

Analysis

```
library(ggplot2)
theme_set(theme_minimal())

library(purrr)
library(pracma)

##
## Attaching package: 'pracma'
## The following object is masked from 'package:purrr':
##
##     cross
##
library(dplyr)

##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
##
library(plyr)

## -----
## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)
## -----
##
## Attaching package: 'plyr'
## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
## The following object is masked from 'package:purrr':
##
##     compact
##
library(stringr)
library(scales)

##
## Attaching package: 'scales'
## The following object is masked from 'package:purrr':
##
##     discard
##
library(repr)
options(repr.plot.width=4, repr.plot.height=3)

df <- read.csv("internal-control-experiments.csv")
df$iteration=factor(df$iteration)
which.nonnum <- function(x) {
```

```

    which(is.na(suppressWarnings(as.numeric(as.character(x)))))
  }

df$time=as.POSIXct(df$time/1000000, origin="1970-01-01")
unique(df$name)

## [1] pcap150  pcap180  pcap210  control0n
## Levels: control0n pcap150 pcap180 pcap210

unique(df$variable)

## [1] sensor-RaplKey (PackageID 1)
## [2] sensor-DownstreamCmdKey (DownstreamCmdID 5a9dad07-0202-4e61-80d7-6039928a1b7c)
## [3] sensor-DownstreamCmdKey (DownstreamCmdID 405b1d73-c345-4c63-9a06-2f5f812a00b3)
## [4] sensor-DownstreamCmdKey (DownstreamCmdID 4bc344f7-5432-4cff-bfa6-e8462e388743)
## [5] actionType
## [6] RaplKey (PackageID 0)(action)
## [7] 0.00015-probability
## [8] 0.00015-cumulativeLoss
## [9] 0.00018-probability
## [10] 0.00018-cumulativeLoss
## [11] 0.00021-probability
## [12] 0.00021-cumulativeLoss
## [13] RaplKey (PackageID 1)(action)
## [14] sensor-DownstreamCmdKey (DownstreamCmdID 71ebb457-47e6-4c63-9a40-8d9ec165e5fd)
## [15] constraint-0
## [16] objective-0
## [17] objective-1
## [18] loss
## [19] sensor-DownstreamCmdKey (DownstreamCmdID 9dc47194-a7f0-4604-badd-9742ad689e85)
## [20] sensor-DownstreamCmdKey (DownstreamCmdID 4b91a304-1806-4fb4-b856-1dacc67a3e1d)
## [21] sensor-DownstreamCmdKey (DownstreamCmdID 9e6cb4ee-e10a-45ea-b41e-73895ac1946d)
## [22] sensor-DownstreamCmdKey (DownstreamCmdID 2eb2242d-24c2-43c9-b0c1-bbf17bcb67bb)
## 22 Levels: 0.00015-cumulativeLoss ... sensor-RaplKey (PackageID 1)

pStatic=200

df_rapl0 = df[which ("sensor-RaplKey (PackageID 1)"==df$variable ), c("name","value","iteration","time")]
df_rapl0$value=as.numeric(as.character(df_rapl0$value))
df_rapl1 = df[which ("sensor-RaplKey (PackageID 0)"==df$variable ), c("name","value","iteration","time")]

df_rapl1$value=as.numeric(as.character(df_rapl1$value))

df_rapl =
  bind_rows(df_rapl0, df_rapl1) %>%
  group_by(name,iteration,time) %>%
  summarise_all(sum)
df_rapl$value=as.numeric(as.character(df_rapl$value))
integrateWattHours <- function(ts,vs) {
  return(tail(cumtrapz(as.numeric(ts)/3600,vs),n=1))
}
df_rapl = ddply(df_rapl, .(name,iteration), summarize, wh=integrateWattHours(time,value))
colnames(df_rapl)<-c("name", "iteration", "wh")
summary(df_rapl)

##          name      iteration      wh.V1
## control0n:2      0:4      Min.    :2.2043169
## pcap150   :2      1:4      1st Qu.:2.3265516
## pcap180   :2                      Median :2.3785431
## pcap210   :2                      Mean    :2.3502982
##                      3rd Qu.:2.4109349

