'
get_predictors(x, ...)

## S3 method for class 'trending_predict'
get_predictors(x, ...)

## S3 method for class 'trending_predict_tbl'
get_predictors(x, ...)

generate(x, ...)

## Default S3 method:
generate(x, ...)

## S3 method for class 'trending_prediction'
generate(x, ...)

## S3 method for class 'trending_predict'
generate(x, ...)

## S3 method for class 'trending_predict_tbl'
generate(x, ...)

Arguments

x An R object.

... Not currently used by any methods.

Details

Methods are provided for trending_model, trending_fit, trending_fit_tbl, trending_predict, and trending_predict_tbl objects.

Value

- get_result(): the captured output.
- get_warnings(): the captured warnings.
- get_errors(): the captured warnings.
- get_fitted_data: The underlying data used to fit the model.
- get_fitted_model(): The underlying fitted model.
- get_formula(): the formula used to model temporal trends.
- get_response(): the response variable of the underlying model.
- get_predictors(): the predictor variable(s) of the underlying model.

Examples

x = rnorm(100, mean = 0)
y = rpois(n = 100, lambda = exp(1.5 + 0.5*x))
dat <- data.frame(x = x, y = y)
poisson_model <- glm_model(y ~ x, family = "poisson")
fitted_poisson <- fit(poisson_model, dat)
get_fitted_model(fitted_poisson)
get_formula(fitted_poisson)
get_response(fitted_poisson)
get_predictors(fitted_poisson)

---

**fit**

*Fit generic*

**Description**

`fit()` is a generic to fit a specified model.

**Usage**

```
fit(x, data, ...)
```

```
## Default S3 method:
fit(x, data, ...)
```

**Arguments**

- **x**: An R object
- **data**: A data frame containing the data to fit.
- **...**: Arguments passed to underlying methods.

**Value**

The value returned depends on the class of the input argument.

**Author(s)**

Tim Taylor

**See Also**

`fit.trending_model()` and `fit.list()`

**Examples**

```
x = rnorm(100, mean = 0)
y = rpois(n = 100, lambda = exp(1.5 + 0.5*x))
dat <- data.frame(x = x, y = y)

poisson_model <- glm_model(y ~ x, family = poisson)
negbin_model <- glm_nb_model(y ~ x)
```
fit.list

Fit method list object

Description

Fits a list of trending_model objects to the given input data.

Usage

## S3 method for class 'list'
fit(x, data, ...)

Arguments

x  
A list of trending_model objects

data  
A data frame containing the data to fit.

...  
Not currently used.

Value

A trending_fit_tbl object which is a tibble subclass with one row for each model and entries:

• model_name (optional): If the input is a named list then the name is extracted.
• result: the resulting fit from calling the underlying model directly, i.e.
  – lm_model: a fitted model object of class \texttt{lm}
  – glm_model: a fitted model object of class \texttt{glm}
  – glm_nb_model: a fitted model object of class \texttt{negbin}
  – brm_model: An object of class \texttt{brmsfit}
    NULL if fitting fails.
• warnings: any warnings generated during fitting
• errors: any errors generated during fitting

Author(s)

Tim Taylor

See Also

fit.trending_model()
Examples

```r
x <- rnorm(100, mean = 0)
y <- rpois(n = 100, lambda = exp(1.5 + 0.5*x))
dat <- data.frame(x = x, y = y)
poisson_model <- glm_model(y ~ x, family = poisson)
negbin_model <- glm_nb_model(y ~ x)

fit(list(poisson_model, negbin_model), dat)
fit(list(pm = poisson_model, nm = negbin_model), dat)
```

---

**fit.trending_model**  
*Fit method for trending_model objects*

**Description**

Fits the specified model to the input data

**Usage**

```r
## S3 method for class 'trending_model'
fit(x, data, as_tibble = TRUE, ...)
```

**Arguments**

- `x`: An R object
- `data`: A data frame containing the data to fit.
- `as_tibble`: Should the output be converted to a tibble subclass.
- `...`: Not currently used.

**Value**

If `as_tibble = FALSE`, then a `trending_fit` object is returned. This is a list subclass with entries:

- `result`: the resulting fit from calling the underlying model directly, i.e.
  - `lm_model`: a fitted model object of class `lm`
  - `glm_model`: a fitted model object of class `glm`
  - `glm_nb_model`: a fitted model object of class `negbin`
  - `brm_model`: An object of class `brmsfit`
  NULL if fitting fails.
- `warnings`: any warnings generated during fitting
- `errors`: any errors generated during fitting

If `as_tibble = TRUE`, a `trending_fit_tbl` object which is a tibble subclass with one row for each model and columns `result`, `warnings` and `errors` with contents as above.
**predict.list**

Author(s)

Tim Taylor

See Also

`fit.list()`

Examples

```r
x = rnorm(100, mean = 0)
y = rpois(n = 100, lambda = exp(1.5 + 0.5*x))
dat <- data.frame(x = x, y = y)

poisson_model <- glm_model(y ~ x, family = poisson)
negbin_model <- glm_nb_model(y ~ x)

fit(poisson_model, dat)
fit(negbin_model, dat)
fit(list(poisson_model, negbin_model), dat)
fit(list(pm = poisson_model, nm = negbin_model), dat)
```

---

**predict.list**

*Predict method for trending_model objects*

Description

Adds estimated values and associated confidence and/or prediction intervals to data based on trending_model fit.

