

Docopt, UseR!2014,

Edwin de Jonge

July 2014

- R is used more and more
- So also for executing R-scripts

Typical usage

```
#!/usr/bin/Rscript  
my_model <- lm(data=iris, Sepal.Width ~ Sepal.Length)  
print(coef(my_model))
```

Typical usage

```
#!/usr/bin/Rscript

my_data <- iris
dep <- "Sepal.Width"
indep <- "Sepal.Length"

f <- as.formula(paste0(dep, " ~ ", indep))
my_model <- lm(data=my_data, f)
print(coef(my_model))
```

- Work with `commandArgs`
- Use `getopt` (primitive)
- `argparse`, Python dependency
- **`optparse`** quite nice, Python inspired
- `docopt`

What is Docopt

- Command Line Interface spec
- You specify the help and the parser takes care of everything.
- The documentation is the specification.
- Originated as Python lib (<http://docopt.org>) test

- Parse positional arguments
- Default values
- Descriptions
- Optional parameter
- Commands
- Ellipses
- Short/Long option parsing
- Self describing

- Basic implementation

- Docopt is implemented using Reference classes (R5) in pure R.
- It is kind-of-port of the original Python project:
`http://docopt.org`
- Available from: `https://github.com/edwindj/docopt.R`
- Very functional, except for:
- multiple identical arguments -vvv
- multiple prog statements.

Loose ends

Thanks

Thanks!