Package ‘StockDistFit’

January 20, 2025

Title Fit Stock Price Distributions

Version 1.0.0

Description The 'StockDistFit' package provides functions for fitting probability distributions to stock price data. The package uses maximum likelihood estimation to find the best-fitting distribution for a given stock. It also offers a function to fit several distributions to one or more assets and compare the distribution with the Akaike Information Criterion (AIC) and then pick the best distribution. References are as follows: Siew et al. (2008) <https://www.jstage.jst.go.jp/article/jappstat/37/1/37_1_1/_pdf/-char/ja> and Benth et al. (2008) <https://books.google.co.ke/books?hl=en&lr=&id=MHNpDQAAQBAJ&oi=fnd&pg=PR7&dq=Stochastic+modeling+of+commodity+prices+using+the+Variance+Gamma+(VG)+model.+&ots=YNIL2QmEYg&sig=XZtGU0lp4oqXHVyPZ-O8x5i7N3w&redir_esc=y#v=onepage&q&f=false>.

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.2.3

Imports dplyr, fGarch, fBasics, fitdistrplus, xts, stats, magrittr, zoo, quantmod, utils, ghyp

Suggests knitr, rmarkdown

VignetteBuilder knitr

Depends R (>= 2.10)

LazyData true

NeedsCompilation no

Author Brian Njuguna [aut, cre] (<https://orcid.org/0009-0002-2119-904X>), Stanely Sayianka [ctb]

Maintainer Brian Njuguna <`briannjuguna133@gmail.com`>

Repository CRAN

Date/Publication 2023-05-09 09:00:08 UTC

Contents

AAPL *Apple Inc. stock prices dataset*

Description

This dataset contains the daily stock prices of Apple Inc. (AAPL) from January 2, 2013 to April 30, 2023. The data includes the open, high, low, and close prices, as well as the volume and adjusted close price. ~~

Usage

```
data("AAPL")
```

Format

A data frame with 2599 observations on the following 7 variables.

`Date` a character vector

`Open` a numeric vector

`High` a numeric vector

`Low` a numeric vector

`Close` a numeric vector

`Volume` a numeric vector

`Adjusted` a numeric vector

References

Data source: Yahoo Finance

Examples

```
data(AAPL)
str(AAPL) ; plot(AAPL)
```

`AMZN` *Amazon.com Inc. Stock Prices Dataset*

Description

This dataset contains the daily stock prices of Amazon.com Inc. (AMZN) from January 2, 2013 to April 30, 2023. The data includes the open, high, low, and close prices, as well as the volume and adjusted close price. ~~

Usage

```
data("AMZN")
```

Format

A data frame with 2599 observations on the following 7 variables.

`Date` a character vector

`Open` a numeric vector

`High` a numeric vector

`Low` a numeric vector

`Close` a numeric vector

`Volume` a numeric vector

`Adjusted` a numeric vector

References

Data source: Yahoo Finance

Examples

```
data(AMZN)
str(AMZN) ; plot(AMZN)
```

annual_return *Compute Annual Returns of a Vector.*

Description

This function takes a vector of asset returns and computes annual returns.

Usage

```
annual_return(vec)
```

Arguments

vec	a numeric vector of asset returns as an xts object with dates as rownames.

Value

A numeric vector of annual returns.

See Also

weekly_return, monthly_return

Examples

```
# Compute annual returns of an asset vector
require(xts)
asset_returns_xts <- xts(c(29.2, 30.0, 36.2, 30.4, 38.5, -35.6, 34.5),
order.by = as.Date(c("2017-05-07", "2018-05-07", "2019-05-07",
"2020-05-07", "2021-05-07", "2022-05-07",
"2023-05-07")))
annual_return(asset_returns_xts)
```

asset_loader *Load Asset Data.*

Description

This function reads in asset data stored in .csv format and returns a time-series object of the asset data.

