R Optimization Survey

Seth Russell

8/7/2018

```
## Loading required package: xml2
```

```
## Loading required package: future
```

```
##
## Attaching package: 'future.apply'
```

```
## The following object is masked from 'package:future':
##
##     future_lapply
```

Introduction

This notebook performs an analysis of use of optimization related packages and methodologies for all R packages avaialble on CRAN.

Two methods are used and results are compared.

Method 1: Grep for non-empty src directories. By convention, packages using compiled code (C, C++, Fortran) place files in the '/src' directory.

Method 2: Check for stated dependencies on packages.

Potential problems/limitations:

The best evidence of optimization would be in history of commits and unit test runs. While all packages have source code available, not all have development history available nor unit tests available.

Use of one of these specified packages doensn't mean user is trying to optimize. Package dependencies can be misleading - I've seen CRAN packages that list packages as dependencies and by inspection of their source code, they never use it.

Grepping for non-empty src directory

liquidSVM is a good example of an optimized and tested package - see https://arxiv.org/pdf/1702.06899.pdf * has no optimization related dependencies (so wont show up in depedency based section) * core of package implemented in C++

Iso package has Fortran code that pre-dates 2013/ * Does it use Fortran code for performance reasons, ease of implementation, legacy reasons, or ???

From manual review, it seems that all C++ and Fortran code is in src directory. External/third party libraries are usually included in other directories.

One potential problem is use of Java - while it seems that due to how rJava allows Java code to be called from R, a memory perfomance hit may occur - but some specific packages do mention that using Java threading improves performance - see package 'rmcfs' as an example

At this point, just look for non empty src directory…

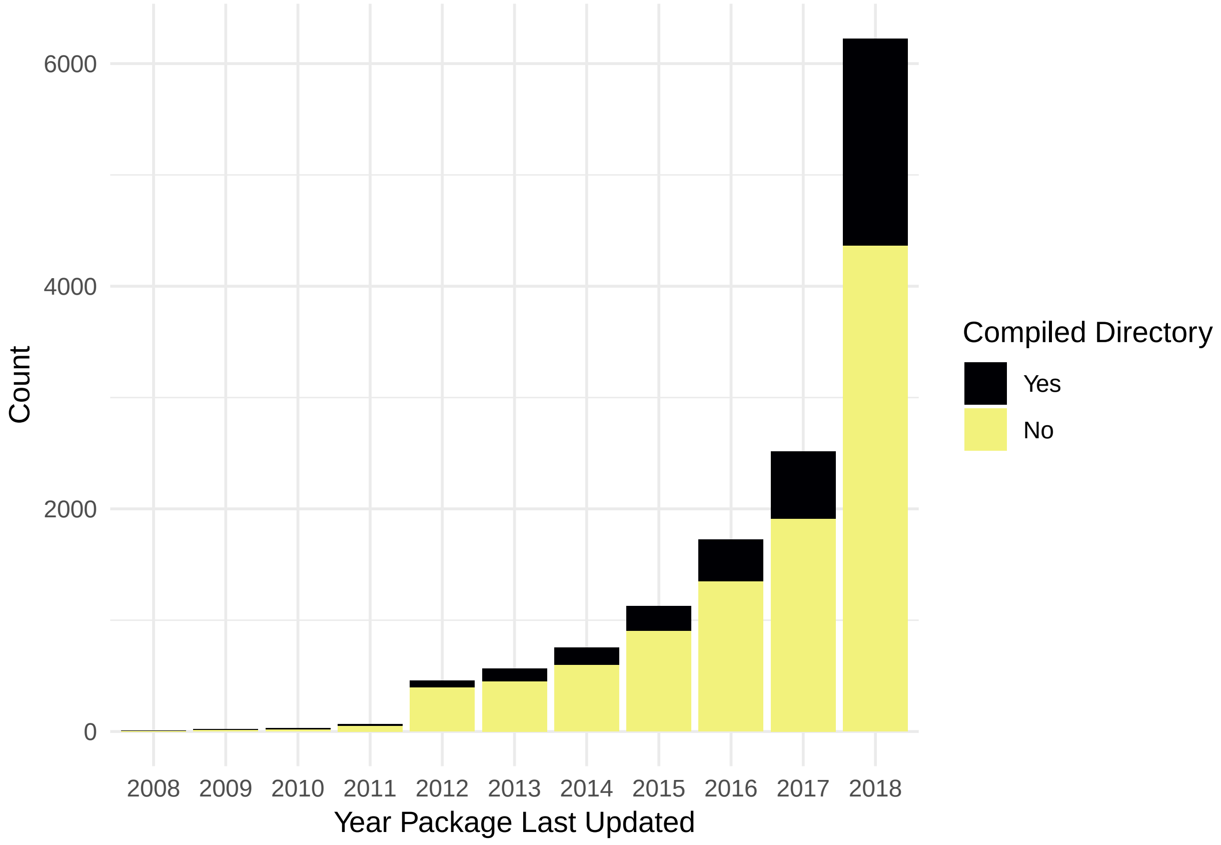
```
# if value 100%, then read all packages, otherwise, randomly select number of packages provided
sample_size <- '100%'
# dont need to untar, but is useful for manual analysis
untar_files <- FALSE

# call code from shared_fn.R to download files and extract them if desired
package_df <- initialize(search_pattern="src[^/]*/.+", sample_size=sample_size, untar_files=untar_files

if (nrow(package_df[package_df$download_error == TRUE, ]) > 0) {
  print('Unable to download or untar these packages - they will not be considered in analysis')
  print(package_df[package_df$download_error == TRUE, ])
} else {
  print('It appears that all files downloaded and untared successfully.')
}

print(paste('As of', date(), 'there are', nrow(package_df), 'packages available on CRAN'))
```

