

travail Rsympy

vincent obertelli

12/7/2020

Nous allons maintenant aborder les fonctions mathématiques de RStudio. Nous allons voir la fonction `rSymPy`. Cette fonction sert à traiter les nombres pairs et impairs, les nombres réels, les complexes, les nombres premiers et l'infini. Pour commencer, nous allons installer la fonction `rSymPy` : `install.packages("rSymPy")`

Il est aussi nécessaire de télécharger Jython et rJava pour utiliser toutes les fonctions de `rSymPy` : `install.packages("rJython")`

`install.packages("rJava")`

Maintenant, nous allons voir certaines possibilités de base de cette librairie :

```
library(rSymPy)
```

```
## Loading required package: rJython
```

```
## Loading required package: rJava
```

```
## Loading required package: rjson
```

```
# créer une variable sympy qu'on appelle x  
sympy("var('x')")
```

```
## [1] "x"
```

```
#mettre x dans une variable y, qui est le carré de x  
sympy("y = x*x")
```

```
## [1] "x**2"
```

```
#afficher y  
sympy("y")
```

```
## [1] "x**2"
```

Ce que nous venons de voir permet de créer une variable `x` et la manipuler avec des commandes simples. C'est la base de fonctions et calculs plus complexes que nous allons voir maintenant.

```
library(rSymPy)
#calcul de la limite
sympy("limit(1/x, x, oo)")
```

```
## [1] "0"
```

```
#calcul utilisant des sinus
sympy("diff(sin(2*x), x, 1)")
```

```
## [1] "2*cos(2*x)"
```

```
sympy("diff(sin(2*x), x, 2)")
```

```
## [1] "-4*sin(2*x)"
```

```
#intégration de la fonction
sympy("integrate(exp(-x), (x, 0, oo))")
```

```
## [1] "1"
```

Dans cette partie, nous avons vu ce que pouvez nous proposer la fonction rSymPy. Il nous est possible d'appliquer des formules mathématiques plus complexes, mais pour le moment, je vais essayer tout d'abord de bien maîtriser les formules mathématiques de bases, pour ensuite approfondir ma connaissance de rSymPy.