Usage

```
asset_loader(data_path, assets, price_col)
```

Arguments

data_path The path to the directory containing the .csv files.

assets A vector of asset names to be loaded.

price_col The name of the price column to be selected (e.g. Open, Close, Low, High).

Value

An xts object with asset data.

Note

The Date column in the files should be of the format "%m/%d/%y", that is 01/14/13 with 01 implying the month, 14 the date and 13 the year

The data to be loaded must be in .csv type and also must have the Date, Open, Low, High and Close Prices of the assest or assests to be loaded.

Examples

```
asset_loader(system.file("extdata", package = "StockDistFit"), c("AAPL", "TSLA"), "Close")
```

best_dist *Find the best distribution based on AIC values*

Description

This function takes in a data frame of AIC values for different distributions and a vector of distribution names, and returns a data frame with the best distribution for each row based on the minimum AIC value. #' You can also write the distribution as "norm" or "cauchy" provided they follow the order in the data frame.

Usage

```
best_dist(aic_df, dist_names)
```

Arguments

aic_df A data frame containing AIC values for different distributions

dist_names A vector of distribution names corresponding to the AIC values

Value

A data frame with the best distribution for each row based on the minimum AIC value

Note

This function takes the data frame obtained from `fit_multiple_dist` function

Examples

```
data <- asset_loader(system.file("extdata", package = "StockDistFit"), c("AAPL", "TSLA"), "Close")
df = fit_multiple_dist(c("norm_fit", "cauchy_fit"), data)
best_dist(df, c("norm_fit", "cauchy_fit"))
```

`cauchy_fit` *Fit Cauchy Distribution to a vector of returns/stock prices.*

Description

This function fits the Cauchy distribution to a given data vector using the fitdist function from the fitdistrplus package. It returns the estimated parameters along with the AIC and BIC values for the fitted distribution.

Usage

```
cauchy_fit(vec)
```

Arguments

`vec` a numeric vector containing the data to be fitted.

Value

a list containing the following elements:

par a numeric vector of length 2 containing the estimated values for the parameters of the fitted distribution: lambda (location) and alpha (scale).

aic the Akaike information criterion (AIC) value for the fitted distribution.

bic the Bayesian information criterion (BIC) value for the fitted distribution.

See Also

`norm_fit`, `t_fit`, `ghd_fit`, `hd_fit`, `sym.ghd_fit`, `sym.hd_fit`, `vg_fit`, `sym.vg_fit`, `nig_fit`, `ged_fit`, `skew.t_fit`, `skew.normal_fit`, `skew.ged_fit`

Examples

```
stock_prices <- c(10, 11, 12, 13, 14, 17, 18)
returns <- diff(log(stock_prices))
cauchy_fit(returns)
```

data.cumret *Compute Cumulative Returns of a Vector.*

Description

This function takes a vector of asset returns and computes the cumulative wealth generated over time, assuming that the initial wealth was `initial_eq`.

Usage

```
data.cumret(df_ret, initial_eq)
```

Arguments

`df_ret`	an xts object of asset returns, with dates as rownames.
`initial_eq`	a numeric value representing the initial wealth.

Value

An xts object of wealth generated over time.

See Also

`weekly_return`, `monthly_return`, `annual_return`

Examples

```
# Compute cumulative returns of an asset vector
library(quantmod)
asset_returns_xts <- xts(c(29.2, 30.0, 36.2, 30.4, 38.5, -35.6, 34.5),
                         order.by = as.Date(c("2023-05-01", "2023-05-02", "2023-05-03",
                                              "2023-05-04", "2023-05-05", "2023-05-06",
                                              "2023-05-07")))
data.cumret(asset_returns_xts, initial_eq = 100)
```

fit_multiple_dist *Fits Multiple Probability Distributions to several assets/stock prices.*

Description

This function fits multiple probability distributions to a dataframe and calculates the Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC) for each distribution and then returns a data frame of the AIC values for each asset where the column names are the names of the fitted distributions.

Usage

```
fit_multiple_dist(dist_names, dataframe)
```

Arguments

`dist_names`	a character vector of distribution names to be fitted.
`dataframe`	a dataframe containing the data to be fitted.