Usage

```r
## S3 method for class 'list'
predict(
  object,
  data,
  name = "estimate",
  alpha = 0.05,
  add_ci = TRUE,
  ci_names = c("lower_ci", "upper_ci"),
  add_pi = TRUE,
  pi_names = c("lower_pi", "upper_pi"),
  simulate_pi = FALSE,
  sims = 2000,
  uncertain = TRUE,
  ...
)
```
Arguments

object  A list of `trending_model` objects.
data  A `data.frame` containing data to which the model is to be fit and estimates derived.
name  Character vector of length one giving the name to use for the calculated estimate.
alpha  The alpha threshold to be used for prediction intervals, defaulting to 0.05, i.e. 95% prediction intervals are derived.
add_ci  Should a confidence interval be added to the output. Default TRUE.
ci_names  Names to use for the resulting confidence intervals.
add_pi  Should a prediction interval be added to the output. Default TRUE.
pi_names  Names to use for the resulting prediction intervals.
simulate_pi  Should the prediction intervals for glm models be simulated. If TRUE, default, `predict()` uses the `ciTools::add_pi()` function to generate the intervals.
sims  The number of simulations to run when simulating prediction intervals for a glm model.
uncertain  Only used for glm models and when `simulate_pi = FALSE`. Default TRUE. If FALSE uncertainty in the fitted parameters is ignored when generating the parametric prediction intervals.
...
Not currently used.

Value

A `trending_predict_tbl` object which is a `tibble` subclass with one row per model and columns:

- result: the input data frame with additional estimates and, optionally, confidence and or prediction intervals. NULL if the associated `predict` method fails.
- warnings: any warnings generated during prediction.
- errors: any errors generated during prediction.

Author(s)

Tim Taylor

See Also

`predict.trending_model()`, `predict.trending_fit()`, `predict.trending_fit_tbl()`.

Examples

```r
x = rnorm(100, mean = 0)
y = rpois(n = 100, lambda = exp(1.5 + 0.5*x))
dat <- data.frame(x = x, y = y)
poisson_model <- glm_model(y ~ x , family = "poisson")
negbin_model <- glm_nb_model(y ~ x)
predict(list(poisson_model, negbin_model), dat)
predict(list(pm = poisson_model, nm = negbin_model), dat)
```
predict.trending_fit

**predict.trending_fit**  
_Predict method for trending_fit object_

---

**Description**

Adds estimated values and associated confidence and/or prediction intervals to trending_fit objects.

**Usage**

```r
## S3 method for class 'trending_fit'
predict(
  object,
  new_data,
  name = "estimate",
  alpha = 0.05,
  add_ci = TRUE,
  ci_names = c("lower_ci", "upper_ci"),
  add_pi = TRUE,
  pi_names = c("lower_pi", "upper_pi"),
  simulate_pi = FALSE,
  sims = 2000,
  uncertain = TRUE,
  as_tibble = TRUE,
  ...
)
```

**Arguments**

- **object**  
  A `trending_fit` object.

- **new_data**  
  A `data.frame` containing data for which estimates are to be derived. If missing, the model frame from the fit data will be used.

- **name**  
  Character vector of length one giving the name to use for the calculated estimate.

- **alpha**  
  The alpha threshold to be used for prediction intervals, defaulting to 0.05, i.e. 95% prediction intervals are derived.

- **add_ci**  
  Should a confidence interval be added to the output. Default TRUE.

- **ci_names**  
  Names to use for the resulting confidence intervals.

- **add_pi**  
  Should a prediction interval be added to the output. Default TRUE.

- **pi_names**  
  Names to use for the resulting prediction intervals.

- **simulate_pi**  
  Should the prediction intervals for glm models be simulated. If TRUE, default, `predict()` uses the `ciTools::add_pi()` function to generate the intervals.

- **sims**  
  The number of simulations to run when simulating prediction intervals for a glm model.

