Package ‘IOL’

March 23, 2014

Type Package

Title Ophthalmology IOL power calculations

Version 0.1.0

Date 2013-10-30

Maintainer Eric N. Brown <eric.n.brown@gmail.com>

Description More about what it does (maybe more than one line)

License AGPL-3

Depends testthat

Roxygen list(wrap = TRUE)

R topics documented:

`IOL-package` *Calculates ophthalmic intraocular lens powers based on biometry and lens parameters.*

Description

A package to calculate ophthalmologic intraocular lens (IOL) powers.

Author(s)

Eric N. Brown `<eric.n.brown@gmail.com>`

See Also

`ELP`, `Power`

`angle` *Compute the angle between two lines*

Description

Internal function for ray-tracing 2D optics

Usage

```
angle(A, B)
```

Arguments

`A`	First line
`B`	Second line

Value

Angle in radians (scalar)

See Also

Other optics: `Circle`; `IntersectLC`; `Line`; `Point`; `RefractLC`; `RefractLL`; `center`; `length.Line`; `normalize`; `orthogonal`; `plot.Circle`; `plot.Line`; `plot.Point`; `rotate`; `scale`

`Binkhorst.adjusted.Power` *Adjusted Binkhorst Formula for IOL Power*

Description

Calculate IOL power for emmetropia given axial length, corneal radius of curvature, and effective lens position. If corneal power (in diopters) is provided instead of the radius of curvature, then the radius of curvature will be computed from the corneal power and corneal index of refraction. A standard corneal index of refraction will be used if one isn't provided for this conversion.

Usage

```
Binkhorst.adjusted.Power(L, R, K, cornea_n, ELP)
```

Arguments

`L`	axial length in millimeters (mm)
`R`	corneal radius of curvature in millimeters (mm)
`K`	corneal curvature in diopters (D)
`cornea_n`	corneal index of refraction
`ELP`	effective lens position in millimeters (mm)

Details

This is an adjustment to the Binkhorst formula to correct for axial length.

Value

IOL power for emmetropia

References

Binkhorst RD. Intraocular Lens Power Calculation Manual: A Guide to The Author's TICC-40 Programs. 3rd Ed. New York: R. D. Binkhorst, 1984.

https://encrypted.google.com/books?id=NhWJsGFK6qgC&pg=PA11

See Also

Power, Binkhorst.Power

Other Power: Binkhorst.Power; Colenbrander.Power; Fyodorov.Power; Haigis.Power; Hoffer.Power; Hoffer.Q.Power; Holladay.1.Power; Holladay.Power; Power; SRK.II.Power; SRK.Power; SRK.T.Power; Shammas.Power; van.der.Heijde.Power

---

Binkhorst.Power *Binkhorst Formula for IOL Power*

Description

Calculate IOL power for emmetropia given axial length, corneal radius of curvature, and effective lens position. If corneal power (in diopters) is provided instead of the radius of curvature, then the radius of curvature will be computed from the corneal power and corneal index of refraction. A standard corneal index of refraction will be used if one isn't provided for this conversion.

Usage

```
Binkhorst.Power(L, R, K, cornea_n, ELP)
```

Arguments

L	axial length in millimeters (mm)
R	corneal radius of curvature in millimeters (mm)
K	corneal curvature in diopters (D)
cornea_n	corneal index of refraction
ELP	effective lens position in millimeters (mm)

Value

IOL power for emmetropia

References

Binkhorst RD. The optical design of intraocular lens implants. Ophthalmic Surg. Fall 1975; 6(3): 17–31. PMID: 1187085.

https://encrypted.google.com/books?id=NhWJsGFK6qgC&pg=PA8

See Also

Power, Binkhorst.adjusted.Power

Other Power: Binkhorst.adjusted.Power; Colenbrander.Power; Fyodorov.Power; Haigis.Power; Hoffer.Power; Hoffer.Q.Power; Holladay.1.Power; Holladay.Power; Power; SRK.II.Power; SRK.Power; SRK.T.Power; Shammas.Power; van.der.Heijde.Power `center` *Center a line by translating so the start point is the origin*

Description

Internal function for ray-tracing 2D optics

Usage

```
center(line)
```

Value

Object of Line class

See Also

Other optics: `Circle`; `IntersectLC`; `Line`; `Point`; `RefractLC`; `RefractLL`; `angle`; `length.Line`; `normalize`; `orthogonal`; `plot.Circle`; `plot.Line`; `plot.Point`; `rotate`; `scale`

`Circle` *Circle class*

Description

Internal class for ray-tracing 2D optics

Usage

```
Circle(center, radius)
```

Arguments

`center`	Center of the circle (Point)
`radius`	Radius of the circle (scalar)

Value

Object of Circle class

See Also

Other optics: `IntersectLC`; `Line`; `Point`; `RefractLC`; `RefractLL`; `angle`; `center`; `length.Line`; `normalize`; `orthogonal`; `plot.Circle`; `plot.Line`; `plot.Point`; `rotate`; `scale`

`Colenbrander.Power` *Colenbrander Formula for IOL Power*

Description

Calculate IOL power for emmetropia given axial length, corneal power, and effective lens position.