Details

Note that the available distributions are

norm_fit - Normal distribution

t_fit - Student's t-distribution

cauchy_fit - Cauchy distribution

ghd_fit - Generalized hyperbolic distribution

hd_fit - Hyperbolic distribution

sym.ghd_fit - Symmetric generalized hyperbolic distribution

sym.hd_fit - Symmetric hyperbolic distribution

vg_fit - Variance-gamma distribution

sym.vg_fit - Symmetric variance-gamma distribution

nig_fit - Normal-inverse Gaussian distribution

ged_fit - Generalized error distribution

skew.t_fit - Skew Student's t-distribution

skew.normal_fit - Skew normal distribution

skew.ged_fit - Skew generalized error distribution

Also note that the distribution to be fitted from the above list must include the '_fit'. The function can also fit one distribution to one asset.

Value

A list of distributions and their corresponding AIC and BIC values.

See Also

asset_loader

Examples

```
data <- asset_loader(system.file("extdata", package = "StockDistFit"), c("AAPL", "TSLA"), "Close")
fit_multiple_dist(c("norm_fit", "cauchy_fit"), data)
```

`ged_fit` *Fit Generalized Error Distribution to a vector of returns/stock prices.*

Description

This function fits the Generalized Error Distribution (GED) to a given data vector using the `ged_fit` function from the `fGarch` package. It returns the estimated parameters along with the AIC and BIC values for the fitted distribution.

Usage

```
ged_fit(vec)
```

Arguments

`vec`	A numeric vector of data.

Value

A list with the following elements:

- **params** A numeric vector of length 3 containing the fitted GED parameters: shape, scale, and location.
- **aic** The Akaike Information Criterion (AIC) for the fitted model.
- **bic** The Bayesian Information Criterion (BIC) for the fitted model.

See Also

`norm_fit`, `t_fit`, `cauchy_fit`, `ghd_fit`, `hd_fit`, `sym.ghd_fit`, `sym.hd_fit`, `vg_fit`, `nig_fit`, `sym.vg_fit`, `skew.t_fit`, `skew.normal_fit`, `skew.ged_fit`

Examples

```
stock_prices <- c(10, 11, 12, 13, 14, 17, 18)
returns <- diff(log(stock_prices))
ged_fit(returns)
```

ghd_fit *Fit Generalized Hyperbolic Distribution to a vector of returns/stock prices.*

Description

This function fits the Generalized Hyperbolic (GH) distribution to a given data vector using the `fit.ghypuv` function from the ghyp package. It returns the estimated parameters along with the AIC and BIC values for the fitted distribution.

Usage

```
ghd_fit(vec)
```

Arguments

`vec`	a numeric vector containing the data to be fitted.

Value

a list containing the following elements:

par a numeric vector of length 5 containing the estimated values for the parameters of the fitted distribution: lambda (location), alpha (scale), mu (degrees of freedom), sigma (standard deviation), and gamma (skewness).

aic the Akaike information criterion (AIC) value for the fitted distribution.

bic the Bayesian information criterion (BIC) value for the fitted distribution.

See Also

`norm_fit`, `t_fit`, `cauchy_fit`, `hd_fit`, `sym.ghd_fit`, `sym.hd_fit`, `vg_fit`, `sym.vg_fit`, `nig_fit`, `ged_fit`, `skew.t_fit`, `skew.normal_fit`, `skew.ged_fit`

Examples

```
stock_prices <- c(10, 11, 12, 13, 14, 16, 24)
returns <- diff(log(stock_prices))
ghd_fit(returns)
```

GOOG *Alphabet Inc Inc. Stock Prices Dataset*

Description

This dataset contains the daily stock prices of Alphabet Inc. (GOOG) from January 2, 2013 to April 30, 2023. The data includes the open, high, low, and close prices, as well as the volume and adjusted close price. ~~

Usage

```
data("GOOG")
```

Format

A data frame with 2599 observations on the following 7 variables.

