Package: yieldcurves (via r-universe)

May 30, 2026

Title Yield Curve Fitting, Analysis, and Decomposition

Version 0.1.0

Description Fits yield curves using Nelson-Siegel (1987) <doi:10.1086/296409>, Svensson (1994) <doi:10.3386/w4871>, and cubic spline methods. Extracts forward rates, discount factors, and par rates from fitted curves. Computes duration and convexity risk measures. Computes Z-spread and key rate durations. Provides principal component decomposition following Litterman and Scheinkman (1991) <doi:10.3905/jfi.1991.692347>, carry and roll-down analysis, and slope measures. All methods are pure computation with no external dependencies beyond base R; works with yield data from any source.

Depends R (>= 4.1.0)

License MIT + file LICENSE

Encoding UTF-8

Language en-US

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Imports cli (>= 3.6.0), graphics, stats

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

URL `https://charlescoverdale.github.io/yieldcurves/`, `https://github.com/charlescoverdale/yieldcurves`

BugReports `https://github.com/charlescoverdale/yieldcurves/issues`

Repository https://charlescoverdale.r-universe.dev

Date/Publication 2026-05-30 14:59:05 UTC

RemoteUrl https://github.com/charlescoverdale/yieldcurves

RemoteRef HEAD

RemoteSha 0a745f9825c5a17e37477e8368d982c828c91f45

Contents

`plot.yc_curve` *Plot Method for Yield Curve Objects*

Description

Plot Method for Yield Curve Objects

Usage

```
## S3 method for class 'yc_curve'
plot(x, ...)
```

Arguments

`x`	A `yc_curve` object.
`...`	Additional arguments passed to `plot()`.

Value

The input object, invisibly.

---

plot.yc_pca — *Plot Method for Yield Curve PCA Objects*

Description

Plots the factor loadings for each principal component across tenors.

Usage

```
## S3 method for class 'yc_pca'
plot(x, ...)
```

Arguments

x	A yc_pca object.
...	Additional arguments passed to plot().

Value

The input object, invisibly.

---

print.yc_curve — *Print Method for Yield Curve Objects*

Description

Print Method for Yield Curve Objects

Usage

```
## S3 method for class 'yc_curve'
print(x, ...)
```

Arguments

x	A yc_curve object.
...	Additional arguments (currently unused).

Value

The input object, invisibly.

print.yc_pca *Print Method for Yield Curve PCA Objects*

Description

Print Method for Yield Curve PCA Objects

Usage

```
## S3 method for class 'yc_pca'
print(x, ...)
```

Arguments

`x`	A `yc_pca` object.
`...`	Additional arguments (currently unused).

Value

The input object, invisibly.

summary.yc_curve *Summary Method for Yield Curve Objects*

Description

Summary Method for Yield Curve Objects

Usage

```
## S3 method for class 'yc_curve'
summary(object, ...)
```

Arguments

`object`	A `yc_curve` object.
`...`	Additional arguments (currently unused).

Value

The input object, invisibly.

summary.yc_pca — *Summary Method for Yield Curve PCA Objects*

Description

Summary Method for Yield Curve PCA Objects

Usage

```
## S3 method for class 'yc_pca'
summary(object, ...)
```

Arguments

`object`	A `yc_pca` object.
`...`	Additional arguments (currently unused).

Value

The input object, invisibly.

yc_bond_duration — *Coupon Bond Duration and Convexity*

Description

Compute Macaulay duration, modified duration, and convexity for a coupon-bearing bond.

Usage

```
yc_bond_duration(
  face = 100,
  coupon_rate,
  maturity,
  yield,
  frequency = 2,
  compounding = c("semi_annual", "annual", "continuous")
)
```

Arguments

face	Numeric. Face (par) value of the bond. Default is 100.
coupon_rate	Numeric. Annual coupon rate as a decimal (e.g., 0.05 for 5 percent).
maturity	Numeric. Time to maturity in years.
yield	Numeric. Yield to maturity as a decimal.
frequency	Integer. Coupon frequency per year: 1 for annual or 2 for semi-annual (default).
compounding	Character. Compounding convention: "semi_annual" (default), "annual", or "continuous".