```

```

##                                Max.      :2.4269632

df_progress = df[which(grepl("sensor-Downstream",df$variable )), c("name","value","iteration")]
df_progress$value=as.numeric(as.character(df_progress$value))
df_progress = ddply(df_progress, .(name,iteration), summarize, instructions=sum(value))
summary(df_progress)

##          name      iteration  instructions
## control0n:2  0:4          Min.    :1.954e+12
## pcap150    :2   1:4          1st Qu.:1.971e+12
## pcap180    :2                Median :1.992e+12
## pcap210    :2                Mean   :1.990e+12
##                3rd Qu.:2.010e+12
##                Max.    :2.025e+12

df_runtime = df[, c("name","iteration","time")]
df_runtime$value=as.numeric(as.character(df_runtime$value))

## Error in `.$<-data.frame`(`*tmp*`, value, value = numeric(0)): replacement has 0 rows, data has 7074

df_runtime = ddply(df_runtime, .(name,iteration), summarize, runtime=max(time)-min(time))
summary(df_runtime)

##          name      iteration   runtime
## control0n:2  0:4          Length:8
## pcap150    :2   1:4          Class :difftime
## pcap180    :2                Mode  :numeric
## pcap210    :2

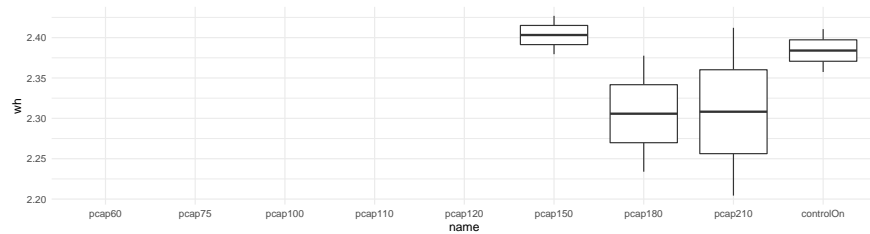
df_post = df_runtime %>%
  full_join(.,df_rapl,by=c("name","iteration")) %>%
  full_join(.,df_progress,by=c("name","iteration"))

df_reference= filter(df_post, name == "pcap210")
runtime_reference = mean(df_reference$runtime)
df_post$constraintBreach = df_post$runtime > (1.1 * runtime_reference)
summary(df_post)

##          name      iteration   runtime          wh.V1
## control0n:2  0:4          Length:8          Min.    :2.2043169
## pcap150    :2   1:4          Class :difftime  1st Qu.:2.3265516
## pcap180    :2                Mode  :numeric  Median :2.3785431
## pcap210    :2                Mean   :2.3502982
##                3rd Qu.:2.4109349
##                Max.    :2.4269632
## instructions      constraintBreach
## Min.    :1.954e+12  Mode :logical
## 1st Qu.:1.971e+12  FALSE:7
## Median :1.992e+12  TRUE :1
## Mean   :1.990e+12
## 3rd Qu.:2.010e+12
## Max.    :2.025e+12

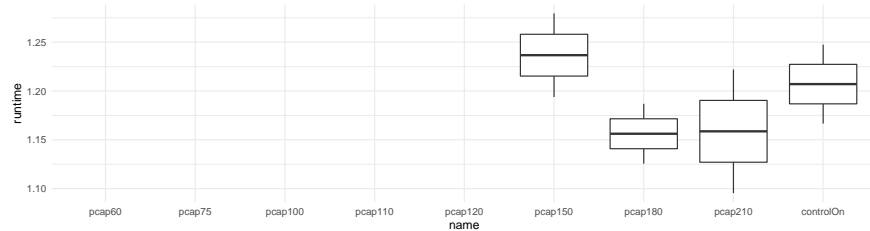
ggplot(data=df_post, aes(x=name, y=wh) ) +
  geom_boxplot() +
  scale_x_discrete(limits=c( "pcap60", "pcap75","pcap100", "pcap110", "pcap120","pcap150", "pcap180", "pcap210"
  ))

```

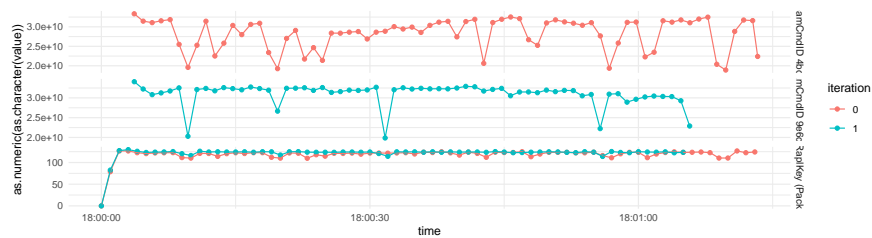


```
ggplot(data=df_post, aes(x=name, y=runtime)) +
  geom_boxplot() +
  scale_x_discrete(limits=c( "pcap60", "pcap75", "pcap100", "pcap110", "pcap120", "pcap150", "pcap180", "pcap210"
  ))
```

