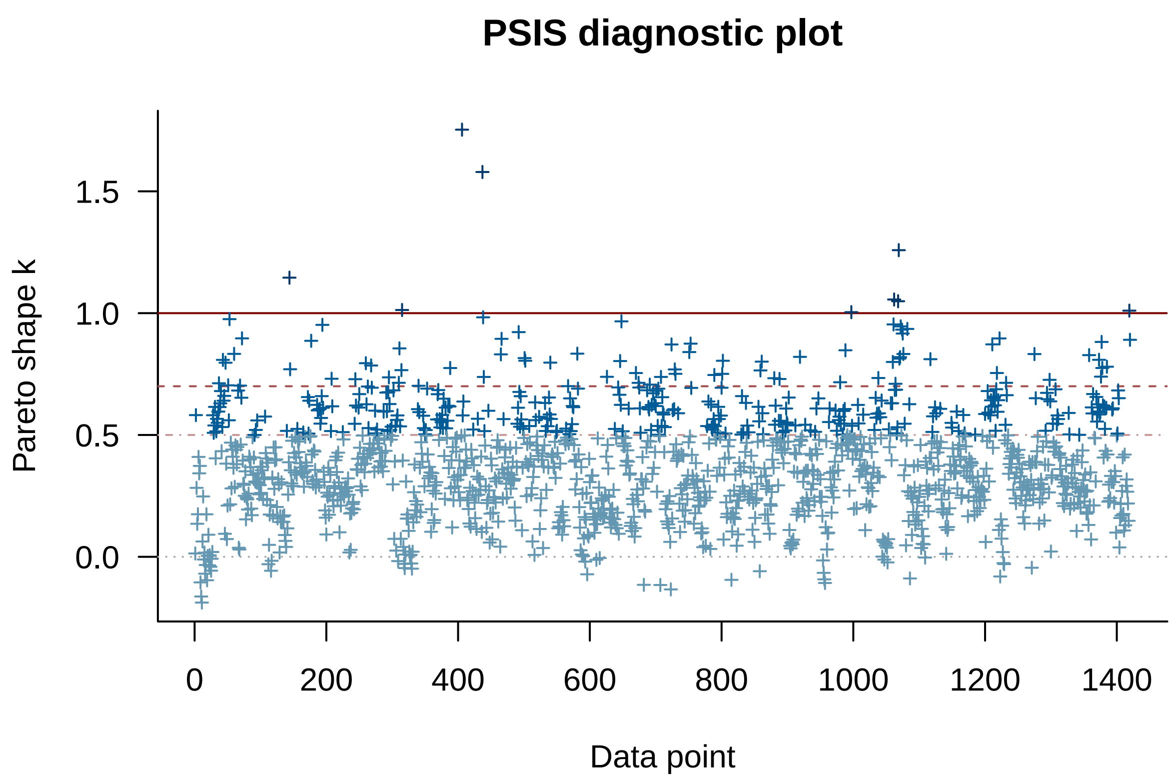
Model comp

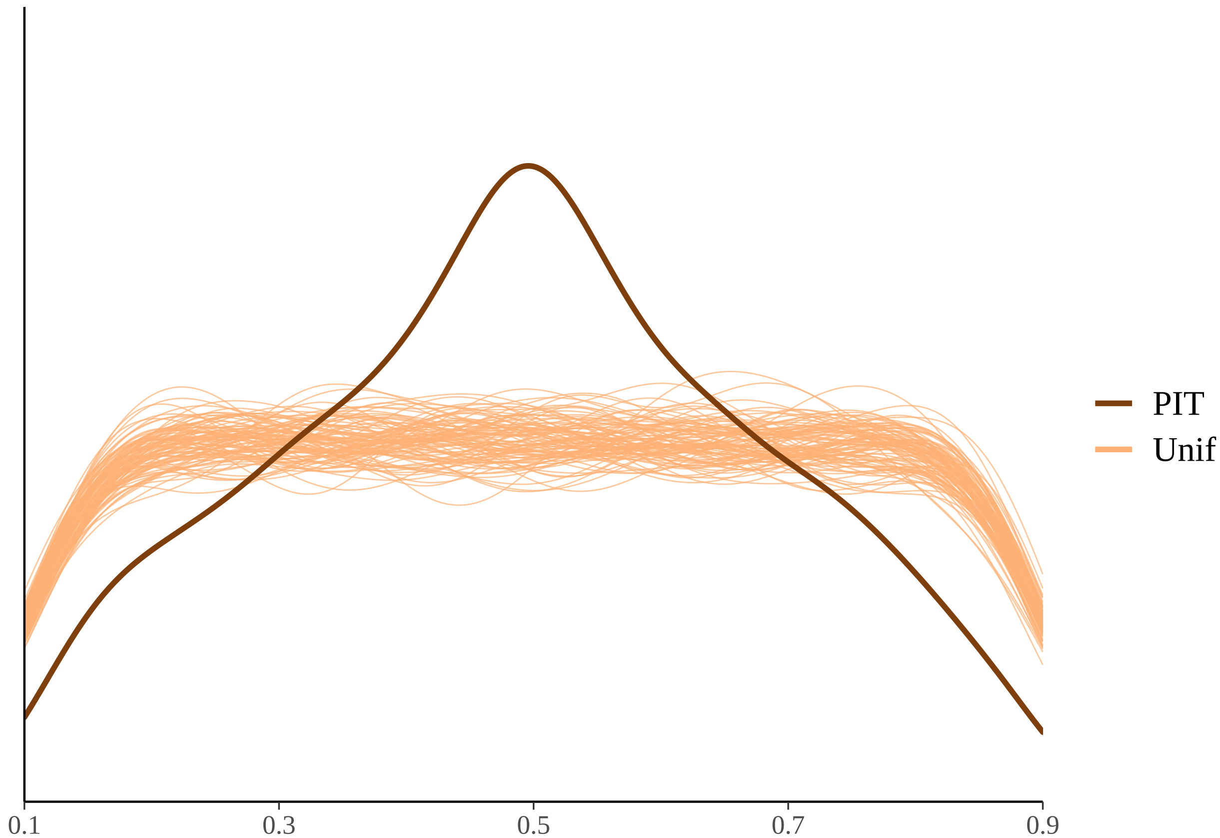
Zhengfan Wang

IN-SAMPLE

```
## [1] 4000 1420
```

```
##
## Computed from 4000 by 1420 log-likelihood matrix
##
##          Estimate    SE
## elpd_loo    861.2  51.8
## p_loo       405.4  21.1
## looic     -1722.4 103.7
## ------
## Monte Carlo SE of elpd_loo is NA.
##
## Pareto k diagnostic values:
##                          Count Pct.    Min. n_eff
## (-Inf, 0.5]   (good)     1054  74.2%   387
##  (0.5, 0.7]   (ok)        279  19.6%   102
##    (0.7, 1]   (bad)        78   5.5%   14
##    (1, Inf)   (very bad)    9   0.6%   2
## See help('pareto-k-diagnostic') for details.
```