Usage

```
Colenbrander.Power(L, K, ELP)
```

Arguments

`L`	axial length in millimeters (mm)
`K`	corneal curvature in diopters (D)
`ELP`	effective lens position in millimeters (mm)

Value

IOL power for emmetropia

References

`https://encrypted.google.com/books?id=NhWJsGFK6qgC&pg=PA8`

See Also

`Power`

Other Power: `Binkhorst.Power`; `Binkhorst.adjusted.Power`; `Fyodorov.Power`; `Haigis.Power`; `Hoffer.Power`; `Hoffer.Q.Power`; `Holladay.1.Power`; `Holladay.Power`; `Power`; `SRK.II.Power`; `SRK.Power`; `SRK.T.Power`; `Shammas.Power`; `van.der.Heijde.Power`

`Constants` *Constants for IOL calculations and biometry*

Description

Constants for IOL calculations and biometry

Usage

```
Constants
```

Format

```
List of 3
 $ IOL     :List of 8
  ..$ A_to_pACD_a0: num -63.9
  ..$ A_to_pACD_a1: num 0.584
  ..$ A_to_S_a0   : num -65.6
  ..$ A_to_S_a1   : num 0.566
  ..$ pACD_to_S_a0: num -3.6
  ..$ pACD_to_S_a1: num 0.97
  ..$ A_to_ACD_a0 : num -68.7
  ..$ A_to_ACD_a1 : num 0.625
 $ Average :List of 10
  ..$ L                : num 23.5
  ..$ L.sd             : num 1.25
  ..$ K                : num 43.8
  ..$ K.sd             : num 1.6
  ..$ ACD              : num 3.1
  ..$ ACD.sd           : num 0.3
  ..$ lens.thickness   : num 4.7
  ..$ lens.thickness.sd: num 0.41
  ..$ age              : num 72
  ..$ age.sd           : num 12
 $ Biometry:List of 14
  ..$ us_to_pci              : num 0.15
  ..$ us_1532_to_us          : num 0.32
  ..$ us_applination_to_us   : num 0.2
  ..$ immersion_us_to_pci_a1 : num 0.995
  ..$ immersion_us_to_pci_a0 : num 0.0779
  ..$ us_immersion_error     : num 0.12
  ..$ us_applination_error   : num 0.13
  ..$ us_immersion_1532      : num 0.13
  ..$ pci_iolmaster_error    : num 0.02
  ..$ corneal_index          : num 1.34
  ..$ corneal_index_2        : num 1.33
  ..$ aqueous_index          : num 1.34
  ..$ retinal_thickness      : num 0.2
  ..$ corneal_principle_plane: num 0.05
```

ELP — *Calculate Effective Lens Position (ELP) from Biometry and IOL Data*

Description

Using specified ocular biometry data including a measure of the eye's optical length and corneal curvature, this function calculates one or more estimates of an intraocular lens's effective lens position (ELP). The ELP is the position in millimeters of the IOL's principle plane from the cornea's principle plane. The ELP used to be referred to as the anterior chamber depth (ACD); however, ELP is a more accurate description.

Usage

```
ELP(L, K, R = NA, cornea_n = NA, ACD = NA, A = NA, pACD = NA,
  S = NA, a0 = NA, a1, a2, which = "modern")
```

Arguments

`L`	length of the eye in millimeters (mm)
`K`	average corneal curvature of the eye in diopters (D)
`R`	average corneal radius in millimeters (mm)
`cornea_n`	effective corneal index of refraction
`ACD`	ultrasound anterior chamber depth (mm)
`A`	IOL A constant (D)
`pACD`	IOL pACD constant (mm)
`S`	IOL surgeon factor constant
`a0`	Haigis formula a0 lens constant (mm)
`a1`	Haigis formula a1 lens constant
`a2`	Haigis formula a2 lens constant
`which`	string vector specifying which ELP formulas to use

Value

Named numeric vector of effective lens position (in mm) for each ELP formula requested.

Note

If the some of the IOL constants A, pACD, or S are not provided, it may be derived from those given. A warning is generally produced when this conversion is performed.