`Open` a numeric vector

`High` a numeric vector

`Low` a numeric vector

`Close` a numeric vector

`Volume` a numeric vector

`Adjusted` a numeric vector

`Date` a character vector

References

Data source: Yahoo Finance

Examples

```
data(GOOG)
str(GOOG) ; plot(GOOG)
```

hd_fit *Fit Hyperbolic distribution to return/stock prices.*

Description

This function fits the Hyperbolic distribution to a given data vector using the `fit.hypuv` function from the ghyp package. It returns the estimated parameters along with the AIC and BIC values for the fitted distribution.

Usage

```
hd_fit(vec)
```

Arguments

`vec`	a numeric vector containing the data to be fitted.

Value

a list containing the following elements:

par a numeric vector of length 4 containing the estimated values for the parameters of the fitted distribution: alpha (scale), mu (location), sigma (standard deviation), and gamma (skewness).

aic the Akaike information criterion (AIC) value for the fitted distribution.

bic the Bayesian information criterion (BIC) value for the fitted distribution.

See Also

norm_fit, sym.ghd_fit, ghd_fit, cauchy_fit, t_fit, sym.hd_fit, vg_fit, sym.vg_fit, nig_fit, ged_fit, skew.t_fit, skew.normal_fit, skew.ged_fit

Examples

```
stock_prices <- c(10, 11, 12, 13, 14, 15, 16)
returns <- diff(log(stock_prices))
hd_fit(returns)
```

---

monthly_return *Compute Monthly Returns of a Vector.*

Description

This function takes a numeric vector of asset returns and computes monthly returns.

Usage

```
monthly_return(vec)
```

Arguments

`vec`	a numeric vector of asset returns.

Value

A numeric vector of monthly returns.

Note

The input data must be an xts object with dates as rownames.

See Also

weekly_return, annual_return

Examples

```
# Compute monthly returns of an asset vector
require(xts)
asset_returns_xts <- xts(c(29.2, 30.0, 36.2, 30.4, 38.5, -35.6, 34.5),
                         order.by = as.Date(c("2022-05-02", "2022-06-02", "2022-07-02",
                                              "2022-08-02", "2022-09-02", "2022-10-02",
                                              "2022-11-02")))
monthly_return(asset_returns_xts)
```

---

nig_fit *Fit Normal Inverse Gaussian (NIG) Distribution to a vector of returns/stock prices.*

Description

This function fits the Normal Inverse Gaussian (NIG) Distribution to a given data vector using the `nig_fit` function from the `fBasics` package. It returns the estimated parameters along with the AIC and BIC values for the fitted distribution.

Usage

```
nig_fit(vec)
```

Arguments

vec	A numeric vector of data.

Value

A list with the following elements:

params The estimated parameters of the NIG distribution: location, scale, skewness, and shape.

aic The Akaike Information Criterion (AIC) for the NIG distribution fit.

bic The Bayesian Information Criterion (BIC) for the NIG distribution fit.

See Also

norm_fit, t_fit, cauchy_fit, ghd_fit, hd_fit, sym.ghd_fit, sym.hd_fit, vg_fit, sym.vg_fit ged_fit, skew.t_fit, skew.normal_fit, skew.ged_fit

Examples

```
stock_prices <- c(10, 11, 12, 13, 14, 17, 18)
returns <- diff(log(stock_prices))
nig_fit(returns)
```

---

norm_fit *Fit Normal Distribution to a Vector/stock prices.*

Description

This function takes a numeric vector and fits a normal distribution to it using the fitdist function from the fitdistrplus package. It returns a list with the mean and standard deviation parameters of the fitted normal distribution, as well as the AIC and BIC values of the fitted distribution.

Usage

```
norm_fit(vec)
```

Arguments

`vec`	a numeric vector to be fitted with a normal distribution.