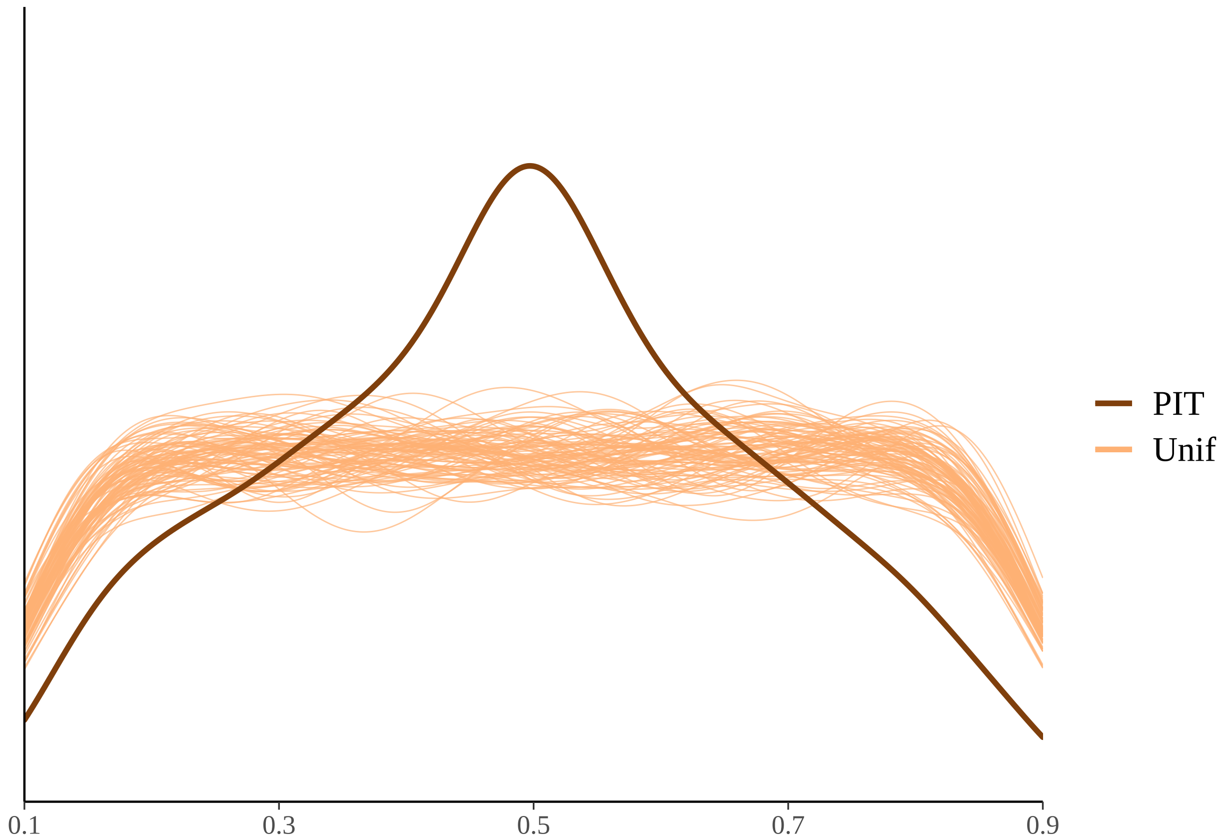


admin data

```
length(which(standata$getj_i == 1)) #num of obs
```

```
## [1] 1094
```

```
loo_by_source(source=1,dat = standata,loo=loo1)
```

