

# VarTable

VarTable is a package to make variation table, in a simple way  
This package is build on top of [fletcher](#)  
(version: 0.1.0)

## Contents

1 - Introduction .....	1
2 - tabvar function .....	2
2.1 - general description .....	2
tabvar .....	2
Parameters .....	2
init .....	2
domain .....	2
arrow .....	3
stroke .....	3
stroke-arrow .....	3
marks-line .....	3
contents .....	3
2.2 - The contents parameter .....	4
2.2.1 - Sign rows format .....	4
2.2.1.1 - A classical sign array .....	4
2.2.1.2 - Custom separation bar .....	5
2.2.1.2.1 style of bar .....	5
2.2.1.2.2 type of bar .....	5
2.2.1.3 - Same sign for more than one value of the variable .....	6
2.2.2 - Variation table .....	6
2.2.2.1 - A classical variation array .....	7
2.2.2.2 - Undefined values .....	7
2.2.2.3 - Skip a value .....	9
3 - More complex examples .....	10
3.1 - $\Gamma$ function on $[0; +\infty]$ .....	10
3.2 - A Rational function .....	11
3.3 Hyperbolic function .....	12
3.3 A weird table for a simple polynomial function .....	13

## 1 - Introduction

This package is designed to simplify the creation of variation tables for functions. To do this, it gives you a typst function, whose parameters are described in detail in this documentation.

**A word of warning:** it's quite normal that during the array creation process, The elements, such as the lines between the various elements, aren't created as they should be. For example, the line between the labels and the rest, which doesn't go all the way to the end.

If you encounter any bugs, please report them on my [GitHub](#).

## 2 - tabvar function

### 2.1 - general description

#### tabvar

Render a variation table and sign table of your functions

#### Parameters

```
tabvar(  
  init: dictionary,  
  domain: array,  
  arrow: string,  
  stroke: lenght color gradient,  
  stroke-arrow: lenght color gradient,  
  marks-line: string,  
  contents: array  
)
```

**init** dictionary

initialitation of the table

- variable is a content block which contains the table's variable name (like  $x$  or  $t$ )
- label is an array of 2 arguments that contains in first position the line's label and in second position, if the line is a variation table or a sign table with this following keys : "Variation" and "Sign"

**Example:** for a variation table of a function  $f$ , you should write:

```
init: (  
  variable:  $x$ ,  
  label: (  
    ([sign of  $f$ ], "Sign"), // the first line is a sign table  
    ([variation of  $f$ ], "Variation") // the second line is a variation table  
  )  
)  
  
Default: (  
  "variable": [],  
  "label": [],  
)
```

**domain** array

values taken by the variable

for example if your funtions changes sign or reaches a max/min for  $x \in \{0, 1, 2, 3\}$   
you should write this :

domain: ( $0$ ,  $1$ ,  $2$ ,  $3$ )

Default: ()

**arrow**    string

**Optional**

The arrow's style

you can use all different kind of "string" arrow of the fletcher package, so I invite you to read the [fletcher documentation](#)

Default: "->"

**stroke**    lenght or color or gradient

**Optional**

The table's color and thickness

**Caution** : this stroke can take only lenght, color or gradient types but none of the others

Default: 1pt + black

**stroke-arrow**    lenght or color or gradient

**Optional**

the arrow's color and thickness

**Caution** : this stroke can take only lenght, color or gradient types but none of the others

Default: 0.6pt + black

**marks-line**    string

**Optional**

if you want to change the style of all separator lines between signs

Warning: this will only change the default lines, the ||, | or 0 lines will not be changed.

Default: "- "

**contents**    array

the content of the table

see below for more details

Default: ( ( ) , )

## 2.2 - The contents parameter

The contents parameter must be an array with one element per line (per label)

Each element is itself an array with one element per column, with a different format for either sign or variation rows

### 2.2.1 - Sign rows format

Should contain as much element as the domain less one (one per interval) + one optional end bar style element

Each element is in either of these form (can be mixed on a same line):

() – Empty : extend previous cell

body – Simple body such as  $++$  or  $--$

(body, bar\_style) – to specify an optional style for the **previous** bar, with one of "|" (simple bar),

"||" (double bar) or "0" (bar with a zero)

NB: the line-0 parameter change the default bar style to "|"

The optional last element is "||"

#### 2.2.1.1 - A classical sign array

A sign array must contain contents like  $++$  or  $--$ , but you can put anything else.