Visualizations and Tables for Grep results

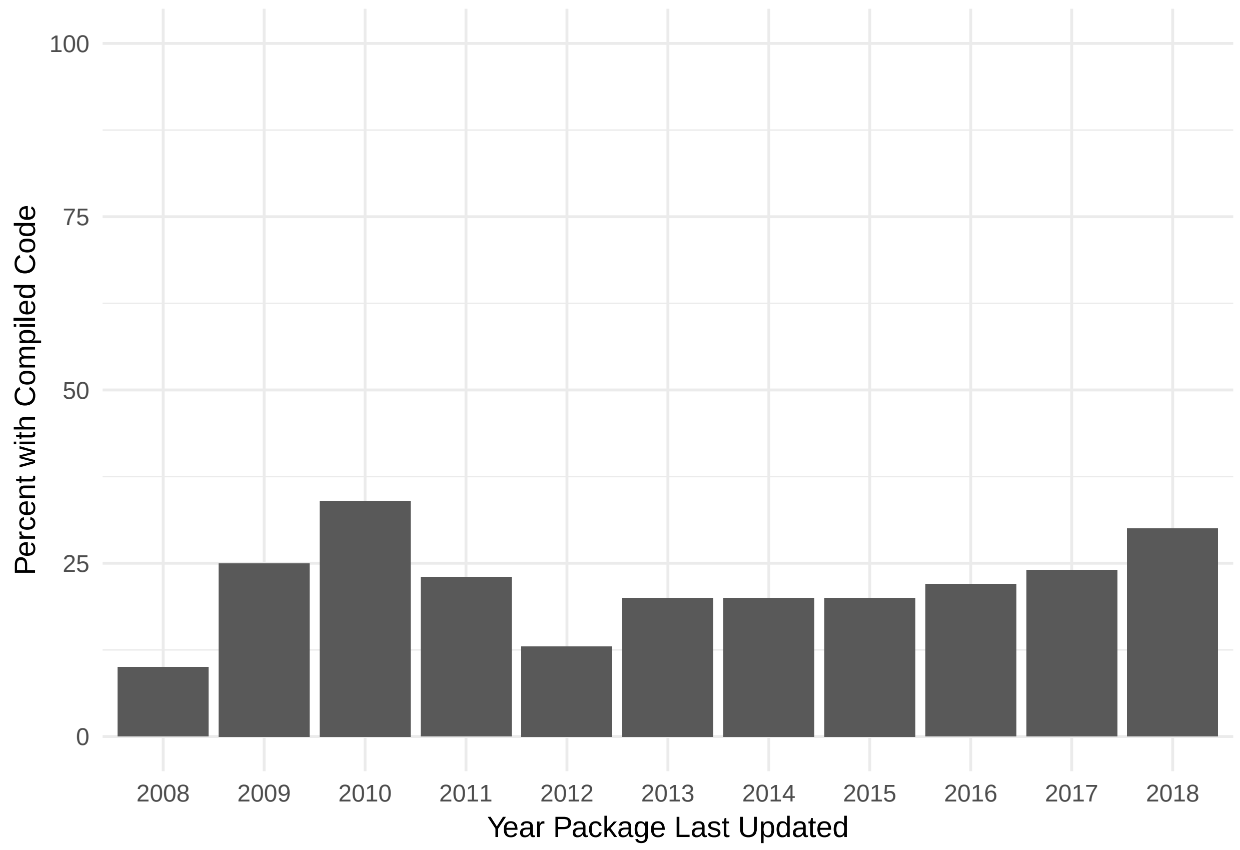
```
# build data needed for plots
gtf <- grep_table_freqs(package_df)
gtt <- grep_table_totals(gtf)
```

```
## [1] "Summary Table for Grep Results"
##       [,1]   [,2]   [,3]   [,4]   [,5]   [,6]   [,7]   [,8]   [,9]
## Year  "2005" "2006" "2007" "2008" "2009" "2010" "2011" "2012" "2013"
## Total "   1" "   4" "   1" "  10" "  24" "  32" "  65" " 457" " 564"
## Grep  NA     "   3" NA     "   1" "   6" "  11" "  15" "  59" " 114"
## Pct   NA     "75"   NA     "10"   "25"   "34"   "23"   "13"   "20"
##       [,10]  [,11]  [,12]  [,13]  [,14]
## Year  "2014" "2015" "2016" "2017" "2018"
## Total " 755" "1127" "1726" "2517" "6226"
## Grep  " 154" " 224" " 376" " 606" "1859"
## Pct   "20"   "20"   "22"   "24"   "30"
```

```
# generate plots
freq_plot <- grep_viz_freq(gtf, image_prefix = 'optimization', label = 'Compiled')
freq_plot
```



```
pct_plot <- grep_viz_pct(gtt, image_prefix = 'optimization', label = 'Compiled')
pct_plot
```



