

Package: vfunc (via r-universe)

May 19, 2026

Type Package

Title Manipulate Virtual Functions

Version 1.0

Depends R (>= 4.2.0)

Suggests testthat, knitr, rmarkdown, onion

Maintainer Robin K. S. Hankin <hankin.robin@gmail.com>

Description If `f <- function(x){x^2}` and `g <- function(x){x+1}` it is a constant source of annoyance that ``f+g`` is not defined. Package 'vfunc' allows you to do this, and we have `(f+g)(2)` returning 6. The other arithmetic operators are similarly implemented. A wide class of coding bugs is eliminated.

License GPL-2

Imports methods

URL <https://github.com/RobinHankin/vfunc>,
<https://robinhankin.github.io/vfunc/>

BugReports <https://github.com/RobinHankin/vfunc/issues>

VignetteBuilder knitr

Repository <https://robinhankin.r-universe.dev>

Date/Publication 2026-03-20 08:27:34 UTC

RemoteUrl <https://github.com/robinhankin/vfunc>

RemoteRef HEAD

RemoteSha d85916701a1d19f4d42f75fed16ccba970d7bd0e

Contents

vfunc-package	2
as.vf	3
Compare-methods	4
Math	4

Math-methods	6
pow	6
vf-class	7
Index	9

vfunc-package	<i>Manipulate Virtual Functions</i>
---------------	-------------------------------------

Description

If `f <- function(x){x^2}` and `g <- function(x){x+1}` it is a constant source of annoyance to me that “`f+g`” is not defined. Package `vfunc` allows you to do this.

Details

The package defines a single **S4** class, `vf`. This has a single slot, `.Data`, of type `function` which means that `vf` objects inherit much of the behaviour of functions, but for which new methods (such as the `Arith` group of `S4` generics) may be defined.

Documentation Index

Index: This package was not yet installed at build time.

Author(s)

Robin K. S. Hankin [aut, cre] (ORCID: <<https://orcid.org/0000-0001-5982-0415>>)

Maintainer: Robin K. S. Hankin <hankin.robin@gmail.com>

Examples

```
f <- as.vf(function(x){x^2})
f + Sin
as.function(f*Sin + Exp)(1:4)
```

as.vf	<i>Coerce functions to a virtual function.</i>
-------	--

Description

Coerce objects to a virtual function. Numeric or complex arguments are coerced to a constant function.

Usage

```
as.vf(x)
```

Arguments

x	Generally, a function or numeric
---	----------------------------------

Value

Returns an object of class vf.

Note

It is rarely necessary to coerce objects such as vectors or matrices to class vf because the Arith methods operate on objects of class ANY directly.

Author(s)

Robin K. S. Hankin

Examples

```
as.vf(\(x)x^2)
Sin + as.vf(\(p){p^3})
```

```
as.vf(1:10)(1e99)
```

 Compare-methods

 Compare *methods for vf objects*

Description

Wouldn't it be nice to say $(f > g)(x)$ rather than the terrible, tedious and error-prone $f(x) > g(x)$? Well, now you can!

Methods

```
signature(e1 = "ANY", e2 = "vf")
signature(e1 = "function", e2 = "vf")
signature(e1 = "vf", e2 = "ANY")
signature(e1 = "vf", e2 = "function")
signature(e1 = "vf", e2 = "vf")
```

Examples

```
x <- seq(from=0, to=2*pi, len=100)
(Sin > Cos*Tan)(x)
```

 Math

*Math group generic functions in the **vfunc** package: trig, exponential, log, etc.*

Description

The S4 Math group contains 35 functions including `sin()`, `log()`, etc. The vfunc equivalents are capitalized, as in `Sin()`, `Log()`, etc.