Value

A list with the following components:

par a numeric vector with the estimated mean and standard deviation parameters of the fitted normal distribution.

aic a numeric value representing the Akaike information criterion (AIC) of the fitted distribution.

bic a numeric value representing the Bayesian information criterion (BIC) of the fitted distribution.

See Also

`t_fit`, `cauchy_fit`, `ghd_fit`, `hd_fit`, `sym.ghd_fit`, `sym.hd_fit`, `vg_fit`, `sym.vg_fit`, `nig_fit`, `ged_fit`, `skew.t_fit`, `skew.normal_fit`, `skew.ged_fit`

Examples

```
# Fit a normal distribution to a vector of returns
stock_prices <- c(10, 11, 12, 13, 14, 17, 18)
returns <- diff(log(stock_prices))
norm_fit(returns)
```

skew.ged_fit *Fit Skewed Generalized Error Distribution to a vector of returns/stock prices.*

Description

This function fits the Skewed Generalized Error Distribution to a given data vector using the `skew.ged_fit` function from the `fGarch` package. It returns the estimated parameters along with the AIC and BIC values for the fitted distribution.

Usage

```
skew.ged_fit(vec)
```

Arguments

`vec`	A numeric vector of data.

Value

A list with the following elements:

params A numeric vector of length 4 containing the fitted SGED parameters: shape, scale, location, and skewness.

aic The Akaike Information Criterion (AIC) for the fitted model.

bic The Bayesian Information Criterion (BIC) for the fitted model.

See Also

`norm_fit`, `t_fit`, `cauchy_fit`, `ghd_fit`, `hd_fit`, `sym.ghd_fit`, `sym.hd_fit`, `vg_fit`, `sym.vg_fit`, `nig_fit`, `ged_fit`, `skew.t_fit`, `skew.normal_fit`

Examples

```
stock_prices <- c(10, 11, 12, 13, 14, 17, 18)
returns <- diff(log(stock_prices))
skew.ged_fit(returns)
```

skew.normal_fit *Fit Skew Normal Distribution to a vector of returns/stock prices.*

Description

This function fits the Skew Normal distribution to a given data vector using the snormFit function from the fGarch package. It returns the estimated parameters along with the AIC and BIC values for the fitted distribution.

Usage

```
skew.normal_fit(vec)
```

Arguments

vec	a numeric vector containing the data to be fitted.

Value

a list containing the following elements:

params a numeric vector of length 3 containing the estimated values for the parameters of the fitted distribution: location (mu), scale (sigma), and skewness (alpha).

aic the Akaike information criterion (AIC) value for the fitted distribution.

bic the Bayesian information criterion (BIC) value for the fitted distribution.

See Also

norm_fit, t_fit, cauchy_fit, ghd_fit, hd_fit, sym.ghd_fit, sym.hd_fit, vg_fit, sym.vg_fit, nig_fit, ged_fit, skew.t_fit, skew.ged_fit

Examples

```
stock_prices <- c(10, 11, 12, 13, 14, 17, 18)
returns <- diff(log(stock_prices))
skew.normal_fit(returns)
```

`skew.t_fit` *Fit Skewed Student-t Distribution to a vector of returns/stock prices.*

Description

This function fits the Skewed Student-t Distribution to a given data vector using the `skew.t_fit` function from the `fGarch` package. It returns the estimated parameters along with the AIC and BIC values for the fitted distribution.

Usage

```
skew.t_fit(vec)
```

Arguments

`vec`	A numeric vector of data.

Value

A list with the following elements:

params A numeric vector of length 4 containing the fitted Skewed Student-t parameters: degrees of freedom, skewness, scale, and location.

aic The Akaike Information Criterion (AIC) for the fitted model.

bic The Bayesian Information Criterion (BIC) for the fitted model.