#### Example:

A normal sign table:

```
#tabvar(  
  init: (  
    variable: $t$,  
    label: ([[sign], "Sign"),  
  ),  
  domain: ($2$, $4$, $6$, $8$),  
  contents: (($+$, $-$, $+$)),  
)
```

$t$	2	4	6	8
sign	+	–	+	

More complex usage:

```
#tabvar(  
  init: (  
    variable: $t$,  
    label: ([[sign], "Sign"),  
  ),  
  domain: ($2$, $4$, $6$, $8$),  
  contents: (  
    (  
      "Hello world !",  
      $-$,  
      $3/2$  
    ),  
  ),  
)
```

$t$	2	4	6	8
sign	Hello world!	–	$\frac{3}{2}$	

**Note:** on the second example the table is squeezed with the scale function)

### 2.2.1.2 - Custom separation bar

#### 2.2.1.2.1 style of bar

you can modify the style of the bars (note that this modifies all the default ones, not the others, see 2.2.1.2.2).

the bar style is passed as a string, like “-” for a plain bar or “.” for dotted lines, I invite you to read [fletcher’s documentation](#) on marks to find out what can be done.

#### Example

```
#tabvar(  
  marks-line: "--",  
  init: (  
    variable: $t$,  
    label: ([sign], "Sign"),  
  ),  
  domain: ($2$, $4$, $6$,),  
  contents: (  
    ($+$, $-$),  
  ),  
)
```

$t$	2	4	6
sign	+	—	

PS: yes, technically these bars are arrows for fletcher but shhh, keep it to yourself to make tables that don’t make sense.

#### 2.2.1.2.2 type of bar

For all signs except the first one, instead of putting the sign directly, you can put a couple, whose first element defines the previous bar’s type.

There are 3 different types of bar:

- “|” : a simple bar
- “0” : a bar with a 0 on the center
- “||” : a double bar, like for the undefined values

**NB** : the marks-line parameter has no effect on these bars

#### Example :

```
#tabvar(  
  init: (  
    variable: $t$,  
    label: ([sign], "Sign"),  
  ),  
  domain: ($2$, $4$, $6$, $8$, $10$,),  
  contents: (  
    (  
      $+$,  
      ("|", $-$),  
      ("0", $-$),  
      ("||", $+$)  
    ),  
  ),  
)
```

$t$	2	4	6	8	10
sign	+	—	0	—	+

**Note** : The line-0 parameter is to default lines to “0” type or “|” type

If you want a double line at the start of the table, you can use a double bar "||" on the very first sign. If you want it at the end, you can add this element "||" at the end of sign array

**Example :**

```
#tabvar(
  init: (
    variable: $t$,
    label: ([sign], "Sign"),
  ),
  domain: ($2$, $4$, $6$, $8$),
  contents: (
    (
      ("||", $+$),
      $-$,
      "||"
    ),
  ),
)
```

$t$	2	4	6
sign	+	—	

### 2.2.1.3 - Same sign for more than one value of the variable

For this, it is pretty easy, instead of putting the sign directly, you can put an empty couple

**Example :**

```
#tabvar(
  init: (
    variable: $t$,
    label: ([sign], "Sign"),
  ),
  domain: ($2$, $4$, $6$, $8$),
  contents: (
    (
      $+$,
      (),
      $-$
    ),
  ),
)
```

$t$	2	4	6	8
sign	+			—

### 2.2.2 - Variation table

Should contains as much elements as the domain

Each element is in either of these forms :

- () to extend the previous arrow
- (position,body) with position being one of top, center or bottom
- (pos1, pos2, "||", body1, body2) to put in 2 value separated by an undefined value (double bar)
- (pos, "||", body) shorthand for (pos, pos, "||", body, body) (see previous format)

### 2.2.2.1 - A classical variation array

A variation array must contain couple with in first position, the element position, and in second position, whatever you want as long as it's of the contents type.

The position can be top, center or bottom, but no other type of alignment

**Example:**

```
#tabvar(
  init: (
    variable: $t$,
    label: ([variation], "Variation"),),
  ),
  domain: ($2$, $4$, $6$, $8$),
  contents: (
    (
      (top, $3$),
      (bottom, $0$),
      (center, $1$),
      (top, $4$),
    ),
  ),
)
```

$t$	2	4	6	8
variation				

### 2.2.2.2 - Undefined values

If your function is not defined on some values like  $f(x) = \frac{1}{x}$  for  $x = 0$ , you certainly want to put a double line meaning that the function is undefined on this value, and you can !

★ For each values of domain except the start and the end.

The array of one value should look like (pos1, pos2, "||", contents1, contents2)

where:

- pos1 and 2 is top, center, bottom and pos1 is for the placement of contents1 similary for pos2
- "||" is to precise the value is undefine
- contents1 and 2 is type of contents and contents1 one is for before the double bar and contents2 for after

**Example:**

```
#tabvar(
  init: (
    variable: $t$,
    label: ([variation], "Variation"),),
  ),
  domain: ($2$, $4$, $6$, $8$),
  contents: (
    (
      (top, $3$),
      (bottom, top, "||", $0$, $2$),
      (bottom, $1$),
    ),
  ),
)
```

$t$	2	4	6
variation			

If pos1 and pos2 is same, you can just fill in one instead of two,  
 In the same way if contents1 and contents2 is same, you can also enter just one