The returned numeric vector is augmented with a `'parameters'` list attribute describing which biometry and IOL parameters were used to calculate each value. For example, if the Hoffer Q ELP is calculated given an axial length, corneal power (via the `K` parameter), and an A-constant (via the `A` parameter), then the `'parameters'` list attribute will have a list named `Hoffer.Q` with three values: `L`, `K`, and `A`.

Author(s)

Eric N. Brown <eric.n.brown@gmail.com>

See Also

`Power`

Other ELP: `Haigis.ELP`; `Hoffer.ELP`; `Hoffer.Q.ELP`; `Holladay.1.ELP`; `Holladay.ELP`; `SRK.T.ELP`

Examples

```
# Get the effective lens position of a normal eye with for the
# Alcon SA60AT lens using the Hoffer Q formula. This will compute
# the required pACD IOL constant from the provided A constant.
(elp <- ELP(L = 24, K = 44, A = 118.4, which = 'Hoffer.Q'))

# Check which parameters were used to calculate the ELP
```

```
attr(elp, 'parameters')$Hoffer.Q

# Get the ELP of a normal eye with modern formulas (Hoffer Q, SRK/T,
# and Holladay 1). Five warnings will be output for the conversion of
# the A constant to ACD for the SRK/T and Hoffer Q formulas, using a
# standard corneal index of refraction (since one wasn't provided) to
# convert corneal power to radius of curvature, and for approximating
# the Holladay 1 surgeon factor from the provided A constant.
(elp <- ELP(L = 24, K = 44, A = 118.4, which = 'modern'))

# Get the ELP of a normal eye with the Holladay 1 formula. Although
# both the IOL A constant and surgeon factor are provided, since the
# formula requires the surgeon factor, the A constant will be ignored
(elp <- ELP(L = 24, K = 44, A = 118.4, S = 1.45, which = 'Holladay.1'))
attr(elp, 'parameters')$Holladay.1
```

`Fyodorov.Power` *Fyodorov Formula for IOL Power*

Description

Calculate IOL power for emmetropia given axial length, corneal power, and effective lens position.

Usage

```
Fyodorov.Power(L, K, ELP)
```

Arguments

`L`	axial length in millimeters (mm)
`K`	corneal curvature in diopters (D)
`ELP`	effective lens position in millimeters (mm)

Value

IOL power for emmetropia

References

https://encrypted.google.com/books?id=NhWJsGFK6qgC&pg=PA8

See Also

`Power`

Other Power: `Binkhorst.Power`; `Binkhorst.adjusted.Power`; `Colenbrander.Power`; `Haigis.Power`; `Hoffer.Power`; `Hoffer.Q.Power`; `Holladay.1.Power`; `Holladay.Power`; `Power`; `SRK.II.Power`; `SRK.Power`; `SRK.T.Power`; `Shammas.Power`; `van.der.Heijde.Power`

Haigis.ELP *Haigis Formula for Effective Lens Position*

Description

Calculate IOL effective lens position for emmetropia given axial length (L), anterior chamber depth (ACD), and a lens constant (a0, pACD, or A must be supplied.) If pACD is given, it is converted to an approximate a0 constant. If A is given, it is first convert to pACD and then to an approximate a0 constant.

Usage

```
Haigis.ELP(L, ACD = 3.37, a0, a1 = 0.4, a2 = 0.1, pACD, A)
```

Arguments

`L`	ultrasound axial length (mm)
`ACD`	ultrasound anterior chamber depth (mm)
`a0`	Haigis a0 lens constant (mm)
`a1`	Haigis a1 lens constant (unitless, defaults to 0.4)
`a2`	Haigis a2 lens constant (unitless, defaults to 0.1)
`pACD`	personalized ACD (mm) lens constant
`A`	SRK A lens constant (D)

Details

Note: Currently this is just the provided pACD constant.

Value

ELP in mm

References

http://www.augenklinik.uni-wuerzburg.de/uslab/ioltxt/haie.htm

See Also

`ELP`

Other ELP: `ELP`; `Hoffer.ELP`; `Hoffer.Q.ELP`; `Holladay.1.ELP`; `Holladay.ELP`; `SRK.T.ELP`

`Haigis.Power` *Haigis Formula for Emmetropic IOL Power*

Description

Calculate IOL power for emmetropia given axial length, corneal curvature, and effective lens position.

