MAR 45% missing - DPMPM

```
# sample MCAR dataset from PUMS
source("../../utils/sampleMAR45.R")
n = 10000
missing_col = c(1,3,7,9,10,11)
set.seed(3)

output_list <- sampleMAR45(n)
df <- output_list[['df']]
df_observed <- output_list[['df_observed']]

apply(is.na(df_observed), MARGIN = 2, mean)
```

```
##    VEH     MV     NP   RMSP    ENG  MARHT   SCHL RACNUM   AGEP    WKL  PINCP
## 0.4456 0.0000 0.3998 0.0000 0.0000 0.0000 0.4842 0.0000 0.4670 0.4478 0.4384
```

DPMPM

Multiple imputation using NPBayesImputeCat package

Ref: https://cran.r-project.org/web/packages/NPBayesImputeCat/NPBayesImputeCat.pdf

1. Create and initialize the Rcpp_Lcm model object using CreateModel with the following arguments:

- X: dataframe to be imptuted = df
- MCZ: dataframe with the definition of structural zero = NULL
- K: the maximum number of mixture components = 40
- Nmax: An upper truncation limit for the augmented sample size = 0
- aalpha: the hyper parameter alpha in stick-breaking prior = 0.25
- balpha: the hyper parameter beta in stick-breaking prior = 0.25
- seed = 0

2. Set the tracer for the sampling process

- k_star: the effective cluster number
- psi: conditional multinomial probabilties
- ImputedX: imputation result

3. Run the model using the method Run of Rcpp_Lcm class with the following arguments:

- burnin = 10000
- iter = 10000
- thinning = 5

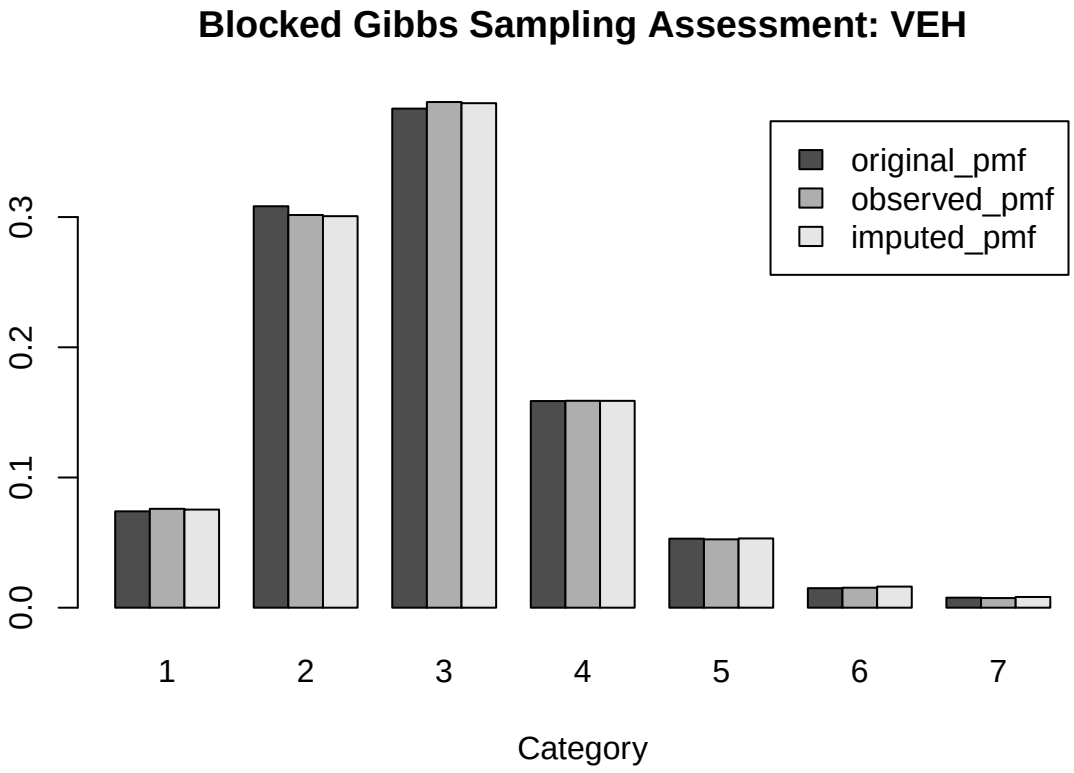
4. Obtain result

```
N = 40
Mon = 10000
B = 10000
thin.int = 5

# 1. Create and initialize the Rcpp_Lcm model object
model = CreateModel(X = df_observed, MCZ = NULL, K = N, Nmax = 0,
                    aalpha = 0.25, balpha = 0.25, seed = 0)
# 2. Set tracer
```

```
model$SetTrace(c('k_star', 'psi', 'ImputedX', 'alpha'),Mon)

# 3. Run model using Run(burnin, iter, thinning)
model$Run(B,Mon,thin.int)
```