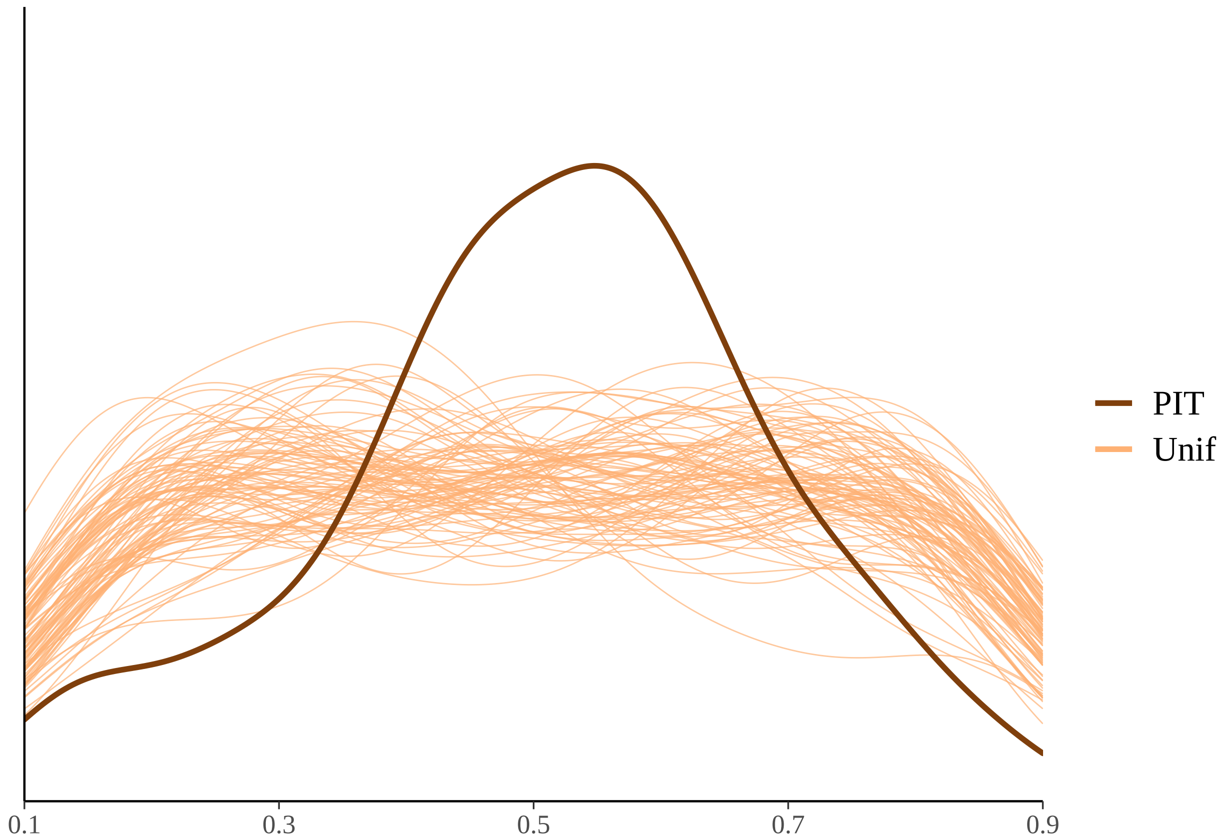


HMIS

```r
length(which(standata$getj_i == 2)) #num of obs
```

```
## [1] 116
```

```r
loo_by_source(source=2,dat = standata,loo=loo1)
```