Value

A list with components macaulay_duration, modified_duration, convexity, and price.

Examples

```
# 2-year 5% bond at 4% yield, semi-annual coupons
yc_bond_duration(face = 100, coupon_rate = 0.05, maturity = 2,
                 yield = 0.04, frequency = 2)
```

---

yc_carry *Carry and Roll-Down Analysis*

Description

Decompose expected return from holding a bond into carry (yield income minus financing cost) and roll-down (capital gain from sliding down the curve).

Usage

```
yc_carry(curve, maturities = NULL, horizon = 1/12, funding_rate = NULL)
```

Arguments

curve	A yc_curve object.
maturities	Numeric vector of bond maturities to analyse. If NULL, uses the curve's own maturities (excluding the shortest).
horizon	Numeric. Holding period in years. Default is 1/12 (one month).
funding_rate	Optional numeric. Overnight funding rate as a decimal. If NULL, uses the shortest rate on the curve.

Value

A data frame with columns maturity, carry, rolldown, and total.

Examples

```
maturities <- c(0.25, 1, 2, 5, 10, 30)
rates <- c(0.050, 0.048, 0.045, 0.042, 0.040, 0.043)
fit <- yc_nelson_siegel(maturities, rates)
yc_carry(fit)
```

---

yc_cubic_spline   *Fit Cubic Spline Yield Curve*

Description

Fit a yield curve using cubic spline interpolation. Provides an exact fit through all observed data points with smooth interpolation between them.

Usage

```
yc_cubic_spline(
  maturities,
  rates,
  method = c("natural", "fmm"),
  type = c("zero", "par", "forward"),
  date = NULL
)
```

Arguments

maturities	Numeric vector of maturities in years.
rates	Numeric vector of observed yields as decimals.
method	Character. Spline method: "natural" (default) or "fmm" (Forsythe, Malcolm, and Moler).
type	Character. Rate type: "zero" (default), "par", or "forward".
date	Optional Date for the curve.

Value

A yc_curve object with method = "cubic_spline".

Examples

```
maturities <- c(0.25, 0.5, 1, 2, 5, 10, 30)
rates <- c(0.052, 0.050, 0.048, 0.045, 0.042, 0.040, 0.043)
fit <- yc_cubic_spline(maturities, rates)
fit
```

yc_curve *Create a Yield Curve Object*

Description

Construct a yc_curve object from observed maturity-rate pairs. This is the core data structure used throughout the package.

Usage

```
yc_curve(maturities, rates, type = c("zero", "par", "forward"), date = NULL)
```

Arguments

maturities	Numeric vector of maturities in years (e.g., 0.25 for 3 months, 2 for 2 years).
rates	Numeric vector of yields as decimals (e.g., 0.05 for 5\ Must be the same length as maturities.
type	Character. The type of rate: "zero" (default), "par", or "forward".
date	Optional Date for the curve observation.

Value

A yc_curve object (S3 class) with components:

maturities Numeric vector of maturities in years.

rates Numeric vector of rates as decimals.

type Character string indicating rate type.

method Character string indicating fitting method.

params List of model parameters (empty for observed curves).

fitted Numeric vector of fitted rates (NULL for observed curves).

residuals Numeric vector of residuals (NULL for observed curves).

date Date of the curve observation.

n_obs Integer count of maturity points.

Examples

```
# US Treasury yields (2Y, 5Y, 10Y, 30Y)
maturities <- c(2, 5, 10, 30)
rates <- c(0.045, 0.042, 0.040, 0.043)
curve <- yc_curve(maturities, rates)
curve
```

yc_discount *Compute Discount Factors*

Description

Calculate discount factors from a yield curve assuming continuous compounding.

Usage

```
yc_discount(
  curve,
  maturities = NULL,
  compounding = c("continuous", "annual", "semi_annual")
)
```

Arguments

curve	A yc_curve object.
maturities	Optional numeric vector of maturities. If NULL, uses the curve's own maturities.
compounding	Character. Compounding convention: "continuous" (default), "annual", or "semi_annual".

Value

A data frame with columns maturity and discount_factor.