Usage

```
Haigis.Power(L, R, ELP, Rx = 0, V = 12)
```

Arguments

`L`	length of the eye in millimeters (mm)
`R`	average corneal radius in millimeters (mm)
`ELP`	IOL effective lens position (mm)
`Rx`	resulting refractive error (D), defaults to 0 D
`V`	resulting refractive error vertex distance (mm), defaults to 12 mm

Value

Power of emmetropic IOL (D)

See Also

`Power`

Other Power: `Binkhorst.Power`; `Binkhorst.adjusted.Power`; `Colenbrander.Power`; `Fyodorov.Power`; `Hoffer.Power`; `Hoffer.Q.Power`; `Holladay.1.Power`; `Holladay.Power`; `Power`; `SRK.II.Power`; `SRK.Power`; `SRK.T.Power`; `Shammas.Power`; `van.der.Heijde.Power`

`Hoffer.ELP` *Hoffer Formula for Effective Lens Position*

Description

Calculate IOL effective lens position for emmetropia given axial length and lens A-constant or pACD constant.

Usage

```
Hoffer.ELP(L, ACD, A)
```

Arguments

`L`	length of the eye in millimeters (mm)
`ACD`	IOL personalized ACD constant (mm)
`A`	IOL A constant (D) used to determine ACD

Details

Note: If the pACD constant, `ACD`, for the lens is not provided, it is calculated from the lens A-constant, `A`, using the Holladay approximation.

Value

ELP in mm

References

https://encrypted.google.com/books?id=NhWJsGFK6qgC&pg=PA8

See Also

`ELP`

Other ELP: `ELP`; `Haigis.ELP`; `Hoffer.Q.ELP`; `Holladay.1.ELP`; `Holladay.ELP`; `SRK.T.ELP`

`Hoffer.Power` *Hoffer Formula for Emmetropic IOL Power*

Description

Calculate IOL power for emmetropia given axial length, corenal curvature, and effective lens position.

Usage

```
Hoffer.Power(L, K, ELP)
```

Arguments

`L`	length of the eye in millimeters (mm)
`K`	average corneal curvature of the eye in diopters (D)
`ELP`	IOL effective lens position (mm)

Value

Power of emmetropic IOL (D)

References

https://encrypted.google.com/books?id=NhWJsGFK6qgC&pg=PA8

See Also

`Power`

Other Power: `Binkhorst.Power`; `Binkhorst.adjusted.Power`; `Colenbrander.Power`; `Fyodorov.Power`; `Haigis.Power`; `Hoffer.Q.Power`; `Holladay.1.Power`; `Holladay.Power`; `Power`; `SRK.II.Power`; `SRK.Power`; `SRK.T.Power`; `Shammas.Power`; `van.der.Heijde.Power`

`Hoffer.Q.ELP` *Hoffer-Q Formula for Effective Lens Position*

Description

Calculate IOL effective lens position for emmetropia given axial length, corenal curvature, and lens A-constant or pACD constant.

Usage

```
Hoffer.Q.ELP(L, K, pACD, A)
```

Arguments

`L`	length of the eye in millimeters (mm)
`K`	average corneal curvature of the eye in diopters (D)
`pACD`	IOL personalized ACD constant (mm)
`A`	IOL A constant (D) used to determine pACD

Details

Note: If the pACD constant, `pACD` for the lens is not provided, it is calculated from the lens A-constant, `A`, using the Holladay approximation.

Value

ELP in mm

See Also

`ELP`

Other ELP: `ELP`; `Haigis.ELP`; `Hoffer.ELP`; `Holladay.1.ELP`; `Holladay.ELP`; `SRK.T.ELP`

`Hoffer.Q.Power` *Hoffer-Q Formula for Emmetropic IOL Power*

Description

Calculate IOL power for emmetropia given axial length, corenal curvature, and effective lens position.

Usage

```
Hoffer.Q.Power(L, K, ELP, Rx = 0, V = 13)
```

Arguments

L	length of the eye in millimeters (mm)
K	average corneal curvature of the eye in diopters (D)
ELP	IOL effective lens position (mm)
Rx	resulting refractive error (D), defaults to 0 D
V	resulting refractive error vertex distance (mm), defaults to 13 mm

Value

Power of emmetropic IOL (D)

See Also

Power

Other Power: Binkhorst.Power; Binkhorst.adjusted.Power; Colenbrander.Power; Fyodorov.Power; Haigis.Power; Hoffer.Power; Holladay.1.Power; Holladay.Power; Power; SRK.II.Power; SRK.Power; SRK.T.Power; Shammas.Power; van.der.Heijde.Power

---

Holladay.1.ELP — *Holladay 1 Formula for Effective Lens Position*

Description

Calculate IOL effective lens position for emmetropia given axial length, corneal radius of curvature, and lens constant.