Package Dependencies

```
# function from shared_fn.R - sets package_df$package_deps
package_df <- calc_dependencies(package_df)
```

```
##           Rcpp       parallel        foreach     doParallel microbenchmark
##           1528            941            507            390             83
##         future   RcppParallel           doMC           Rmpi     rbenchmark
##             39             35             26             23             21
##           snow         doSNOW   future.apply    parallelMap       sparklyr
##             21             16             16             14             11
##         tictoc   parallelDist          doMPI     batchtools          profr
##              5              5              5              3              2
##         benchr        profvis         SparkR       partools            DSL
##              1              1              1              1              1
##          bench
##              0
```

```
# are there any that have more than one dependency listed?
# these are all counted multiple times in histogram
show_multiple_dependencies(package_df, look_for_packages)
```

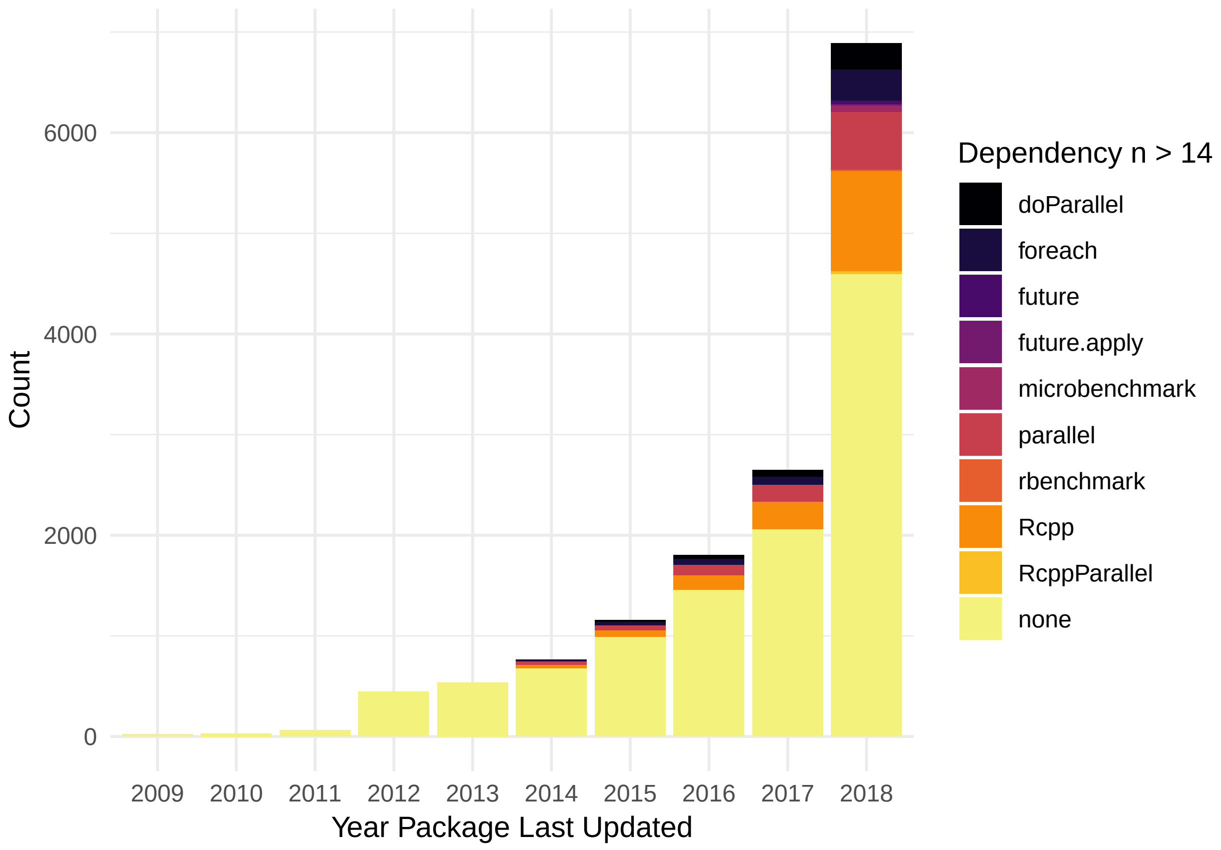
```
## Packages with 0 dependencies: 10899
## Packages with 1 dependencies: 1911
## Packages with 2 dependencies: 407
## Multiple Dependency Table (dep=2):
```

```
##     batchtools,parallel  batchtools,parallelMap             benchr,Rcpp
##                       1                       1                       1
##            doMC,foreach           doMC,parallel               doMC,Rcpp
##                      10                       2                       1
##      doParallel,foreach     doParallel,parallel         doParallel,Rcpp
##                     110                      14                       4
##     doParallel,sparklyr          foreach,future        foreach,parallel
##                       1                       1                      12
##            foreach,Rcpp     future,future.apply         future,parallel
##                      25                       6                       5
##             future,Rcpp microbenchmark,parallel     microbenchmark,Rcpp
##                       2                       3                      34
##       parallelDist,Rcpp    parallel,parallelMap     parallel,rbenchmark
##                       1                       3                       3
##           parallel,Rcpp           parallel,Rmpi           parallel,snow
##                     124                       6                       1
##         rbenchmark,Rcpp       Rcpp,RcppParallel               Rcpp,Rmpi
##                       8                      23                       1
##             Rcpp,tictoc               Rmpi,snow
##                       1                       3
## Packages with 3 dependencies: 220
## Multiple Dependency Table (dep=3):
##     batchtools,future,future.apply            doMC,doParallel,foreach
##                                  1                                  4
##                doMC,doSNOW,foreach              doMC,foreach,parallel
##                                  1                                  1
##                  doMC,foreach,Rcpp           doMPI,doParallel,foreach
##                                  1                                  1
##             doMPI,foreach,parallel  doParallel,foreach,microbenchmark
##                                  1                                  1
##        doParallel,foreach,parallel    doParallel,foreach,parallelDist
##                                133                                  1
##           doParallel,foreach,profr            doParallel,foreach,Rcpp
##                                  1                                 39
##     doParallel,microbenchmark,Rcpp           doParallel,parallel,Rcpp
##                                  1                                  4
##            doSNOW,foreach,parallel                doSNOW,foreach,Rcpp
##                                  5                                  3
##                doSNOW,foreach,snow            foreach,future,parallel
##                                  2                                  1
##              foreach,parallel,Rmpi              foreach,parallel,snow
##                                  1                                  