See Also

`norm_fit`, `t_fit`, `cauchy_fit`, `ghd_fit`, `hd_fit`, `sym.ghd_fit`, `sym.hd_fit`, `vg_fit`, `nig_fit`, `sym.vg_fit`, `ged_fit`, `skew.normal_fit`, `skew.ged_fit`

Examples

```
stock_prices <- c(10, 11, 12, 13, 14, 17, 18)
returns <- diff(log(stock_prices))
skew.t_fit(returns)
```

sym.ghd_fit *Fit Symmetric Generalized Hyperbolic Distribution to returns/stock prices.*

Description

This function fits the Symmetric Generalized Hyperbolic (sGH) distribution to a given data vector using the `fit.ghypuv` function from the ghyp package. It returns the estimated parameters along with the AIC and BIC values for the fitted distribution.

Usage

```
sym.ghd_fit(vec)
```

Arguments

`vec`	a numeric vector containing the data to be fitted.

Value

a list containing the following elements:

par a numeric vector of length 5 containing the estimated values for the parameters of the fitted distribution: lambda (location), alpha (scale), mu (degrees of freedom), sigma (standard deviation), and gamma (skewness).

aic the Akaike information criterion (AIC) value for the fitted distribution.

bic the Bayesian information criterion (BIC) value for the fitted distribution.

See Also

`norm_fit`, `t_fit`, `cauchy_fit`, `ghd_fit`, `hd_fit`, `sym.hd_fit`, `vg_fit`, `sym.vg_fit`, `nig_fit`, `ged_fit`, `skew.t_fit`, `skew.normal_fit`, `skew.ged_fit`

Examples

```
stock_prices <- c(10, 11, 12, 13, 14, 16, 15)
returns <- diff(log(stock_prices))
sym.ghd_fit(returns)
```

sym.hd_fit *Fit a Symmetric Hyperbolic Distribution to a vector of return/stock prices.*

Description

This function fits a Symmetric Hyperbolic distribution to a data vector using the `fit.hypuv` function from the ghyp package. It returns the estimated parameters along with the AIC and BIC values for the fitted distribution.

Usage

```
sym.hd_fit(vec)
```

Arguments

`vec`	a numeric vector containing the symmetric data to be fitted.

Value

a list containing the following elements:

par a numeric vector of length 4 containing the estimated values for the parameters of the fitted distribution: alpha (scale), mu (degrees of freedom), sigma (standard deviation), and gamma (skewness).

aic the Akaike information criterion (AIC) value for the fitted distribution.

bic the Bayesian information criterion (BIC) value for the fitted distribution.

See Also

`norm_fit`, `t_fit`, `cauchy_fit`, `ghd_fit`, `hd_fit`, `sym.ghd_fit`, `vg_fit`, `sym.vg_fit`, `nig_fit`, `ged_fit`, `skew.t_fit`, `skew.normal_fit`, `skew.ged_fit`

Examples

```
stock_prices <- c(10, 11, 12, 13, 14, 20, 21)
returns <- diff(log(stock_prices))
sym.hd_fit(returns)
```

sym.vg_fit *Fit Symmetric Variance Gamma Distribution to a vector of returns/stock prices.*

Description

This function fits the Symmetric Variance Gamma (sVG) distribution to a given data vector using the `fit.VGuv` function from the ghyp package. It returns the estimated parameters along with the AIC and BIC values for the fitted distribution.

Usage

```
sym.vg_fit(vec)
```

Arguments

`vec`	a numeric vector containing the data to be fitted.

Value

a list containing the following elements:

par a numeric vector of length 4 containing the estimated values for the parameters of the fitted distribution: lambda (scale), mu (location), sigma (volatility), and gamma (skewness).

aic the Akaike information criterion (AIC) value for the fitted distribution.

bic the Bayesian information criterion (BIC) value for the fitted distribution.

See Also

`norm_fit`, `t_fit`, `cauchy_fit`, `ghd_fit`, `hd_fit`, `sym.ghd_fit`, `sym.hd_fit`, `vg_fit`, `nig_fit`, `ged_fit`, `skew.t_fit`, `skew.normal_fit`, `skew.ged_fit`

Examples

```
stock_prices <- c(10, 11, 12, 13, 14, 17, 18)
returns <- diff(log(stock_prices))
sym.vg_fit(returns)
```

TSLA *Tesla Inc. Stock Prices Dataset*

Description

This dataset contains the daily stock prices of Tesla, Inc. (TSLA) from January 2, 2013 to May 6, 2023. The data includes the open, high, low, and close prices, as well as the volume and adjusted close price. ~~

Usage

```
data("TSLA")
```

Format

A data frame with 2599 observations on the following 7 variables.