Usage

```
Holladay.1.ELP(L, R, S, K, cornea_n, A, pACD)
```

Arguments

L	length of the eye in millimeters (mm)
R	corneal radius of curvature (mm)
S	IOL surgeon-factor constant (mm)
K	corneal power (D) used to determine corneal radius of curvature
cornea_n	corneal index of refraction used to determine corneal radius of curvature from K
A	IOL A constant (D) used to determine surgeon-factor
pACD	IOL personalized ACD constant (mm) used to determine surgeon-factor

Details

Note: If the Holladay Surgeon-factor constant for the lens is not provided, it is calculated from the lens A-constant or pACD-constant. Similarily, if the corneal radius of curvature is not provided, it is calculated from the corneal curvature (in diopters) and corneal index of refraction.

Value

ELP in mm

References

https://encrypted.google.com/books?id=NhWJsGFK6qgC&pg=PA8

See Also

ELP

Other ELP: ELP; Haigis.ELP; Hoffer.ELP; Hoffer.Q.ELP; Holladay.ELP; SRK.T.ELP

`Holladay.1.Power` *Holladay-1 Formula for IOL Power*

Description

Calculate IOL power for emmetropia given axial length, corenal curvature, and effective lens position.

Usage

```
Holladay.1.Power(L, R, ELP, K, cornea_n = 1.3375, aqueous_n = 1.336,
  RT = 0.2, Rx = 0, V = 13)
```

Arguments

`L`	length of the eye in millimeters (mm)
`R`	corneal radius of curvature (mm)
`ELP`	IOL effective lens position (mm)
`K`	corneal power (D) used to determine corneal radius of curvature
`cornea_n`	corneal index of refraction used to determine corneal radius of curvature from K, defaults to 1.3375
`aqueous_n`	aqueous index of refraction, defaults to 1.3375
`RT`	retinal thickness (mm), defaults to 0.2 mm
`Rx`	desired resulting refractive error (D), defaults to 0 D
`V`	vertex distance of Rx (mm), defaults to 13 mm

Details

Note: if the corneal radius, `R`, is not provided, it is determined from the corneal power, `K`, and the corneal index of refraction, `cornea_n`.

Value

Power of IOL (D)

References

https://encrypted.google.com/books?id=NhWJsGFK6qgC&pg=PA8

See Also

Power

Other Power: Binkhorst.Power; Binkhorst.adjusted.Power; Colenbrander.Power; Fyodorov.Power; Haigis.Power; Hoffer.Power; Hoffer.Q.Power; Holladay.Power; Power; SRK.II.Power; SRK.Power; SRK.T.Power; Shammas.Power; van.der.Heijde.Power

---

Holladay.ELP *Holladay Formula for Effective Lens Position*

Description

Calculate IOL effective lens position for emmetropia given lens A-constant.

Usage

```
Holladay.ELP(A)
```

Arguments

A	IOL A-constant (D)

Value

ELP in mm

References

http://books.google.com/books?id=SHjgbQ9auqYC&pg=PT93

See Also

ELP

Other ELP: ELP; Haigis.ELP; Hoffer.ELP; Hoffer.Q.ELP; Holladay.1.ELP; SRK.T.ELP

`Holladay.Power` *Holladay Formula for IOL Power*

Description

Calculate IOL power for emmetropia given axial length, corenal curvature, and effective lens position.

Usage

```
Holladay.Power(L, K, ELP, Rx = 0, V = 13)
```

Arguments

`L`	length of the eye in millimeters (mm)
`K`	corneal power (D)
`ELP`	IOL effective lens position (mm)
`Rx`	desired resulting refractive error (D), defaults to 0 D
`V`	vertex distance of Rx (mm), defaults to 13 mm

Value

Power of IOL (D)

References

http://books.google.com/books?id=SHjgbQ9auqYC&pg=PT93

See Also

`Power`

Other Power: `Binkhorst.Power`; `Binkhorst.adjusted.Power`; `Colenbrander.Power`; `Fyodorov.Power`; `Haigis.Power`; `Hoffer.Power`; `Hoffer.Q.Power`; `Holladay.1.Power`; `Power`; `SRK.II.Power`; `SRK.Power`; `SRK.T.Power`; `Shammas.Power`; `van.der.Heijde.Power`

`IntersectLC` *Intersect and Line and Circle*

Description

Internal function for ray-tracing 2D optics

Usage

```
IntersectLC(line, circle)
```

Arguments

line	Line to intersect (treated as an infinite line)
circle	Circle to intersect

Value

A list of 0, 1, or 2 Points of intersection between the line and circle

See Also

Other optics: Circle; Line; Point; RefractLC; RefractLL; angle; center; length.Line; normalize; orthogonal; plot.Circle; plot.Line; plot.Point; rotate; scale

---

length.Line *Return the length of a line*

Description

Internal function for ray-tracing 2D optics

Usage

```
## S3 method for class 'Line'
length(line)
```

Value

Length (scalar)