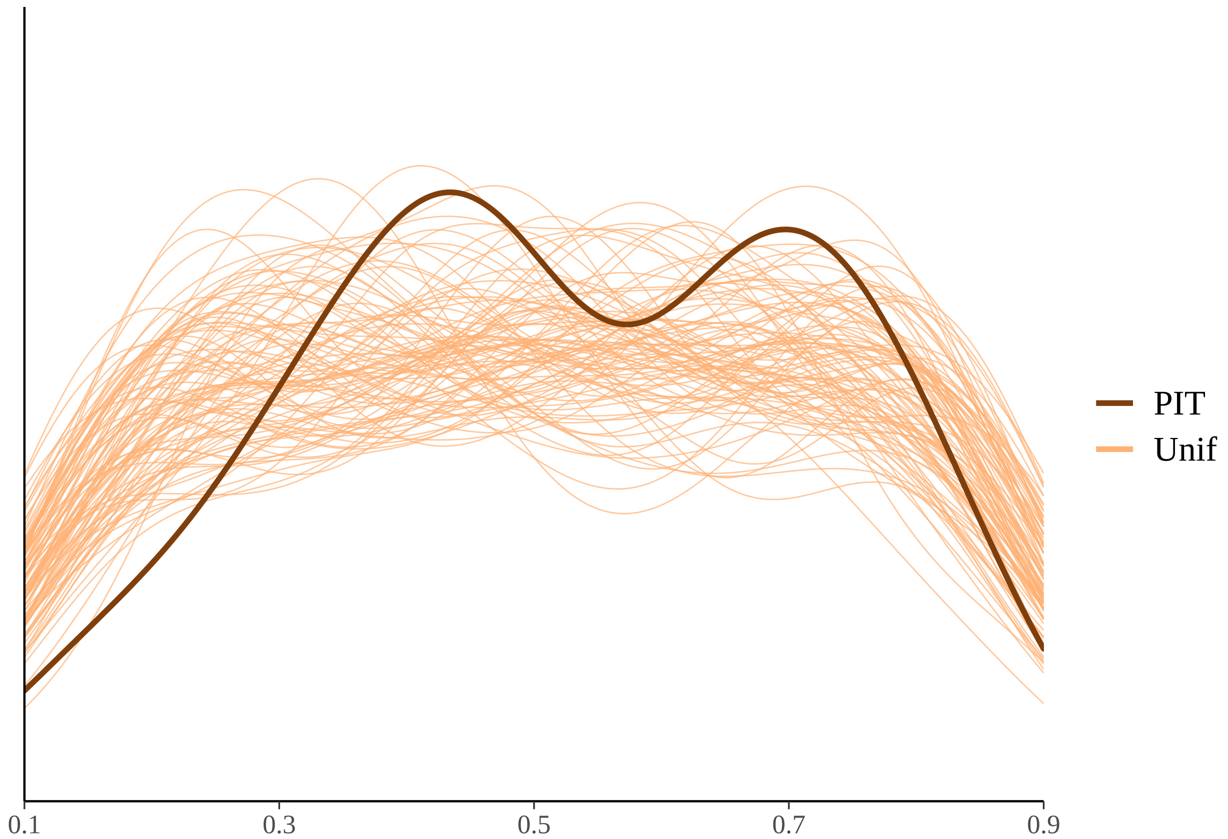


subnat LR

```
length(which(standata$getj_i == 3)) #num of obs
```

```
## [1] 120
```

```
loo_by_source(source=3,dat = standata,loo=loo1)
```

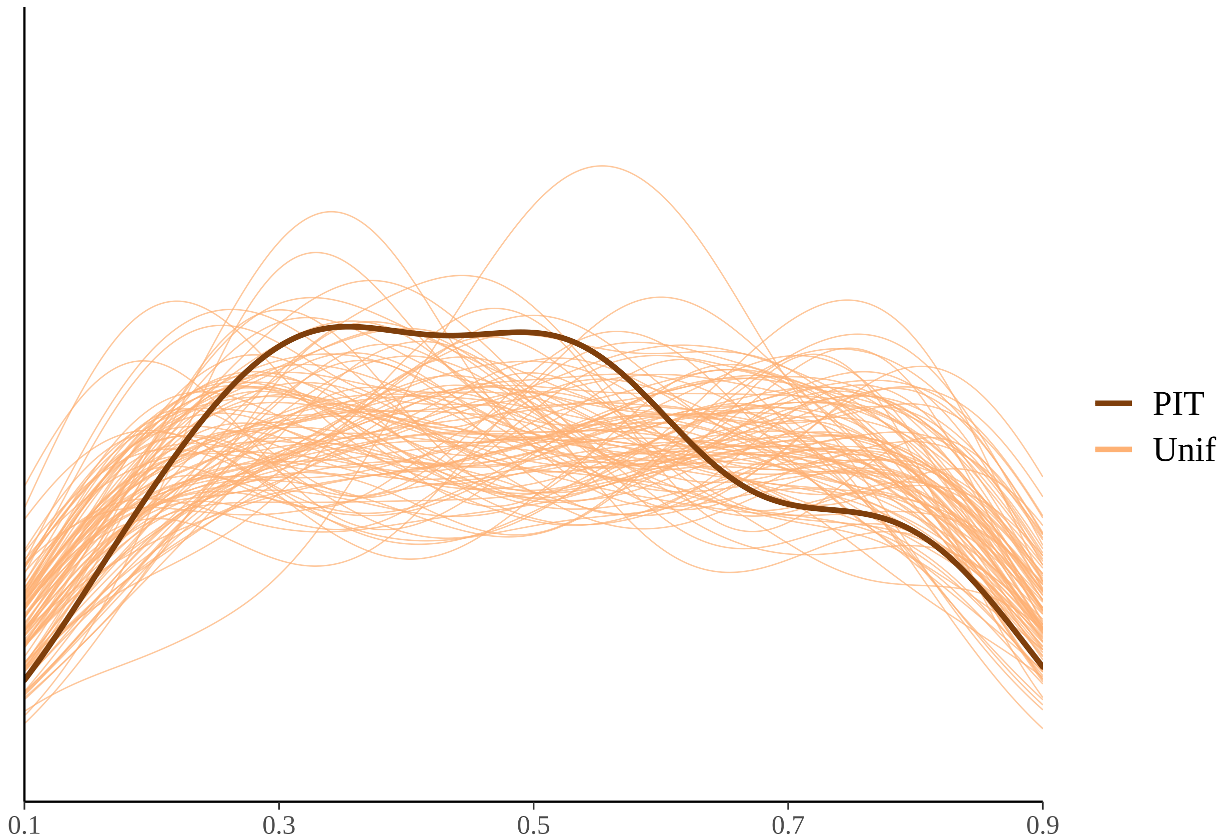


survey

```r
length(which(standata$getj_i == 4)) #num of obs
```

```
## [1] 90
```

```r
loo_by_source(source=4,dat = standata,loo=loo1)
```