- **uncertain**  
  Only used for glm models and when `simulate_pi = FALSE`. Default TRUE. If FALSE uncertainty in the fitted parameters is ignored when generating the parametric prediction intervals.
predict.trending_fit_tbl

The function `predict.trending_fit_tbl` is a predict method for `trending_fit_tbl` objects, which adds estimated values and associated confidence and/or prediction intervals to the `trending_fit_tbl` object.

### Description

This function is used to predict values from a `trending_fit_tbl` object. It takes into account the initial data frame and additional information to provide estimated values along with confidence and prediction intervals.

### Value

If `as_tibble = FALSE`, the output is a `trending_predict` object, which is a list subclass with entries:

- **result**: the input data frame with additional estimates and, optionally, confidence and or prediction intervals. NULL if the associated `predict` method fails.
- **warnings**: any warnings generated during prediction.
- **errors**: any errors generated during prediction.

If `as_tibble = TRUE`, the output is a `trending_predict_tbl` object which is a `tibble` subclass with one row per model and columns 'result', 'warnings' and 'errors' with contents as above.

### Author(s)

Tim Taylor

### See Also

`predict.trending_fit_tbl()` and `predict.trending_model()`

### Examples

```r
x = rnorm(100, mean = 0)
y = rpois(n = 100, lambda = exp(1.5 + 0.5*x))
dat <- data.frame(x = x, y = y)
poisson_model <- glm_model(y ~ x, family = "poisson")
fitted_poisson <- fit(poisson_model, dat)
predict(fitted_poisson)
predict(fitted_poisson, as_tibble = FALSE)
```
Usage

```r
## S3 method for class 'trending_fit_tbl'
predict(
  object,
  new_data,
  name = "estimate",
  alpha = 0.05,
  add_ci = TRUE,
  ci_names = c("lower_ci", "upper_ci"),
  add_pi = TRUE,
  pi_names = c("lower_pi", "upper_pi"),
  simulate_pi = FALSE,
  sims = 2000,
  uncertain = TRUE,
  ...
)
```

Arguments

- **object**: A `trending_fit_tbl` object.
- **new_data**: A `data.frame` containing data for which estimates are to be derived. If missing, the model frame from the fit data will be used.
- **name**: Character vector of length one giving the name to use for the calculated estimate.
- **alpha**: The alpha threshold to be used for prediction intervals, defaulting to 0.05, i.e. 95% prediction intervals are derived.
- **add_ci**: Should a confidence interval be added to the output. Default TRUE.
- **ci_names**: Names to use for the resulting confidence intervals.
- **add_pi**: Should a prediction interval be added to the output. Default TRUE.
- **pi_names**: Names to use for the resulting prediction intervals.
- **simulate_pi**: Should the prediction intervals for glm models be simulated. If TRUE, default, `predict()` uses the `ciTools::add_pi()` function to generate the intervals.
- **sims**: The number of simulations to run when simulating prediction intervals for a glm model.
- **uncertain**: Only used for glm models and when `simulate_pi = FALSE`. Default TRUE. If FALSE uncertainty in the fitted parameters is ignored when generating the parametric prediction intervals.
- **...**: Not currently used.

Value

A `trending_predict_tbl` object which is a `tibble` subclass with one row per model and columns 'result', 'warnings' and 'errors' with contents as above:

- result: the input data frame with additional estimates and, optionally, confidence and or prediction intervals. NULL if the associated `predict` method fails.
- warnings: any warnings generated during prediction.
- errors: any errors generated during prediction.
Author(s)

Tim Taylor

See Also

predict.trending_fit(), predict.trending_fit_tbl() and predict.trending_model()

Examples

```r
x = rnorm(100, mean = 0)
y = rpois(n = 100, lambda = exp(1.5 + 0.5*x))
dat <- data.frame(x = x, y = y)
poisson_model <- glm_model(y ~ x, family = "poisson")
negbin_model <- glm_nb_model(y ~ x)
fitted_tbl <- fit(list(poisson_model, negbin_model), dat)

predict(fitted_tbl)
```

predict.trending_model

*Predict method for trending_model objects*

Description

Adds estimated values and associated confidence and/or prediction intervals to data based on trending_model fit.

Usage

```r
## S3 method for class 'trending_model'
predict(
  object,
  data,
  name = "estimate",
  alpha = 0.05,
  add_ci = TRUE,
  ci_names = c("lower_ci", "upper_ci"),
  add_pi = TRUE,
  pi_names = c("lower_pi", "upper_pi"),
  simulate_pi = FALSE,
  sims = 2000,
  uncertain = TRUE,
  as_tibble = TRUE,
  ...
)
```
predict.trending_model