Examples

```
maturities <- c(1, 2, 5, 10)
rates <- c(0.045, 0.043, 0.042, 0.040)
curve <- yc_curve(maturities, rates)
yc_discount(curve)
yc_discount(curve, compounding = "annual")
```

yc_duration *Duration and Convexity*

Description

Compute Macaulay duration, modified duration, and convexity for zero-coupon bonds at each maturity on the curve.

Usage

```
yc_duration(
  curve,
  maturities = NULL,
  compounding = c("continuous", "annual", "semi_annual")
)
```

Arguments

curve	A yc_curve object.
maturities	Optional numeric vector of maturities. If NULL, uses the curve's own maturities.
compounding	Character. Compounding convention: "continuous" (default), "annual", or "semi_annual".

Value

A data frame with columns maturity, macaulay_duration, modified_duration, and convexity.

Examples

```
maturities <- c(0.25, 1, 2, 5, 10, 30)
rates <- c(0.050, 0.048, 0.045, 0.042, 0.040, 0.043)
fit <- yc_nelson_siegel(maturities, rates)
yc_duration(fit)
```

---

yc_fit — *Fit a Yield Curve*

Description

Unified interface for fitting a yield curve using different methods. Dispatches to yc_nelson_siegel(), yc_svensson(), or yc_cubic_spline().

Usage

```
yc_fit(
  maturities,
  rates,
  method = c("nelson_siegel", "svensson", "cubic_spline"),
  type = c("zero", "par", "forward"),
  date = NULL,
  ...
)
```

Arguments

maturities	Numeric vector of maturities in years.
rates	Numeric vector of observed yields as decimals.
method	Character. Fitting method: "nelson_siegel" (default), "svensson", or "cubic_spline".
type	Character. Rate type: "zero" (default), "par", or "forward".
date	Optional Date for the curve.
...	Additional arguments passed to the fitting function.

Value

A yc_curve object.

Examples

```
maturities <- c(0.25, 0.5, 1, 2, 5, 10, 30)
rates <- c(0.052, 0.050, 0.048, 0.045, 0.042, 0.040, 0.043)
fit <- yc_fit(maturities, rates, method = "nelson_siegel")
fit
```

---

yc_forward — *Extract Forward Rates*

Description

Compute forward rates from a yield curve. Can compute either instantaneous forward rates or forward-forward rates between two tenors.

Usage

```
yc_forward(curve, maturities = NULL, horizon = NULL)
```

Arguments

curve	A yc_curve object.
maturities	Optional numeric vector of maturities at which to compute forward rates. If NULL, uses the curve's own maturities.
horizon	Optional numeric. If provided, computes the forward rate from each maturity to maturity + horizon (forward-forward rate).

Details

The instantaneous forward rate is derived as:

$$f(m) = r(m) + m \cdot r'(m)$$

Value

A data frame with columns maturity and forward_rate.

Examples

```
maturities <- c(0.25, 0.5, 1, 2, 5, 10, 30)
rates <- c(0.052, 0.050, 0.048, 0.045, 0.042, 0.040, 0.043)
fit <- yc_nelson_siegel(maturities, rates)
yc_forward(fit)
yc_forward(fit, maturities = c(1, 5, 10), horizon = 1)
```

---

yc_interpolate *Interpolate Yield Curve*

Description

Interpolate rates at arbitrary maturities from an observed or fitted yield curve.

Usage

```
yc_interpolate(curve, maturities, method = c("linear", "log_linear", "cubic"))
```

Arguments

curve	A yc_curve object.
maturities	Numeric vector of maturities at which to interpolate.
method	Character. Interpolation method: "linear" (default), "log_linear", or "cubic".

Value

A data frame with columns maturity and rate.

Examples

```
maturities <- c(1, 2, 5, 10, 30)
rates <- c(0.045, 0.043, 0.042, 0.040, 0.043)
curve <- yc_curve(maturities, rates)
yc_interpolate(curve, c(3, 7, 15, 20))
```

yc_key_rate_duration *Key Rate Durations*

Description

Compute key rate durations by bumping the yield curve at specific tenors. Each bump is triangular: the full shift is applied at the key rate tenor and linearly interpolated to zero at adjacent key rate tenors.