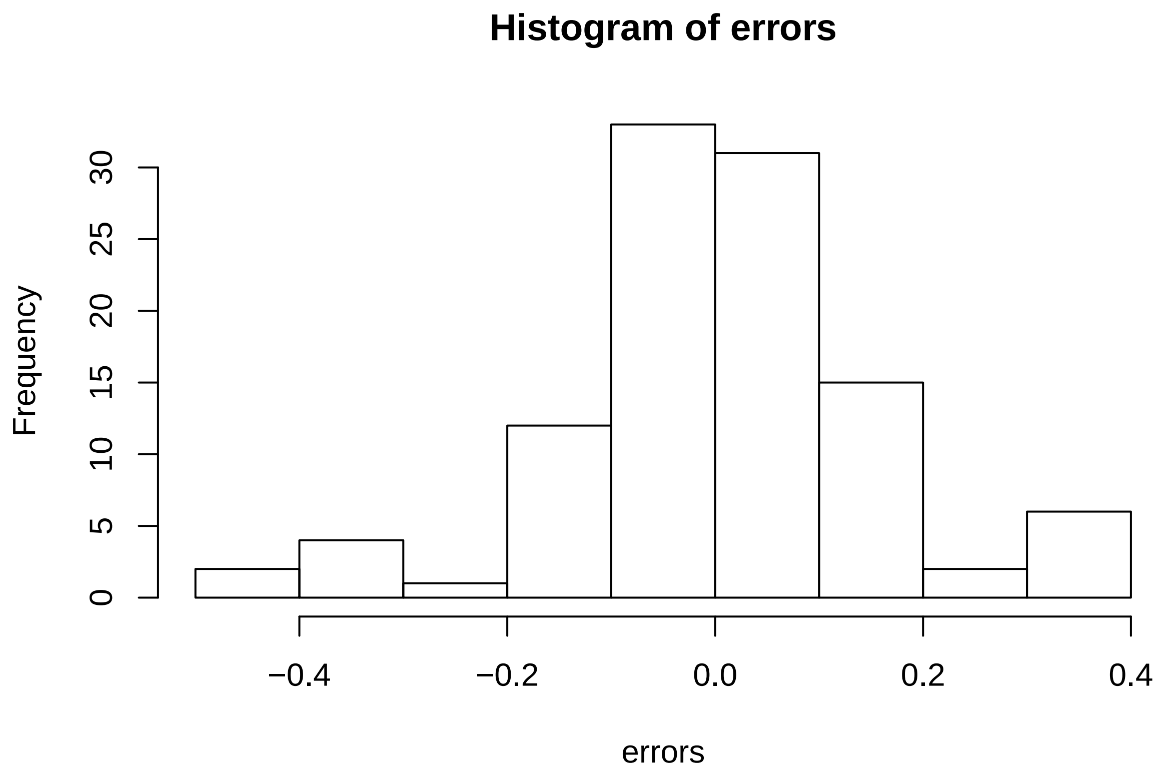


OUT-SAMPLE

```
fit <- readRDS(file = "rdsoutput/base_val.rds")
stan.data <- readRDS(file = "output/stan_data/nhs_val.rds")

array <- rstan::extract(fit)
# indices of test set
getitest <- setdiff(seq(1,stan.data$N), stan.data$getitrain_k)

# errors
pred <- apply(array$prep,2,median)
errors <- stan.data$Y[getitest] - pred[getitest]
hist(errors)
```




```r
# summarize
mean(errors)
```

```
## [1] 0.007586468
```

```r
median(errors)
```

```
## [1] 0.002718375
```

```r
mean(abs(errors))
```

```
## [1] 0.1088321
```

