

# Implementation Comparison

Patrick Ding

8/14/2020

## Unit Gaussian

### Rcpp implementation

```
n_rep <- 30

d <- 4
mu <- rep(0, d)
Sigma <- diag(d)
lb <- rep(0, d)
ub <- rep(Inf, d)

set.seed(0)
rcpp_ests <-
  replicate(n_rep,
            lincongauss::ptmvm(mu, Sigma, lb, ub,
                               n_sub_samples = 512, n_sub_skip = 1,
                               n_hdr_samples = 512, n_hdr_skip = 1)
  )
```

### Python implementation

```
import sys
sys.path.append('../LinConGauss/src/LinConGauss/sampling')
sys.path.append('../LinConGauss/src/LinConGauss/core')
sys.path.append('../LinConGauss/src/LinConGauss/multilevel_splitting')

import LinConGauss
import numpy as np

# equivalent constraint
d = 4
A = np.eye(d)
b = np.zeros(d, 1)
np.random.seed(0)

# define the linear constraints with LinConGauss
lincon = lcg.LinearConstraints(A=A, b=b)

py_ests = np.zeros(n_rep)
for i in range(n_rep):
```

```

subsetsim = lcg.multilevel_splitting.SubsetSimulation(linear_constraints=lincon,
                                                    n_samples=16,
                                                    domain_fraction=0.5,
                                                    n_skip=3)

subsetsim.run(verbose=False)

hdr = lcg.multilevel_splitting.HDR(linear_constraints=lincon,
                                   shift_sequence=shifts,
                                   n_samples=512,
                                   n_skip=9,
                                   X_init=subsetsim.tracker.x_inits())

hdr.run(verbose=False)

hdr_integral = hdr.tracker.integral()
py_ests[i] = hdr_integral
par(mfrow = c(1, 2))
boxplot(rcpp_ests)
abline(h = 1 / 2^d)
print(py$py_ests)

## NULL
# boxplot(py$py_ests)
# abline(h = 1 / 2^d)

```