See Also

Other optics: Circle; IntersectLC; Line; Point; RefractLC; RefractLL; angle; center; normalize; orthogonal; plot.Circle; plot.Line; plot.Point; rotate; scale

---

Line *Line class*

Description

Internal class for ray-tracing 2D optics

Usage

```
Line(a, b)
```

Arguments

A	Starting point (of Point class)
B	Ending point (of Point class)

Value

Object of Line class

See Also

Other optics: `Circle`; `IntersectLC`; `Point`; `RefractLC`; `RefractLL`; `angle`; `center`; `length.Line`; `normalize`; `orthogonal`; `plot.Circle`; `plot.Line`; `plot.Point`; `rotate`; `scale`

`normalize` *Normalize a line by scaling it to be unit length*

Description

Internal function for ray-tracing 2D optics

Usage

```
normalize(line)
```

Value

Object of Line class

See Also

Other optics: `Circle`; `IntersectLC`; `Line`; `Point`; `RefractLC`; `RefractLL`; `angle`; `center`; `length.Line`; `orthogonal`; `plot.Circle`; `plot.Line`; `plot.Point`; `rotate`; `scale`

`orthogonal` *Return an orthogonal line*

Description

Internal function for ray-tracing 2D optics

Usage

```
orthogonal(line)
```

Value

Object of Line class

See Also

Other optics: `Circle`; `IntersectLC`; `Line`; `Point`; `RefractLC`; `RefractLL`; `angle`; `center`; `length.Line`; `normalize`; `plot.Circle`; `plot.Line`; `plot.Point`; `rotate`; `scale` `plot.Circle` *Plot a Circle class on an existing plot*

Description

Internal function for ray-tracing 2D optics

Usage

```
## S3 method for class 'Circle'
plot(c, points = 100, around = NULL, ...)
```

Arguments

`points`	Number of points with which to approximate a circle
`around`	Optional Point or Line to limit the extent of the drawn circle
`...`	Other options to pass to the base graphics lines function

See Also

Other optics: `Circle`; `IntersectLC`; `Line`; `Point`; `RefractLC`; `RefractLL`; `angle`; `center`; `length.Line`; `normalize`; `orthogonal`; `plot.Line`; `plot.Point`; `rotate`; `scale`

`plot.Line` *Plot a Line class on an existing plot*

Description

Internal function for ray-tracing 2D optics

Usage

```
## S3 method for class 'Line'
plot(ln, ...)
```

Arguments

`...`	Other options to pass to the base graphics segments function

See Also

Other optics: `Circle`; `IntersectLC`; `Line`; `Point`; `RefractLC`; `RefractLL`; `angle`; `center`; `length.Line`; `normalize`; `orthogonal`; `plot.Circle`; `plot.Point`; `rotate`; `scale`

`plot.Point` *Plot a Point class on an existing plot*

Description

Internal function for ray-tracing 2D optics

Usage

```
## S3 method for class 'Point'
plot(pt, ...)
```

Arguments

`...`	Other options to pass to the base graphics points function

See Also

Other optics: `Circle`; `IntersectLC`; `Line`; `Point`; `RefractLC`; `RefractLL`; `angle`; `center`; `length.Line`; `normalize`; `orthogonal`; `plot.Circle`; `plot.Line`; `rotate`; `scale`

`Point` *Point class*

Description

Internal class for ray-tracing 2D optics

Usage

```
Point(x, y)
```

Arguments

`x`	x coordinate
`y`	y coordinate

Value

Object of Point class

See Also

Other optics: `Circle`; `IntersectLC`; `Line`; `RefractLC`; `RefractLL`; `angle`; `center`; `length.Line`; `normalize`; `orthogonal`; `plot.Circle`; `plot.Line`; `plot.Point`; `rotate`; `scale`

Power *Calculate IOL Power from Biometry Data and ELP*

Description

Using specified ocular biometry data including a measure of the eye's optical length and corneal curvature, this function calculates one or more estimates of an intraocular lens's effective lens position (ELP). The ELP is the position in millimeters of the IOL's principle plane from the cornea's principle plane. The ELP used to be referred to as the anterior chamber depth (ACD); however, ELP is a more accurate description.