Usage

```
yc_key_rate_duration(
  coupon_rate,
  maturity,
  curve,
  key_rates = c(1, 2, 5, 10, 30),
  shift = 1e-04,
  face = 100,
  frequency = 2
)
```

Arguments

`coupon_rate`	Numeric. Annual coupon rate as a decimal.
`maturity`	Numeric. Time to maturity in years.
`curve`	Either a `yc_curve` object or a list/data frame with components `maturities` and `rates`.
`key_rates`	Numeric vector of key rate tenors in years. Default is `c(1, 2, 5, 10, 30)`.
`shift`	Numeric. Size of the rate bump in decimal (default 0.0001, i.e. 1 basis point).
`face`	Numeric. Face value. Default is 100.
`frequency`	Integer. Coupon frequency: 1 (annual) or 2 (semi-annual, default).

Value

A data frame with columns `tenor` and `key_rate_duration`.

Examples

```
curve <- yc_curve(c(1, 2, 5, 10, 30), c(0.03, 0.035, 0.04, 0.042, 0.045))
yc_key_rate_duration(coupon_rate = 0.04, maturity = 10,
                     curve = curve, key_rates = c(1, 2, 5, 10, 30))
```

yc_level_slope_curvature — *Extract Level, Slope, and Curvature Factors*

Description

For Nelson-Siegel or Svensson curves, extracts the estimated factors directly from the model parameters. For other curves, computes empirical measures.

Usage

```
yc_level_slope_curvature(curve)
```

Arguments

`curve`	A `yc_curve` object.

Value

A named list with:

level Long-run level (beta0 for NS/Svensson, or mean rate).

slope Slope factor (beta1 for NS/Svensson, or short - long rate).

curvature Curvature factor (beta2 for NS/Svensson, or 2*mid - short - long rate).

Examples

```
maturities <- c(0.25, 0.5, 1, 2, 5, 10, 30)
rates <- c(0.052, 0.050, 0.048, 0.045, 0.042, 0.040, 0.043)
fit <- yc_nelson_siegel(maturities, rates)
yc_level_slope_curvature(fit)
```

yc_nelson_siegel — *Fit Nelson-Siegel Yield Curve*

Description

Estimate a Nelson-Siegel (1987) yield curve model from observed maturity-rate pairs. The model decomposes the yield curve into three factors: level, slope, and curvature.

Usage

```
yc_nelson_siegel(
  maturities,
  rates,
  tau_init = 1,
  weights = NULL,
  type = c("zero", "par", "forward"),
  date = NULL
)
```

Arguments

`maturities`	Numeric vector of maturities in years.
`rates`	Numeric vector of observed yields as decimals.
`tau_init`	Numeric. Initial value for the decay parameter tau. Default is 1.
`weights`	Optional numeric vector of weights for each observation. Must be the same length as `maturities`. Useful for emphasising liquid tenors. If NULL (default), all observations are equally weighted.
`type`	Character. Rate type: `"zero"` (default), `"par"`, or `"forward"`.
`date`	Optional Date for the curve.

Details

The Nelson-Siegel model is:

$$r(m) = \beta_0 + \beta_1 \frac{1 - e^{-m/\tau}}{m/\tau} + \beta_2 \left(\frac{1 - e^{-m/\tau}}{m/\tau} - e^{-m/\tau} \right)$$

Value

A `yc_curve` object with `method = "nelson_siegel"` and `params` containing `beta0`, `beta1`, `beta2`, and `tau`.

References

Nelson, C.R. and Siegel, A.F. (1987). Parsimonious Modeling of Yield Curves. *The Journal of Business*, 60(4), 473–489. doi:10.1086/296409

Examples

```
maturities <- c(0.25, 0.5, 1, 2, 3, 5, 7, 10, 20, 30)
rates <- c(0.052, 0.050, 0.048, 0.045, 0.043, 0.042, 0.041,
           0.040, 0.042, 0.043)
fit <- yc_nelson_siegel(maturities, rates)
fit
```

yc_par_to_zero *Convert Par Rates to Zero Rates*

Description

Bootstrap zero (spot) rates from par (coupon) rates using iterative stripping.