Usage

```
Abs(x)
Sign(x)
Sqrt(x)
Ceiling(x)
Floor(x)
Trunc(x)
Cummax(x)
Cummin(x)
Cumprod(x)
Cumsum(x)
Log(x)
Log10(x)
```

Log2(x)
 Log1p(x)
 Acos(x)
 Acosh(x)
 Asin(x)
 Asinh(x)
 Atan(x)
 Atanh(x)
 Exp(x)
 Expm1(x)
 Cos(x)
 Cosh(x)
 Cosp(x)
 Sin(x)
 Sinh(x)
 Sinpi(x)
 Tan(x)
 Tanh(x)
 Tanpi(x)
 Gamma(x)
 Lgamma(x)
 Digamma(x)
 Trigamma(x)

Arguments

`x` Generally take a single argument of class `numeric`, `function`, or `vf`

Details

The reason for this rather untransparent device is that primitive functions such as `sin()` behave somewhat differently from other functions. We have:

```
Sin <- as.vf(function(x){sin(x)})
setMethod("sin", "vf", function(x){as.vf(function(o){Sin(x(o))})})
```

We define `Sin()` to be an object of class `vf`; the call to `setMethod()` ensures that `Sin(f)` operates as intended.

Value

Given a `numeric`, return a `numeric`; given a `vf`, return a `vf`

Note

Note that `sin <- as.vf(sin)` does not work as desired, giving a runtime error; trying to get round this with things like `sin <- as.vf(function(x)sin)` and similar means that `sin(3)` does not work.

There is no way to inform all `vf` objects that, if used as a function with an argument of a primitive such as `sin`, to return another `vf` object—and not to try and evaluate `f(sin)`, which fails:

```
f <- as.vf(function(x){x^2 + 1})
f(Sin)
#> An object of class "vf"
#> function (...)
#> {
#>   e1(...) + e2
#> }
#> <bytecode: 0x6065e7c8a900>
#> <environment: 0x6065e7c8a548>
f(sin)
#> Error in x^2: non-numeric argument to binary operator
```

Above, we see `f(sin)` returning an error (it tries to evaluate “`sin^2 + 1`”). Observe that “`Sin^2 + 1`” is perfectly OK, for `Sin` is a virtual function.

Author(s)

Robin K. S. Hankin

Examples

`Sin + Exp`

```
c((Sin + Exp)(.02232) ,sin(0.02232) + exp(0.02232))
```

Math-methods

Methods for Function Math, Arith in the vfunc package

Description

Various S4 methods to work with vf objects. Comparison methods are documented at `Compare-methods`.

pow

Iterated functions; functional powers

Description

Given a function $f: X \rightarrow X$, we define

$$f^0 = \text{id}_X$$

$$f^{n+1} = f \circ f^n = f^n \circ f, \quad n \geq 0$$

This gives us $f^{n+m} = f^n \circ f^m$ and $(f^m)^n = f^{mn}$, which motivates the notation. For example, $\sin^3 = \sin \circ \sin \circ \sin$, so $\sin^3(x) = \sin(\sin(\sin x))$.

The operator is well-defined due to the power associativity of function composition.

Usage

```
pow(x, n)
```

Arguments

x	Object of class vf
n	Non-negative integer

Value

Returns an object of class vf

Note

There are possibly more efficient methods requiring fewer compositions, e.g. `pow(f, 9)` (which would require 8 function compositions) could be evaluated by `pow(pow(f, 3), 3)` (which requires only four). But I am not sure that this would actually be any faster, and I have not got round to thinking about it yet.

Also, package idiom for the caret “^” is reserved for arithmetic exponentiation [so, for example, $(f^3)(x) == f(x)*f(x)*f(x)$]. I believe this is sub-optimal but was unable to overload the caret to implement functional iteration.

Author(s)

Robin K. S. Hankin

Examples

```
pow(Sin, 5)
Sin^5

f <- as.vf(function(x){x^2+1})

pow(f+Sin, 4)
pow(f+Sin, 4)(2)
```

 vf-class

 Class "vf"

Description

Class vf stands for “virtual function”

Objects from the Class

Objects can be created by calls of the form `new("vf", ...)`.

Slots

.Data: Object of class "function"

Methods

Arith signature(e1 = "function", e2 = "vf"): ...
Arith signature(e1 = "ANY", e2 = "vf"): ...
Arith signature(e1 = "vf", e2 = "function"): ...
Arith signature(e1 = "vf", e2 = "missing"): ...
Arith signature(e1 = "vf", e2 = "ANY"): ...
Arith signature(e1 = "vf", e2 = "vf"): ...
as.function signature(x = "vf"): ...
as.vf signature(x = "vf"): ...
coerce signature(from = "function", to = "vf"): ...
coerce signature(from = "ANY", to = "vf"): ...
coerce signature(from = "vf", to = "function"): ...
Compare signature(e1 = "function", e2 = "vf"): ...
Compare signature(e1 = "ANY", e2 = "vf"): ...
Compare signature(e1 = "vf", e2 = "function"): ...
Compare signature(e1 = "vf", e2 = "ANY"): ...
Compare signature(e1 = "vf", e2 = "vf"): ...
Math signature(x = "vf"): ...