1
## future,future.apply,microbenchmark       future,future.apply,parallel
##                                  1                                  1
##           future,future.apply,Rcpp               future,parallel,Rcpp
##                                  2                                  2
##       microbenchmark,parallel,Rcpp   microbenchmark,Rcpp,RcppParallel
##                                  1                                  2
##         parallel,parallelDist,Rcpp        parallel,parallelMap,tictoc
##                                  1                                  1
##         parallel,Rcpp,RcppParallel                 parallel,Rcpp,Rmpi
##                                  2                                  1
##                 parallel,Rmpi,snow       rbenchmark,Rcpp,RcppParallel
##                                  2                                  1
## Packages with 4 dependencies: 53
## Multiple Dependency Table (dep=4):
##            doMC,doParallel,foreach,Rcpp
##                                       1
##     doMPI,doParallel,foreach,rbenchmark
##                                       1
##      doParallel,doSNOW,foreach,parallel
##                                       1
##          doParallel,doSNOW,foreach,Rcpp
##                                       1
##          doParallel,foreach,future,Rcpp
##                                       1
##       doParallel,foreach,parallel,profr
##                                       1
##        doParallel,foreach,parallel,Rcpp
##                                      39
##        doParallel,foreach,parallel,Rmpi
##                                       3
##            doSNOW,foreach,parallel,Rcpp
##                                       1
##    foreach,future,future.apply,parallel
##                                       1
##       future,future.apply,parallel,Rcpp
##                                       1
##    microbenchmark,parallel,profvis,Rcpp
##                                       1
## microbenchmark,parallel,rbenchmark,Rcpp
##                                       1
## Packages with 5 dependencies: 16
## Multiple Dependency Table (dep=5):
##                  doMPI,doParallel,foreach,parallel,Rcpp
##                                                       1
##        doParallel,foreach,microbenchmark,parallel,Rcpp
##                                                       5
##            doParallel,foreach,parallel,rbenchmark,Rcpp
##                                                       1
##          doParallel,foreach,parallel,Rcpp,RcppParallel
##                                                       5
##                  doParallel,foreach,parallel,Rcpp,Rmpi
##                                                       1
##                      doSNOW,foreach,parallel,Rcpp,snow
##                                                       1
##              foreach,future,future.apply,parallel,snow
##                                                       1
## microbenchmark,parallel,parallelDist,Rcpp,RcppParallel
##                                                       1
## Packages with 6 dependencies: 2
## Multiple Dependency Table (dep=6):
##               doMC,doParallel,doSNOW,foreach,parallel,snow
##                                                          1
## doParallel,foreach,parallel,parallelDist,Rcpp,RcppParallel
##                                                          1
## Packages with 7 dependencies: 1
## Multiple Dependency Table (dep=7):
```

```
## doMPI,doParallel,foreach,future,parallel,parallelMap,snow
##                                                          1
## Packages with 8 dependencies: 0
## Packages with 9 dependencies: 0
## Packages with 10 dependencies: 0
## Packages with 11 dependencies: 0
## Packages with 12 dependencies: 0
## Packages with 13 dependencies: 0
## Packages with 14 dependencies: 0
## Packages with 15 dependencies: 0
## Packages with 16 dependencies: 0
## Packages with 17 dependencies: 0
## Packages with 18 dependencies: 0
## Packages with 19 dependencies: 0
## Packages with 20 dependencies: 0
## Packages with 21 dependencies: 0
## Packages with 22 dependencies: 0
## Packages with 23 dependencies: 0
## Packages with 24 dependencies: 0
## Packages with 25 dependencies: 0
## Packages with 26 dependencies: 0
## Packages with 27 dependencies: 0
```