Usage

```
yc_par_to_zero(maturities, par_rates, frequency = 1)
```

Arguments

`maturities`	Numeric vector of maturities in years (must be positive integers or half-years).
`par_rates`	Numeric vector of par rates as decimals.
`frequency`	Integer. Coupon frequency per year: 1 for annual (default) or 2 for semi-annual.

Value

A data frame with columns `maturity` and `zero_rate`.

Examples

```
maturities <- c(1, 2, 3, 5, 10)
par_rates <- c(0.040, 0.042, 0.043, 0.044, 0.045)
yc_par_to_zero(maturities, par_rates)

# Semi-annual coupons
yc_par_to_zero(c(0.5, 1, 2), c(0.04, 0.042, 0.043), frequency = 2)
```

yc_pca *Principal Component Analysis of Yield Curves*

Description

Perform PCA on a time series of yield curves to extract the dominant factors (level, slope, curvature) following Litterman and Scheinkman (1991).

Usage

```
yc_pca(curves_matrix, n_components = 3, scale = FALSE)
```

Arguments

`curves_matrix` Numeric matrix where each row is a yield curve observation (e.g., daily curves) and each column is a tenor. Column names should be maturity labels.

`n_components` Integer. Number of principal components to retain. Default is 3 (level, slope, curvature).

`scale` Logical. Whether to scale variables before PCA. Default is `FALSE` (use covariance matrix, standard in yield curve PCA).

Value

A yc_pca object (S3 class) with components:

loadings Matrix of factor loadings (tenors x components).

scores Matrix of factor scores (observations x components).

variance_explained Numeric vector of proportion of variance explained by each component.

cumulative_variance Numeric vector of cumulative variance explained.

sdev Standard deviations of each component.

n_components Number of components retained.

tenors Column names from the input matrix.

References

Litterman, R. and Scheinkman, J. (1991). Common Factors Affecting Bond Returns. *The Journal of Fixed Income*, 1(1), 54–61. doi:10.3905/jfi.1991.692347

Examples

```
# Simulate 100 days of yield curves at 5 tenors
set.seed(42)
n_days <- 100
tenors <- c(1, 2, 5, 10, 30)
base_rates <- c(0.045, 0.043, 0.042, 0.040, 0.043)
curves <- matrix(NA, n_days, length(tenors))
colnames(curves) <- paste0(tenors, "Y")
level <- cumsum(rnorm(n_days, 0, 0.001))
slope <- cumsum(rnorm(n_days, 0, 0.0005))
for (i in seq_len(n_days)) {
  curves[i, ] <- base_rates + level[i] + slope[i] * (tenors - mean(tenors)) / 30
}
pca_result <- yc_pca(curves)
pca_result
```

yc_predict — *Predict Rates from a Fitted Yield Curve*

Description

Evaluate a fitted yield curve at new maturities.

Usage

```
yc_predict(curve, maturities)
```

Arguments

`curve`	A `yc_curve` object from a fitting function.
`maturities`	Numeric vector of maturities at which to predict rates.

Value

A data frame with columns `maturity` and `rate`.

Examples

```
maturities <- c(0.25, 0.5, 1, 2, 5, 10, 30)
rates <- c(0.052, 0.050, 0.048, 0.045, 0.042, 0.040, 0.043)
fit <- yc_nelson_siegel(maturities, rates)
yc_predict(fit, c(3, 7, 15, 20))
```

yc_slope — *Yield Curve Slope Measures*

Description

Compute common slope and curvature measures from a yield curve.

Usage

```
yc_slope(curve)
```

Arguments

`curve`	A `yc_curve` object.

Value

A named list with slope measures:

spread_2s10s 10-year minus 2-year rate (the most common slope measure).

spread_2s30s 30-year minus 2-year rate.

spread_5s30s 30-year minus 5-year rate.

spread_3m10y 10-year minus 3-month rate (term premium proxy).

butterfly_2s5s10s 2 * 5-year minus 2-year minus 10-year (curvature measure).

Returns NA for any measure whose required tenors fall outside the curve range.

