Pathfinder Investigation

Intro

This is a simple repo investigating some behavior I am seeing with the Pathfinder algorithm. We are currently using the version of CmdStan with Pathfinder+Laplace and associated verison of cmdstanr found in Steve Bronder's repo here https://github.com/SteveBronder/laplace_testing

The STAN model we run is `fh.stan`

Code

First read in libraries

```
####################
## load libraries
####################
suppressMessages(  library(cmdstanr) )
suppressMessages(  library(dplyr) )
suppressMessages(  library(jsonlite) )

#set cmdstan path
# SET THIS TO YOUR VERSION OF PATHFINDER!!
set_cmdstan_path("/ext/work/cmdstan/laplace_testing/cmdstan")
```

```
## CmdStan path set to: /ext/work/cmdstan/laplace_testing/cmdstan
```

```
#read in our dataset
stan_data <- readRDS("stan_data.rds")
```

Run 1, In Parallel

Note: In order to not flood this document with text, the output from the Pathfinder algorithm can be found in `output1.txt` inside this repo.

```
####### Compile with parallel

#removing executable to force recompile (had trouble with force_recompile)
temp <- file.remove("fh")

### compile stan script
## note that I use capture.output to make the markdown cleaner
temp <- capture.output( mod1     <- cmdstanr::cmdstan_model("fh.stan", cpp_options=list(stan_threads=TRU

fileConn<-file("output1.txt")
output <- capture.output({
fit1 = mod1$pathfinder(algorithm = "multi",
```

```
                      data = stan_data,
                    refresh = 1,
                      threads=12,
                      num_threads = 12,
                      num_paths = 12)
}) #end capture output
cat(output, file=fileConn , sep="\n")
close(fileConn)

draws_df1 <- fit1$draws("y_rep",format="df")
```

Investigate Output from 1st Run

```
head(draws_df1)
```

```
## # A draws_df: 6 iterations, 1 chains, and 560 variables
##   y_rep[1] y_rep[2] y_rep[3] y_rep[4] y_rep[5] y_rep[6] y_rep[7] y_rep[8]
## 1    -1.71    -0.18     -2.2    -0.55     -1.8   -0.218     1.11      4.2
## 2    -1.71    -0.18     -2.2    -0.55     -1.8   -0.218     1.11      4.2
## 3    -1.71    -0.18     -2.2    -0.55     -1.8   -0.218     1.11      4.2
## 4    -1.13    -0.72     -2.1    -1.61     -2.3    1.023     1.06     -1.4
## 5    -0.41    -1.44     -2.7    -2.07      0.1    0.091     1.00      1.8
## 6     0.28    -1.62     -1.8    -1.46     -2.5    0.027     0.92      2.3
## # ... with 552 more variables
## # ... hidden reserved variables {'.chain', '.iteration', '.draw'}
```

```
draw_dist1 <-  draws_df1 %>% select( starts_with("y_rep" )) %>% distinct()
```

```
## Warning: Dropping 'draws_df' class as required metadata was removed.
```

```
n_dist1 <- draw_dist1 %>% nrow()
```

Note: Out of 1999, there are 10 unique rows from the draws.

Run 2, Not in Parallel

Note: In order not to flood this document with text, the output from the Pathfinder algorithm can be found in `output2.txt` inside this repo.

```
#removing executable to force recompile (had trouble with force_recompile)
temp <- file.remove("fh")

### compile stan script
## note that I use capture.output to make the markdown cleaner
temp <- capture.output( mod2     <- cmdstanr::cmdstan_model("fh.stan"))

fileConn<-file("output2.txt")
output <- capture.output({

fit2 = mod2$pathfinder(algorithm = "multi",
                      data = stan_data,
                    refresh = 1,
                      num_paths = 12)
```

```
})
cat(output, file=fileConn, sep="\n" )
close(fileConn)

draws_df2 <- fit2$draws("y_rep",format="df")
```

Taking a look at the head of this dataset, we can see repeated values

```
head(draws_df2)
```

```
## # A draws_df: 6 iterations, 1 chains, and 560 variables
##   y_rep[1] y_rep[2] y_rep[3] y_rep[4] y_rep[5] y_rep[6] y_rep[7] y_rep[8]
## 1    0.097     -2.8     -2.6    -0.37    -0.54      3.5      1.3     0.52
## 2    0.097     -2.8     -2.6    -0.37    -0.54      3.5      1.3     0.52
## 3    0.097     -2.8     -2.6    -0.37    -0.54      3.5      1.3     0.52
## 4    0.097     -2.8     -2.6    -0.37    -0.54      3.5      1.3     0.52
## 5    0.097     -2.8     -2.6    -0.37    -0.54      3.5      1.3     0.52
## 6    0.097     -2.8     -2.6    -0.37    -0.54      3.5      1.3     0.52
## # ... with 552 more variables
## # ... hidden reserved variables {'.chain', '.iteration', '.draw'}
```

Further investigating this we can see the number of unique rows.

```
draw_dist2 <-  draws_df2 %>% select( starts_with("y_rep" )) %>% distinct()
```

```
## Warning: Dropping 'draws_df' class as required metadata was removed.
```

```
n_dist2 <- draw_dist2 %>% nrow()
```

Out of 1999, there are 4 unique rows from the draws.