

# drake

reproducible workflow management in R

Dominik Rafacz

18.05.2019

# drake package



roperenci/drake

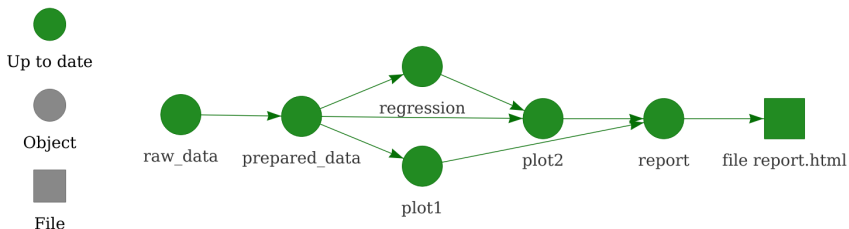
Will Landau

## drake plan

```
plan <- drake_plan(  
  raw_data      = read.csv(file_in("apartments.csv")),  
  prepared_data = prepare(raw_data),  
  plot1         = make_plot1(prepared_data),  
  regression    = lm(m2.price ~ ., prepared_data),  
  plot2         = make_plot2(prepared_data, regression),  
  report        = render(  
    knitr_in("report.Rmd"),  
    output_file = file_out("report.html")  
  )  
)  
make(plan)  
readd(plot1)
```

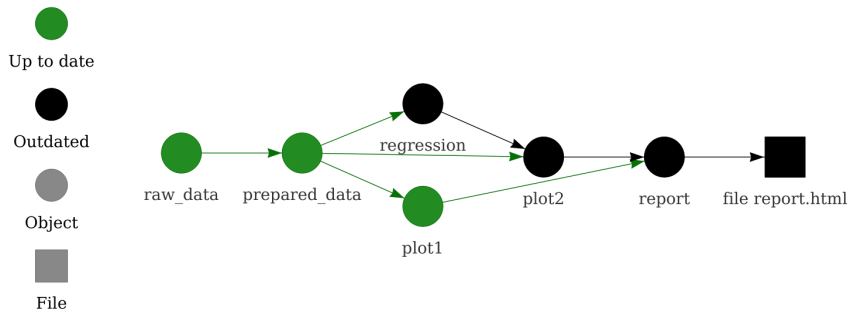
# visualizing plan

```
config <- drake_config(plan)  
vis_drake_graph(config)
```



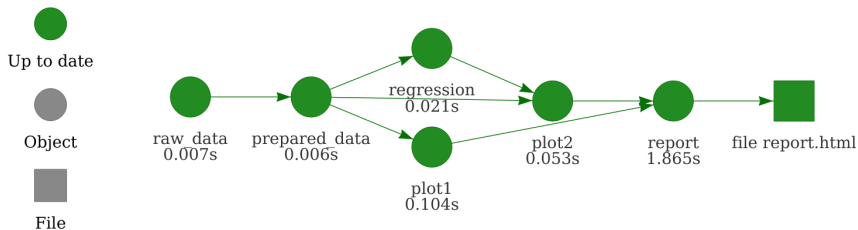
# visualizing plan

```
config <- drake_config(plan)
vis_drake_graph(config)
```



# visualizing plan

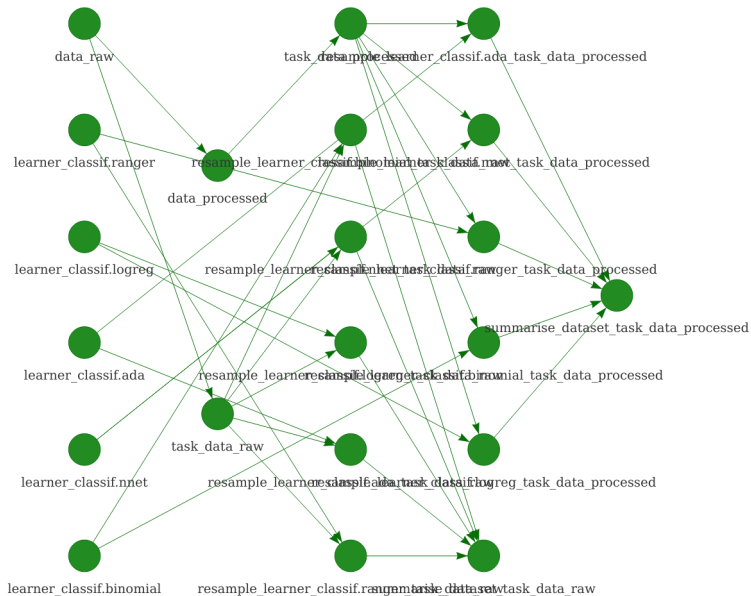
```
config <- drake_config(plan)
vis_drake_graph(config)
```



# parallelization

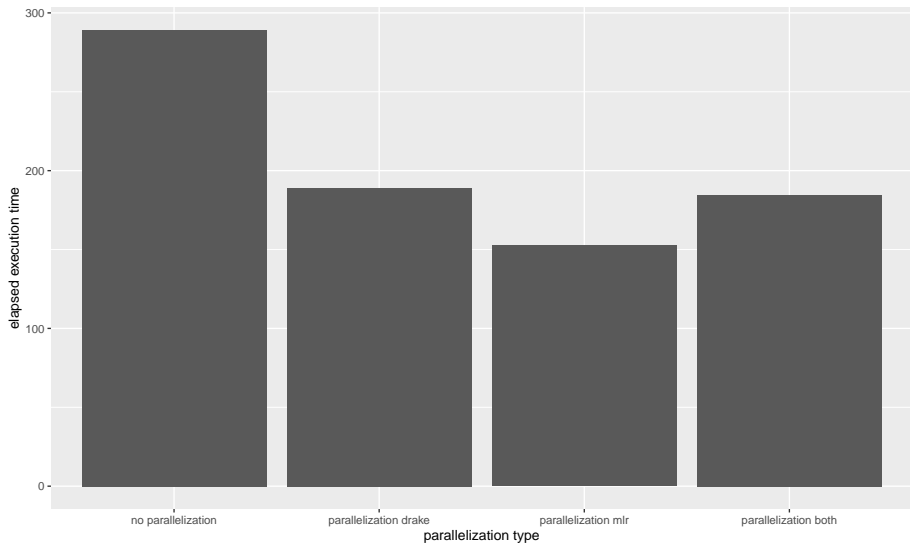
```
plan <- drake_plan(  
  data_raw = read.csv(file_in("phplE7q6h.csv")),  
  data_processed = normalizeFeatures(data_raw, target = "class"),  
  task = target(  
    makeClassifTask("task", data, "class"),  
    transform = map(data = c(data_raw, data_processed))  
  ),  
  learner = target(  
    makeLearner(model),  
    transform = map(model = c("classif.ranger",  
                              "classif.logreg",  
                              "classif.ada",  
                              "classif.nnet",  
                              "classif.binomial"))  
  ),  
  resample = target(  
    resample(learner, task, cv10),  
    transform = cross(learner, task)  
  ),  
  summarise_dataset = target(  
    summary(resample),  
    transform = combine(resample, .by = task)  
  )  
)
```

# parallelization





# parallelization



# summary

- reproducibility

# summary

- reproducibility
- code management

# summary

- reproducibility
- code management
- parallelization

the end

Thanks for your attention

**Acknowledgements:**

Michał Burdukiewicz (Warsaw University of Technology)