Arguments

- **object**: A `trending_model` object.
- **data**: A `data.frame` containing data to which the model is to be fit and estimates derived.
- **name**: Character vector of length one giving the name to use for the calculated estimate.
- **alpha**: The alpha threshold to be used for prediction intervals, defaulting to 0.05, i.e. 95% prediction intervals are derived.
- **add_ci**: Should a confidence interval be added to the output. Default TRUE.
- **ci_names**: Names to use for the resulting confidence intervals.
- **add_pi**: Should a prediction interval be added to the output. Default TRUE.
- **pi_names**: Names to use for the resulting prediction intervals.
- **simulate_pi**: Should the prediction intervals for glm models be simulated. If TRUE, default, `predict()` uses the `ciTools::add_pi()` function to generate the intervals.
- **sims**: The number of simulations to run when simulating prediction intervals for a glm model.
- **uncertain**: Only used for glm models and when `simulate_pi = FALSE`. Default TRUE. If FALSE uncertainty in the fitted parameters is ignored when generating the parametric prediction intervals.
- **as_tibble**: Should the output be converted to a tibble subclass.
- **...**: Not currently used.

Value

If `as_tibble = FALSE`, a `trending_predict` object, which is a list subclass, with entries:

- **result**: the input data frame with additional estimates and, optionally, confidence and or prediction intervals. NULL if the associated predict method fails.
- **warnings**: any warnings generated during prediction.
- **errors**: any errors generated during prediction.

If `as_tibble = TRUE`, a `trending_predict_tbl` object which is a tibble subclass with one row per model and columns 'result', 'warnings' and 'errors' with contents as above.

Author(s)

Tim Taylor

See Also

- `predict.trending_fit()` and `predict.trending_fit_tbl()`
Examples

```r
tax <- rnorm(100, mean = 0)
y <- rpois(n = 100, lambda = exp(1.5 + 0.5*x))
dat <- data.frame(x = x, y = y)
poisson_model <- glm_model(y ~ x, family = "poisson")
predict(poisson_model, dat)
predict(poisson_model, dat, as_tibble = FALSE)
```

### Description

These functions wrap various modelling tools to ensure a consistent input for *trending* functions. They work by capturing the underlying model call and decoupling it from the data specification. This makes it easy to use the same underlying model specification and fitting procedure across different data sets. See details for available model interfaces.

### Usage

- `lm_model(formula, ...)`
- `glm_model(formula, family = gaussian, ...)`
- `glm_nb_model(formula, ...)`
- `brm_model(formula, ...)`

### Arguments

- `formula` The formula of the model, with the response variable on the left of a tilde symbol, and predictors on the right hand-side; variable names used in the formula will need to be matched by columns in the data input to other functions.
- `...` Further arguments passed to the underlying models with the exception of `data`.
- `family` Link function to be used for the glm model.

### Details

The following interfaces are available:

- `lm_model`: interface for linear models implemented in `stats::lm()`.
- `glm_model`: interface for generalised linear models (GLMs) implemented in `stats::glm()`.
- `glm_nb_model`: interface for negative binomial generalised linear models implemented in `MASS::glm.nb()`.
- `brm_model`: interface for Bayesian regression models implemented in `brms::brm()`.
These interfaces will accept the same inputs as the underlying model functions but do not require, nor will they accept, a data argument. Fitting is handled instead by the `fit()` generic and associated methods.

Value

A `trending_model` object.

Author(s)

Tim Taylor

Examples

```r
x = rnorm(100, mean = 0)
y = rpois(n = 100, lambda = exp(1.5 + 0.5*x))

poisson_model <- glm_model(y ~ x, family = "poisson")
negbin_model <- glm_nb_model(y ~ x)
```
Index

accessors, 2
brm_model(trending_model), 16
brms::brm(), 16
brmsfit, 7, 8
ciTools::add_pi(), 10, 11, 13, 15
fit, 6
fit(), 17
fit.list(), 7
fit.list(), 6, 9
fit.trending_model, 8
fit.trending_model(), 6, 7
get_errors (accessors), 2
get_estimate (accessors), 2
get_fitted_data (accessors), 2
get_fitted_model (accessors), 2
get_formula (accessors), 2
get_predictors (accessors), 2
get_response (accessors), 2
get_result (accessors), 2
get_warnings (accessors), 2
glm, 7, 8
glm_model(trending_model), 16
glm_nb_model(trending_model), 16
lm, 7, 8
lm_model(trending_model), 16
MASS::glm.nb(), 16
negbin, 7, 8
predict.list, 9
predict.trending_fit, 11
predict.trending_fit(), 10, 14, 15
predict.trending_fit_tbl, 12
predict.trending_fit_tbl(), 10, 12, 14, 15
predict.trending_model, 14
predict.trending_model(), 10, 12, 14
stats::glm(), 16
stats::lm(), 16
tibble, 7, 8, 10, 12, 13, 15
trending_fit, 11
trending_fit_tbl, 13
trending_model, 10, 15, 16
trending_models (trending_model), 16