Visualizations and Tables for Dependency results

```r
# need to merge transformed_df with package_df to keep those with no dependency and create better histo
transformed_df <- transform_df(package_df)
transformed_df <- merge(transformed_df, package_df[package_df$package_deps == 'none', ][colnames(transf

dep_freqs <- aggregate(strftime(transformed_df$date, "%Y"),
                       by=list(strftime(transformed_df$date, "%Y"), transformed_df$package_deps),
                       FUN=length)

names(dep_freqs) <- c('Year', 'Dependency', 'Count')
dep_freqs$Dependency <- reorder_none(dep_freqs$Dependency)

fplot <- ggplot(data=dep_freqs[dep_freqs$Count > 14, ], aes(x=Year, y=Count, fill=Dependency)) +
  geom_bar(stat="identity") +
  scale_fill_viridis_d(option="inferno", end=0.96) +
  xlab("Year Package Last Updated") +
  labs(fill = "Dependency n > 14")
fplot
```



```r
ggsave(filename = paste0(image_base, "optimization_dependency_stacked_bar.png"), fplot,
       width = 7.2, height = 5.5, dpi = 600, units = "in", device='png')

dep_totals <- dependency_table(dep_freqs)
print('Dependency Table:')
```

```
## [1] "Dependency Table:"
```

```r
dep_totals
```

```
##                    [,1]   [,2]   [,3]   [,4]   [,5]   [,6]   [,7]   [,8]
## Year               "2005" "2006" "2007" "2008" "2009" "2010" "2011" "2012"
## Count              "   1" "   4" "   1" "  10" "  24" "  32" "  66" " 463"
## Rcpp               "  0"  "  0"  "  0"  "  0"  "  0"  "  1"  "  0"  "  4"
## Rcpp_Pct           " 0"   " 0"   " 0"   " 0"   " 0"   " 3"   " 0"   " 1"
## tictoc             "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"
## rbenchmark         " 0"   " 0"   " 0"   " 0"   " 0"   " 0"   " 0"   " 0"
## microbenchmark     " 0"   " 0"   " 0"   " 0"   " 0"   " 0"   " 0"   " 0"
## microbenchmark_Pct "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"
## benchr             "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"
## profr              "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"
## profvis            "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"
## snow               " 0"   " 0"   " 0"   " 0"   " 0"   " 0"   " 0"   " 4"
## snow_Pct           "0"    "0"    "0"    "0"    "0"    "0"    "0"    "1"
## doSNOW             "0"    "0"    "0"    "0"    "0"    "0"    "0"    "1"
## parallel           "  0"  "  0"  "  0"  "  0"  "  0"  "  0"  "  0"  "  3"
## parallel_Pct       "0"    "0"    "0"    "0"    "0"    "0"    "0"    "1"
## doParallel         "  0"  "  0"  "  0"  "  0"  "  0"  "  0"  "  0"  "  0"
## doParallel_Pct     "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"
## Rmpi               " 0"   " 0"   " 0"   " 0"   " 0"   " 0"   " 0"   " 2"
## Rmpi_Pct           "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"
## foreach            "  0"  "  0"  "  0"  "  0"  "  0"  "  0"  "  1"  "  3"
## foreach_Pct        "0"    "0"    "0"    "0"    "0"    "0"    "2"    "1"
## future             " 0"   " 0"   " 0"   " 0"   " 0"   " 0"   " 0"   " 0"
## future_Pct         "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"
## future.apply       " 0"   " 0"   " 0"   " 0"   " 0"   " 0"   " 0"   " 0"
## SparkR             "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"
## sparklyr           "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"
## batchtools         "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"
## RcppParallel       " 0"   " 0"   " 0"   " 0"   " 0"   " 0"   " 0"   " 0"
## parallelDist       "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"
## parallelMap        "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"
## doMC               " 0"   " 0"   " 0"   " 0"   " 0"   " 0"   " 1"   " 1"
## doMC_Pct           "0"    "0"    "0"    "0"    "0"    "0"    "2"    "0"
## doMPI              "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"
## partools           "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"
## DSL                "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"
##                    [,9]   [,10]  [,11]  [,12]  [,13]  [,14]
## Year               "2013" "2014" "2015" "2016" "2017" "2018"
## Count              " 566" " 773" "1170" "1822" "2699" "6964"
## Rcpp               "  6"  " 37"  " 60"  "149"  "273"  "998"
## Rcpp_Pct           " 1"   " 5"   " 5"   " 8"   "10"   "14"
## tictoc             "0"    "0"    "0"    "0"    "1"    "4"
## rbenchmark         " 0"   " 0"   " 1"   " 1"   " 3"   "16"
## microbenchmark     " 0"   " 0"   " 0"   " 5"   "11"   "67"
## microbenchmark_Pct "0"    "0"    "0"    "0"    "0"    "1"
## benchr             "0"    "0"    "0"    "0"    "0"    "1"
## profr              "0"    "0"    "0"    "1"    "0"    "1"
## profvis            "0"    "0"    "0"    "0"    "0"    "1"
## snow               " 1"   " 0"   " 1"   " 0"   " 5"   "10"
## snow_Pct           "0"    "0"    "0"    "0"    "0"    "0"
## doSNOW             "0"    "2"    "0"    "0"    "4"    "9"
## parallel           " 14"  " 32"  " 53"  "102"  "169"  "568"
## parallel_Pct       "2"    "4"    "5"    "6"    "6"    "8"
## doParallel         "  1"  "  4"  " 20"  " 38"  " 68"  "259"
## doParallel_Pct     "0"    "1"    "2"    "2"    "3"    "4"
## Rmpi               " 0"   " 0"   " 7"   " 0"   " 4"   "10"
## Rmpi_Pct           "0"    "0"    "1"    "0"    "0"    "0"
## foreach            "  5"  " 19"  " 30"  " 61"  " 83"  "305"
## foreach_Pct        "1"    "2"    "3"    "3"    "3"    "4"
## future             " 0"   " 0"   " 0"   " 0"   " 4"   "35"
## future_Pct         "0"    "0"    "0"    "0"    "0"    "1"
## future.apply       " 0"   " 0"   " 0"   " 0"   " 0"   "16"
## SparkR             "0"    "0"    "0"    "0"    "0"    "1"
## sparklyr           "0"    "0"    "0"    "1"    "1"    "9"
## batchtools         "0"    "0"    "0"    "0"    "0"    "3"
## RcppParallel       " 0"   " 0"   " 0"   " 2"   " 6"   "27"
## parallelDist       "0"    "0"    "0"    "0"    "0"    "5"
## parallelMap        "0"    "0"    "0"    "2"    "4"    "8"
## doMC               " 1"   " 1"   " 3"   " 5"   " 4"   "10"
## doMC_Pct           "0"    "0"    "0"    "0"    "0"    "0"
## doMPI              "0"    "0"    "0"    "0"    "0"    "5"
## partools            "0"    "0"    "0"    "0"    "1"    "0"
## DSL                 "0"    "0"    "1"    "0"    "0"    "0"
```