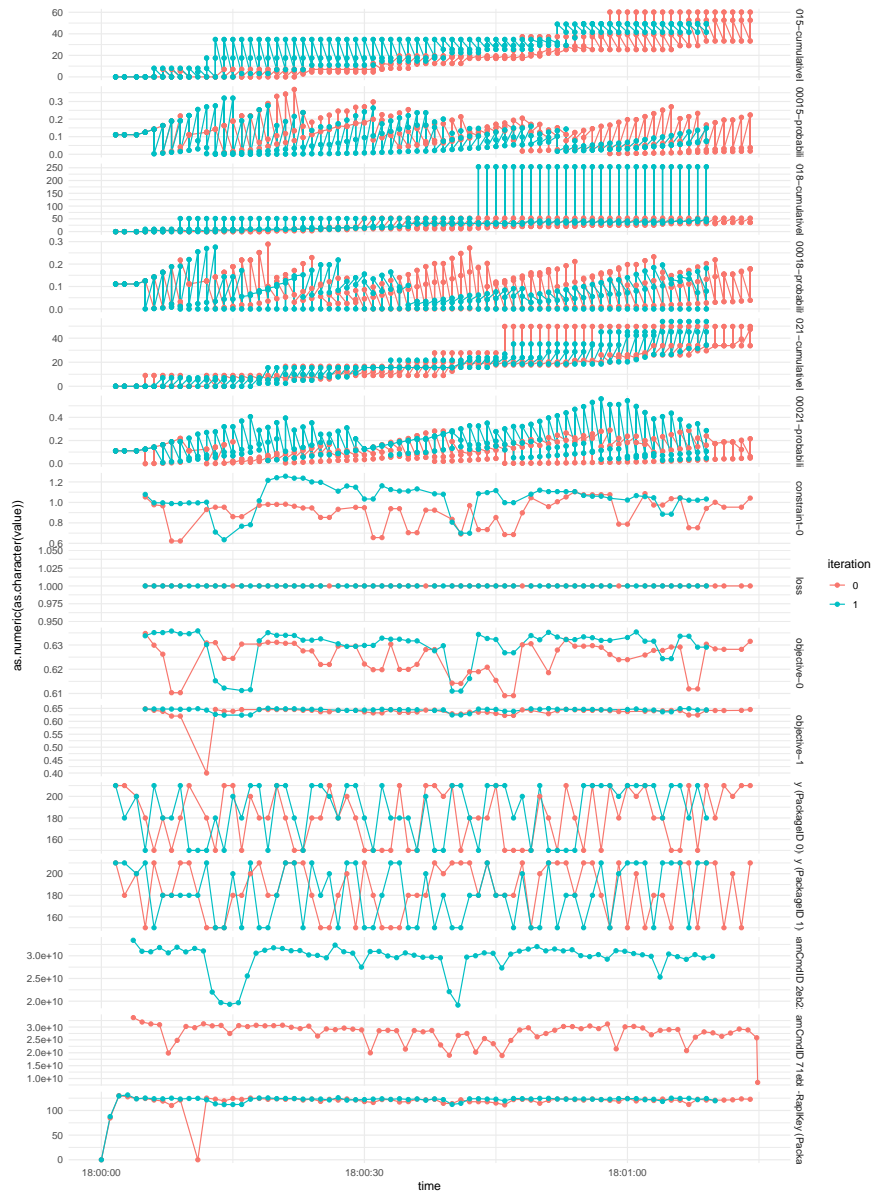
Don't know how to automatically pick scale for object of type difftime. Defaulting to continuous.



```
ggplot(data=df[which(df$name=="pcap210"),], aes(time, as.numeric(as.character(value)), group=iteration, color=iteration)) +
  geom_point() +
  geom_line() +
  facet_grid(rows = vars(variable), scales="free_y") +
  scale_x_datetime()
```



```
df_control=df[which(df$name=="controlOn"),]
ggplot(df_control[which(df_control["variable"] != "actionType"),], aes(x=time, y=as.numeric(as.character(value)))) +
  geom_line() +
  geom_point() +
  facet_grid(rows = vars(variable), scales="free_y") +
  scale_x_datetime()
```



```
ggplot(df_control[which(df_control["variable"] == "actionType"),], aes(x=time, y=as.character(value), group=iteration)) +
  geom_line() +
  geom_point() +
  facet_grid(rows = vars(variable), scales="free_y") +
  scale_x_datetime()
```