Usage

```
Power(L, K, A, ELP, Rx = 0, V = 13, which = "all")
```

Arguments

L	length of the eye in millimeters (mm)
K	average corneal curvature of the eye in diopters (D)
A	IOL A constant
ELP	effective lens position in millimeters (mm)
Rx	desired refractive outcome in diopters (D), defaults to emmetropia
V	vertex of refractive outcome in millimeters (mm), defaults to 13 mm
which	string vector specifying which IOL power formulas to use

Value

Named numeric vector of optimal IOL powers (in diopters, D) for each formula requested

Author(s)

Eric N. Brown <eric.n.brown@gmail.com>

See Also

ELP

Other Power: Binkhorst.Power; Binkhorst.adjusted.Power; Colenbrander.Power; Fyodorov.Power; Haigis.Power; Hoffer.Power; Hoffer.Q.Power; Holladay.1.Power; Holladay.Power; SRK.II.Power; SRK.Power; SRK.T.Power; Shammas.Power; van.der.Heijde.Power

RefractLC *Refract a line into a circle*

Description

Internal function for ray-tracing 2D optics.

Usage

```
RefractLC(line, circle, n1 = 1, n2 = 1, first = TRUE, inside = !first,
  plot = TRUE, ...)
```

Arguments

`line`	Infinite line to refract
`normal`	Normal of the refracting surface (the surface is located at the normal's starting point)
`n1`	Index of refraction for the incident ray
`n2`	Index of refraction for the refracted ray
`first`	Whether to refract at the first intersection of the line and circle or the second.
`inside`	Whether the incident line is leaving the circle (starting inside) and the refracting ray is outside of the circle
`plot`	Whether to plot the incident ray, surface normal, and refracted ray

Details

Refracts a line into a circle by first computing the intersection of the line and the circle and then computing the surface normal of the circle at that intersection point. The parameters first and inside choose whether the first (left/top) or second (right/bottom) intersection of the line with the circle is the chosen refraction surface. The inside parameter determines whether the refraction is taken as a ray intersecting in-to or out-of the circle.

Value

The refracted ray with the starting point being the normal's starting point (class Line)

See Also

Other optics: `Circle`; `IntersectLC`; `Line`; `Point`; `RefractLL`; `angle`; `center`; `length.Line`; `normalize`; `orthogonal`; `plot.Circle`; `plot.Line`; `plot.Point`; `rotate`; `scale`

RefractLL *Refract a line given a normal direction and two index of refractions*

Description

Internal function for ray-tracing 2D optics.

Usage

```
RefractLL(line, normal, n1 = 1, n2 = 1, plot = TRUE, ...)
```

Arguments

`line`	Infinite line to refract
`normal`	Normal of the refracting surface (the surface is located at the normal's starting point)
`n1`	Index of refraction for the incident ray
`n2`	Index of refraction for the refracted ray
`plot`	Whether to plot the incident ray, surface normal, and refracted ray

Details

The line argument provides the direction for the incident ray which is assumed to intersect with the normal at the normal's starting point. This is not checked and only the direction of the incident line matters (not its location or scale). The resulting refracted ray also starts at the normal's starting point.

Value

The refracted ray with the starting point being the normal's starting point (class Line)

See Also

Other optics: `Circle`; `IntersectLC`; `Line`; `Point`; `RefractLC`; `angle`; `center`; `length.Line`; `normalize`; `orthogonal`; `plot.Circle`; `plot.Line`; `plot.Point`; `rotate`; `scale`

rotate *Rotate a line about the starting point*

Description

Internal function for ray-tracing 2D optics

Usage

```
rotate(line, angle)
```

Arguments

`angle`	Angle to rotate (in radians)

Value

Object of Line class

See Also

Other optics: `Circle`; `IntersectLC`; `Line`; `Point`; `RefractLC`; `RefractLL`; `angle`; `center`; `length.Line`; `normalize`; `orthogonal`; `plot.Circle`; `plot.Line`; `plot.Point`; `scale`

`scale` *Scale a line by adjusting the end point*

Description

Internal function for ray-tracing 2D optics

Usage

```
scale(line, s)
```

Arguments

`s`	Scale

Value

Object of Line class

See Also

Other optics: `Circle`; `IntersectLC`; `Line`; `Point`; `RefractLC`; `RefractLL`; `angle`; `center`; `length.Line`; `normalize`; `orthogonal`; `plot.Circle`; `plot.Line`; `plot.Point`; `rotate`

`Shammas.Power` *Shammas Formula for IOL Power*

Description

Calculate IOL power for emmetropia given axial length, corenal curvature, and effective lens position.

Usage

```
Shammas.Power(L, K, ELP)
```

Arguments

`L`	axial length of the eye in millimeters (mm) from immersion ultrasound
`K`	corneal power (D)
`ELP`	IOL effective lens position (mm)

Details

Note: The axial length is assumed to be from immersion ultrasound. For contact ultrasound, add 0.24 mm.