Method comparison

As the results from method 1 do not match the results from method 2, explore some of the differences.

Some points discovered: * It is possible to list a one of the sought for packages as a dependency and not acutally use the package * Speculation - Optimization can occur in many ways. Some well known items are (at the time of this writing) are to prefer matrices over data.frame or to avoid using data.frames with the apply family of functions. It's impossible to determine for all packages what process the developer went through and what options were considered when arriving at the currently available package version.

As new packages are being released and updated all the time, these numbers will change.

```
# About 846 more appear to use compiled code
print("Difference between packages that have src directory vs dependency on optimization framework:")
```

```
## [1] "Difference between packages that have src directory vs dependency on optimization framework:"
```

```
nrow(package_df[package_df$found == TRUE, ])  - nrow(package_df[package_df$package_deps != 'none', ])
```

```
## [1] 818
```

```
# About 1701 packages that don't list a dependency to one of the specified packages, but have a src dir
only_grep <- package_df[package_df$found == TRUE & package_df$package_deps == 'none', ]
print(paste("Number of packages that don't list a dependency to one of the specified packages, but have
```

```
## [1] "Number of packages that don't list a dependency to one of the specified packages, but have a src
```

```
# About 855 packages that list a dependency to one of the specified packages but don't have compiled co
only_dep <- package_df[package_df$found == FALSE & package_df$package_deps != 'none', ]
print(paste("Number of packages that list a dependency to one of the specified packages but don't have
```

```
## [1] "Number of packages that list a dependency to one of the specified packages but don't have compil
```