`Open` a numeric vector

`High` a numeric vector

`Low` a numeric vector

`Close` a numeric vector

`Volume` a numeric vector

`Adjusted` a numeric vector

`Date` a character vector

References

Data source: Yahoo Finance

Examples

```
data(TSLA)
str(TSLA) ; plot(TSLA)
```

t_fit *Fit Student's t Distribution to a vector of returns/stock prices.*

Description

This function fits the Student's t distribution to a given data vector using the `fit.tuv` function from the ghyp package. It returns the estimated parameters along with the AIC and BIC values for the fitted distribution.

Usage

```
t_fit(vec)
```

Arguments

`vec` a numeric vector containing the data to be fitted.

Value

a list containing the following elements:

par a numeric vector of length 5 containing the estimated values for the parameters of the fitted distribution: lambda (location), alpha (scale), mu (degrees of freedom), sigma (standard deviation), and gamma (skewness).

aic the Akaike information criterion (AIC) value for the fitted distribution.

bic the Bayesian information criterion (BIC) value for the fitted distribution.

See Also

`norm_fit`, `cauchy_fit`, `ghd_fit`, `hd_fit`, `sym.ghd_fit`, `sym.hd_fit`, `vg_fit`, `sym.vg_fit`, `nig_fit`, `ged_fit`, `skew.t_fit`, `skew.normal_fit`, `skew.ged_fit`

Examples

```
stock_prices <- c(10, 11, 12, 13, 14, 17, 18)
returns <- diff(log(stock_prices))
t_fit(returns)
```

vg_fit *Fit Variance Gamma Distribution to a vector of return/stock prices.*

Description

This function fits the Variance Gamma (VG) distribution to a given data vector using the `fit.VGuv` function from the ghyp package. It returns the estimated parameters along with the AIC and BIC values for the fitted distribution.

Usage

```
vg_fit(vec)
```

Arguments

`vec` a numeric vector containing the data to be fitted.

Value

a list containing the following elements:

par a numeric vector of length 4 containing the estimated values for the parameters of the fitted distribution: lambda (location), mu (scale), sigma (shape), and gamma (skewness).

aic the Akaike information criterion (AIC) value for the fitted distribution.

bic the Bayesian information criterion (BIC) value for the fitted distribution.

See Also

`norm_fit`, `t_fit`, `cauchy_fit`, `ghd_fit`, `hd_fit`, `sym.ghd_fit`, `sym.hd_fit`, `sym.vg_fit`, `nig_fit`, `ged_fit`, `skew.t_fit`, `skew.normal_fit`, `skew.ged_fit`

Examples

```
stock_prices <- c(10, 11, 12, 13, 14, 15, 17)
returns <- diff(log(stock_prices))
vg_fit(returns)
```

`weekly_return` *Compute Weekly Returns of a Vector.*

Description

This function takes a numeric vector of asset returns and computes weekly returns.

Usage

```
weekly_return(vec)
```

Arguments

`vec`	a numeric vector of asset returns.

Value

A numeric vector of weekly returns.

Note

The input data must be an xts object with dates as rownames.

See Also

`monthly_return`, `annual_return`

Examples

```
# Compute weekly returns of an asset vector
require(xts)
asset_returns_xts <- xts(c(29.2, 30.0, 36.2, 30.4, 38.5, -35.6, 34.5),
                         order.by = as.Date(c("2022-05-01", "2022-05-08", "2022-05-15",
                                              "2022-05-22", "2022-05-29", "2022-06-05",
                                              "2022-06-12")))
weekly_return(asset_returns_xts)
```

Index