Examples

```
maturities <- c(0.25, 0.5, 1, 2, 5, 10, 30)
rates <- c(0.052, 0.050, 0.048, 0.045, 0.042, 0.040, 0.043)
fit <- yc_nelson_siegel(maturities, rates)
yc_slope(fit)
```

`yc_svensson` *Fit Svensson Yield Curve*

Description

Estimate a Svensson (1994) yield curve model from observed maturity-rate pairs. Extends Nelson-Siegel by adding a second curvature term with its own decay parameter, providing greater flexibility for curves with two humps.

Usage

```
yc_svensson(
  maturities,
  rates,
  tau1_init = 1,
  tau2_init = 5,
  weights = NULL,
  type = c("zero", "par", "forward"),
  date = NULL
)
```

Arguments

`maturities`	Numeric vector of maturities in years.
`rates`	Numeric vector of observed yields as decimals.
`tau1_init`	Numeric. Initial value for the first decay parameter. Default is 1.
`tau2_init`	Numeric. Initial value for the second decay parameter. Default is 5.

`weights`	Optional numeric vector of weights for each observation. Must be the same length as `maturities`. If NULL (default), all observations are equally weighted.
`type`	Character. Rate type: `"zero"` (default), `"par"`, or `"forward"`.
`date`	Optional Date for the curve.

Value

A `yc_curve` object with `method = "svensson"` and `params` containing `beta0`, `beta1`, `beta2`, `beta3`, `tau1`, and `tau2`.

References

Svensson, L.E.O. (1994). Estimating and Interpreting Forward Interest Rates: Sweden 1992–1994. *NBER Working Paper*, 4871. doi:10.3386/w4871

Examples

```
maturities <- c(0.25, 0.5, 1, 2, 3, 5, 7, 10, 20, 30)
rates <- c(0.052, 0.050, 0.048, 0.045, 0.043, 0.042, 0.041,
           0.040, 0.042, 0.043)
fit <- yc_svensson(maturities, rates)
fit
```

`yc_zero_to_par` *Convert Zero Rates to Par Rates*

Description

Compute par (coupon) rates from zero (spot) rates. The par rate for maturity T is the coupon rate that makes a bond price equal to par.

Usage

```
yc_zero_to_par(maturities, zero_rates, frequency = 1)
```

Arguments

`maturities`	Numeric vector of maturities in years.
`zero_rates`	Numeric vector of zero rates as decimals.
`frequency`	Integer. Coupon frequency per year: 1 for annual (default) or 2 for semi-annual.

Value

A data frame with columns `maturity` and `par_rate`.

Examples

```
maturities <- c(1, 2, 3, 5, 10)
zero_rates <- c(0.040, 0.042, 0.043, 0.044, 0.045)
yc_zero_to_par(maturities, zero_rates)

# Semi-annual coupons
yc_zero_to_par(c(0.5, 1, 2), c(0.04, 0.042, 0.043), frequency = 2)
```

---

yc_zspread *Z-Spread*

Description

Compute the Z-spread (zero-volatility spread) for a bond. The Z-spread is the constant spread added to each zero rate on the benchmark curve that makes the discounted cash flows equal the market price.

Usage

```
yc_zspread(price, coupon_rate, maturity, curve, face = 100, frequency = 2)
```

Arguments

price	Numeric. Market price of the bond.
coupon_rate	Numeric. Annual coupon rate as a decimal.
maturity	Numeric. Time to maturity in years.
curve	Either a yc_curve object or a list/data frame with components maturities and rates.
face	Numeric. Face value of the bond. Default is 100.
frequency	Integer. Coupon frequency per year: 1 for annual or 2 for semi-annual (default).

Value

A list with components zspread (the Z-spread as a decimal), price (the input price), and model_price (the price implied by the curve with the Z-spread applied).

Examples

```
# Create a benchmark curve
curve <- yc_curve(c(0.5, 1, 2, 5, 10), c(0.03, 0.035, 0.04, 0.042, 0.045))

# A bond priced below par (positive Z-spread)
yc_zspread(price = 95, coupon_rate = 0.04, maturity = 5,
           curve = curve, frequency = 2)
```

Index