Value

Power of IOL (D)

References

https://encrypted.google.com/books?id=NhWJsGFK6qgC&pg=PA8

See Also

`Power`

Other Power: `Binkhorst.Power`; `Binkhorst.adjusted.Power`; `Colenbrander.Power`; `Fyodorov.Power`; `Haigis.Power`; `Hoffer.Power`; `Hoffer.Q.Power`; `Holladay.1.Power`; `Holladay.Power`; `Power`; `SRK.II.Power`; `SRK.Power`; `SRK.T.Power`; `van.der.Heijde.Power`

`SRK.II.Power` *SRK-II Formula for IOL Power*

Description

Calculate IOL power for emmetropia given axial length, corenal curvature, and IOL A-constant.

Usage

```
SRK.II.Power(L, K, A, ELP)
```

Arguments

`L`	axial length of the eye in millimeters (mm)
`K`	corneal power (D)
`A`	IOL A-constant (D)
`ELP`	IOL effective lens position (mm) used to calculate equivalent A-constant

Value

Power of IOL (D)

See Also

Power

Other Power: Binkhorst.Power; Binkhorst.adjusted.Power; Colenbrander.Power; Fyodorov.Power; Haigis.Power; Hoffer.Power; Hoffer.Q.Power; Holladay.1.Power; Holladay.Power; Power; SRK.Power; SRK.T.Power; Shammas.Power; van.der.Heijde.Power

---

SRK.Power *SRK Formula for IOL Power*

Description

Calculate IOL power for emmetropia given axial length, corenal curvature, and IOL A-constant.

Usage

```
SRK.Power(L, K, A, ELP)
```

Arguments

L	axial length of the eye in millimeters (mm)
K	corneal power (D)
A	IOL A-constant (D)
ELP	IOL effective lens position (mm) used to calculate equivalent A-constant

Value

Power of IOL (D)

See Also

Power

Other Power: Binkhorst.Power; Binkhorst.adjusted.Power; Colenbrander.Power; Fyodorov.Power; Haigis.Power; Hoffer.Power; Hoffer.Q.Power; Holladay.1.Power; Holladay.Power; Power; SRK.II.Power; SRK.T.Power; Shammas.Power; van.der.Heijde.Power

---

SRK.T.ELP *SRK/T Formula for IOL Effective Lens Position*

Description

Calculate IOL effective lens position for emmetropia given axial length, corenal curvature, and IOL A-constant.

Usage

```
SRK.T.ELP(L, K, ACD, A)
```

Arguments

`L`	axial length of the eye in millimeters (mm)
`K`	corneal power (D)
`ACD`	IOL anterior chamber depth constant (mm)
`A`	IOL A-constant (D) used to calculate equivalent ACD

Details

Note: A warning is provided if the combination of corneal curvature and axial length produces an unexpected corneal height.

Value

Effective lens position of IOL (mm)

See Also

`ELP`

Other ELP: `ELP`; `Haigis.ELP`; `Hoffer.ELP`; `Hoffer.Q.ELP`; `Holladay.1.ELP`; `Holladay.ELP`

`SRK.T.Power` *SRK/T Formula for IOL Power*

Description

Calculate IOL power for emmetropia given axial length, corenal curvature, and effective lens position.

Usage

```
SRK.T.Power(L, K, ELP)
```

Arguments

`L`	axial length of the eye in millimeters (mm)
`K`	corneal power (D)
`ELP`	IOL effective lens position (mm)

Value

Power of IOL (D)

See Also

`Power`

Other Power: `Binkhorst.Power`; `Binkhorst.adjusted.Power`; `Colenbrander.Power`; `Fyodorov.Power`; `Haigis.Power`; `Hoffer.Power`; `Hoffer.Q.Power`; `Holladay.1.Power`; `Holladay.Power`; `Power`; `SRK.II.Power`; `SRK.Power`; `Shammas.Power`; `van.der.Heijde.Power`

`van.der.Heijde.Power` *van der Heijde Formula for IOL Power*

Description

Calculate IOL power for emmetropia given axial length, corenal curvature, and effective lens position

Usage

```
van.der.Heijde.Power(L, K, ELP)
```

Arguments

`L`	axial length of the eye in millimeters (mm)
`K`	corneal power (D)
`ELP`	IOL effective lens position (mm)

Value

Power of IOL (D)

See Also

`Power`

Other Power: `Binkhorst.Power`; `Binkhorst.adjusted.Power`; `Colenbrander.Power`; `Fyodorov.Power`; `Haigis.Power`; `Hoffer.Power`; `Hoffer.Q.Power`; `Holladay.1.Power`; `Holladay.Power`; `Power`; `SRK.II.Power`; `SRK.Power`; `SRK.T.Power`; `Shammas.Power`

Index