Author(s)

Robin K. S. Hankin

Examples

```
showClass("vf")
```

Index

- * **classes**
 - vf-class, 7
- * **methods**
 - Compare-methods, 4
 - Math-methods, 6
- * **package**
 - vfunc-package, 2

- Abs (Math), 4
- abs, vf-method (Math), 4
- Acos (Math), 4
- acos, vf-method (Math), 4
- Acosh (Math), 4
- acosh, vf-method (Math), 4
- Arith, ANY, vf-method (Math-methods), 6
- Arith, function, vf-method (Math-methods), 6
- Arith, vf, ANY-method (Math-methods), 6
- Arith, vf, function-method (Math-methods), 6
- Arith, vf, missing-method (Math-methods), 6
- Arith, vf, vf-method (Math-methods), 6
- Arith-methods (Math-methods), 6
- as.function, vf-method (vf-class), 7
- as.vf, 3
- as.vf, ANY-method (as.vf), 3
- as.vf, function-method (as.vf), 3
- as.vf, vf-method (vf-class), 7
- Asin (Math), 4
- asin, vf-method (Math), 4
- Asinh (Math), 4
- asinh, vf-method (Math), 4
- Atan (Math), 4
- atan, vf-method (Math), 4
- Atanh (Math), 4
- atanh, vf-method (Math), 4

- Ceiling (Math), 4
- ceiling, vf-method (Math), 4

- coerce, ANY, vf-method (vf-class), 7
- coerce, function, vf-method (vf-class), 7
- coerce, vf, function-method (vf-class), 7
- Compare, ANY, vf-method (Compare-methods), 4
- Compare, function, vf-method (Compare-methods), 4
- Compare, vf, ANY-method (Compare-methods), 4
- Compare, vf, function-method (Compare-methods), 4
- Compare, vf, vf-method (Compare-methods), 4
- Compare-methods, 4
- Cos (Math), 4
- cos, vf-method (Math), 4
- Cosh (Math), 4
- cosh, vf-method (Math), 4
- Cospi (Math), 4
- cospi, vf-method (Math), 4
- Cummax (Math), 4
- cummax, vf-method (Math), 4
- Cummin (Math), 4
- cummin, vf-method (Math), 4
- Cumprod (Math), 4
- cumprod, vf-method (Math), 4
- Cumsum (Math), 4
- cumsum, vf-method (Math), 4

- Digamma (Math), 4
- digamma, vf-method (Math), 4

- Exp (Math), 4
- exp, vf-method (Math), 4
- Expn1 (Math), 4
- expn1, vf-method (Math), 4

- Floor (Math), 4
- floor, vf-method (Math), 4

- Gamma (Math), 4

gamma, vf-method (Math), 4

Lgamma (Math), 4
lgamma, vf-method (Math), 4
Log (Math), 4
log, vf-method (Math), 4
Log10 (Math), 4
log10, vf-method (Math), 4
Log1p (Math), 4
log1p, vf-method (Math), 4
Log2 (Math), 4
log2, vf-method (Math), 4

Math, 4
Math, vf-method (Math-methods), 6
Math-methods, 6

pow, 6
power (pow), 6

Sign (Math), 4
sign, vf-method (Math), 4
Sin (Math), 4
sin, vf-method (Math), 4
Sinh (Math), 4
sinh, vf-method (Math), 4
Sinpi (Math), 4
sinpi, vf-method (Math), 4
Sqrt (Math), 4
sqrt, vf-method (Math), 4

Tan (Math), 4
tan, vf-method (Math), 4
Tanh (Math), 4
tanh, vf-method (Math), 4
Tanpi (Math), 4
tanpi, vf-method (Math), 4
Trigamma (Math), 4
trigamma, vf-method (Math), 4
Trunc (Math), 4
trunc, vf-method (Math), 4

vf-class, 7
vfunc (vfunc-package), 2
vfunc-package, 2