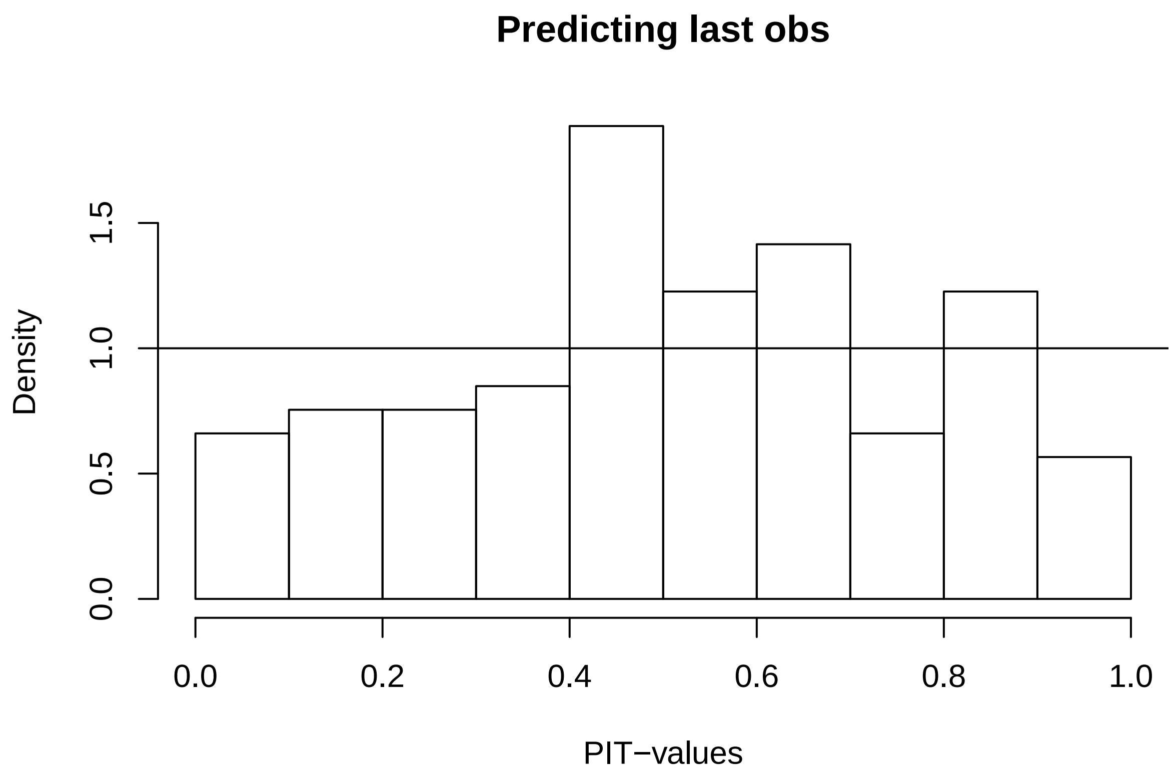
```r
median(abs(errors))
```

```
## [1] 0.07437609
```

```r
ntest <- length(getitest)
ntest
```

```
## [1] 106
```

```r
# PIT
pit.j <- rep(NA, ntest)
for (j in 1:ntest){
  i <- getitest[j]
  yrepi.s <- array$prep[,i]
  pit.j[j] <- mean(yrepi.s <= stan.data$Y[i])
}
hist(pit.j, freq = F, xlab = "PIT-values", main = "Predicting last obs")  # should look uniform
abline(h=1)
```




```r
p <- 0.1
qbinom(c(0.1, 0.9), size = ntest, prob = p)/ntest # 80% PI for prop of left out obs in one bin of PIT v
```

```
## [1] 0.06603774 0.14150943
```

```r
qbinom(c(0.25, 0.75), size = ntest, prob = p)/ntest # 50% PI for prop of left out obs in one bin of PIT
```

```
## [1] 0.0754717 0.1226415
```

```r
p <- 0.25
qbinom(c(0.1, 0.9), size = ntest, prob = p)/ntest # 80% PI for prop of left out obs in one bin of PIT v
```

```
## [1] 0.1981132 0.3018868
```

```r
qbinom(c(0.25, 0.75), size = ntest, prob = p)/ntest # 50% PI for prop of left out obs in one bin of PIT
```

```
## [1] 0.2169811 0.2735849
```

```r
# coverage follows from pit
mean(pit.j < 0.025) # % below 95% PI
```

```
## [1] 0
```

```r
mean(pit.j < 0.05) # % below 90% PI
```

```
## [1] 0.03773585
```

```r
mean(pit.j < 0.1)
```

```
## [1] 0.06603774
```

```r
mean(pit.j > 0.975) # % above 95% PI
```

```
## [1] 0
```

```r
mean(pit.j > 0.95)
```

```
## [1] 0.01886792
```

```r
mean(pit.j > 0.9)
```

```
## [1] 0.05660377
```