### Example:

Instead of (top, top, "||" , \$0\$, \$0\$) you can use (top, "||" , \$0\$)

```
#tabvar(
  init: (
    variable: $t$,
    label: ([variation], "Variation"),),
  ),
  domain: ($2$, $4$, $6$, $8$),
  contents: (
    (
      (top, $3$),
      (bottom, "||", $0$, $1$),
      (top, center, "||", $2$),
      (top, "||", $3$),
      (bottom, $1$),
    ),
  ),
)
```

$t$	2	4	6	8
variation				

★ For the first and the end values

It a basic array but with "||" this parameter at the array's center

For example (top, "||", \$3\$)

```
#tabvar(
  init: (
    variable: $t$,
    label: ([variation], "Variation"),),
  ),
  domain: ($2$, $4$, $6$, $8$),
  contents: (
    (
      (top, "||", $3$),
      (bottom, $1$),
      (top, "||", $2$),
    ),
  ),
)
```

$t$	2	4	6
variation			



### 2.2.2.3 - Skip a value

When you want to use several functions in the same table, you will probably want to skip some values, to do this, as with sign arrays, you must create an empty array

**Example:**

```
#tabvar(  
  init: (  
    variable: $t$,  
    label: ([variation], "Variation"),  
  ),  
  domain: ($2$, $4$, $6$, $8$),  
  contents: (  
    (  
      (top, "||", $3$),  
      (),  
      (bottom, $2$),  
    ),  
  ),  
)
```

$t$	2	4	6
variation			

### 3 - More complex examples

Here is a little bundle of what you can do with the package.

#### 3.1 - $\Gamma$ function on $[0; +\infty]$

Where it takes a minimum on  $[0; +\infty[$  for  $x = \alpha$

Code:

```
#tabvar(  
  init: (  
    variable:  $t$ ,  
    label: (  
      ([sign of  $\Gamma$ ], "Sign"),  
      ([variation of  $\Gamma$ ], "Variation"),  
    ),  
  ),  
  domain: ( $0$ ,  $\alpha$ ,  $+\infty$ ),  
  contents: (  
    ( $-$ ,  $+$ ),  
    (  
      (top, " $||$ ",  $+\infty$ ),  
      (bottom,  $\Gamma(\alpha)$ ),  
      (top,  $+\infty$ ),  
    ),  
  ),  
)
```

Result:

$t$	$0$	$\alpha$	$+\infty$
sign of $\Gamma$	$-$	$+$	
variation of $\Gamma$			

### 3.2 - A Rational function

Take  $f(x) = \frac{4x^2+12x+29}{4(x^2+3x+2)}$

So we have  $f'(x) = \frac{-2x-3}{16(x^2+3x+2)^2}$

And finally, we get:

Code:

```
#tabvar(
  init: (
    variable: $t$,
    label: (
      ([sign of $f'$], "Sign"),
      ([variation of $f$], "Variation"),
    ),
  ),
  domain: ($ -\infty $, $ -2 $, $ -3 / 2 $, $ -1 $, $ +\infty $),
  contents: (
    ($+$, ("||", $+$), $-$, ("||", $-$)),
    (
      (bottom, $1$),
      (top, bottom, "||", $+\infty$, $-\infty$),
      (top, $-20$),
      (bottom, top, "||", $-\infty$, $+\infty$),
      (bottom, $1$),
    ),
  ),
)
```

Result:

$t$	$-\infty$	$-2$	$-\frac{3}{2}$	$-1$	$+\infty$
sign of $f'$	+	+	-	-	
variation of $f$					

### 3.3 Hyperbolic function

Code:

```
#tabvar(
  arrow: "|-harpoon",
  stroke-arrow:
gradient.linear(..

```

Result:

$t$	$-\infty$	$0$	$+\infty$
sign of cosh	—		+
variation of cosh			
sign of sinh and tanh	+		
variation of sinh			
variation of tanh			

### 3.3 A weird table for a simple polynomial function

Take  $g(t) = t^2 - t^3$

So, we have  $g'(t) = 2t - 3t^2$

And has local extrema for  $x = 0$  and  $x = \frac{2}{3}$

Code:

```
#tabvar(

stroke: 5pt + red,
arrow: "X-*-<>",
stroke-arrow: purple + 1.4pt,
marks-line: "<-->",
init: (
  variable: $t$,
  label: (
    ([sign of $g'$], "Sign"),
    ([variation of $g$], "Variation"),
  ),
),
domain: ($ -\infty $, $ 0 $, $ 2 / 3 $, $ +\infty $),
contents: (
  ($-$, ("|", $+$), $-$),
  (
    (top, $+\infty$),
    (bottom, $0$),
    (center, $ 4 / 27 $),
    (bottom, $-\infty$),
  ),
),
),
)
```

Result:

$t$	$-\infty$	$0$	$\frac{2}{3}$	$+\infty$
sign of $g'$	—	+	—	
variation of $g$				