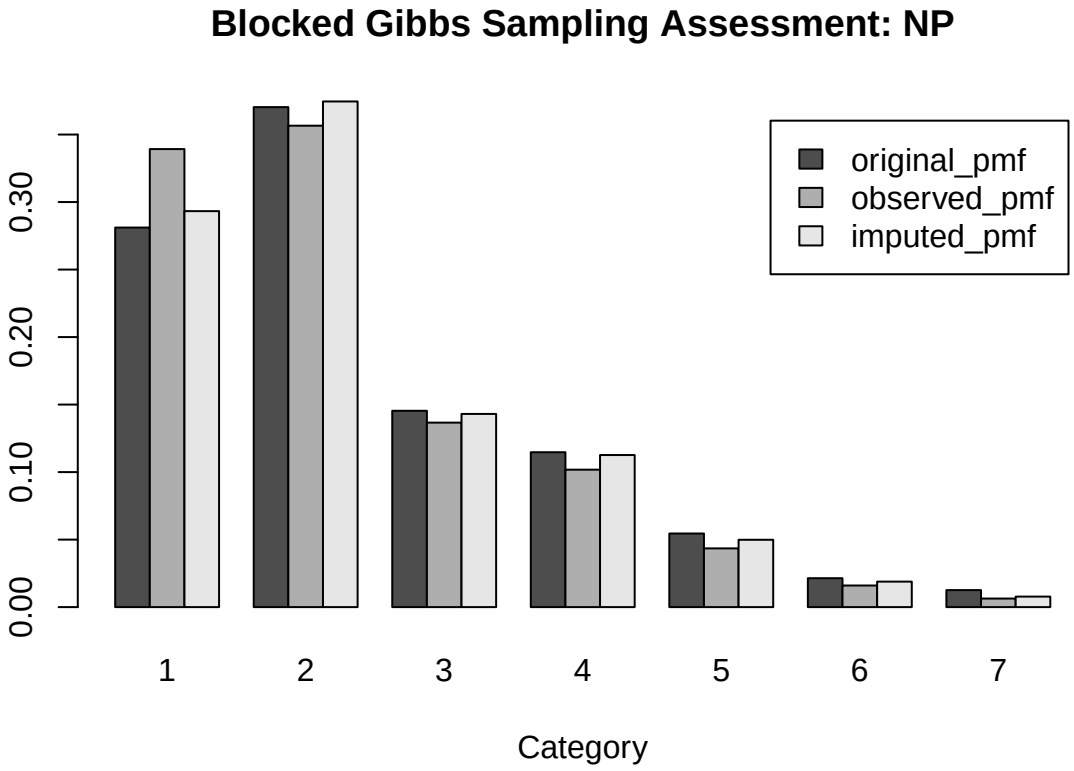
```
# Extract results
output <- model$GetTrace()
k_star <- output$k_star
psi <- output$psi
imputed_df <- output$ImputedX
alpha <- output$alpha

#retrieve parameters from the final iteration
result <- model$snapshot

#convert ImputedX matrix to dataframe, using proper factors/names etc.
ImputedX <- GetDataFrame(result$ImputedX,df)
```

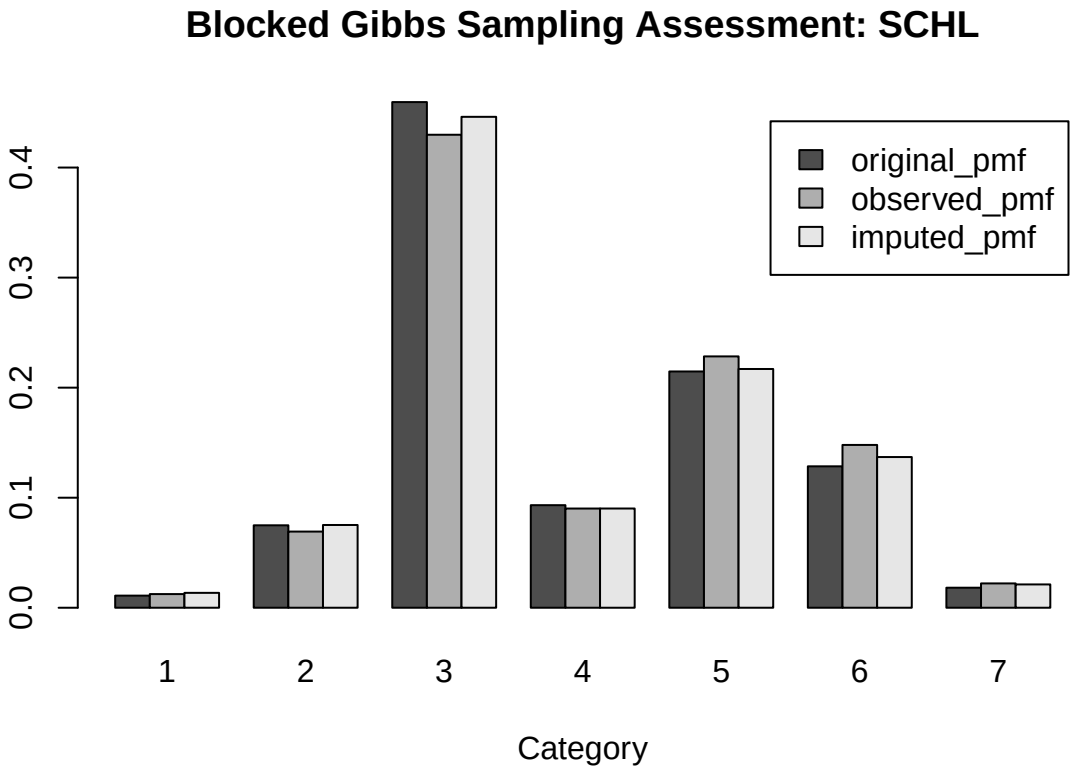
Diagnostics

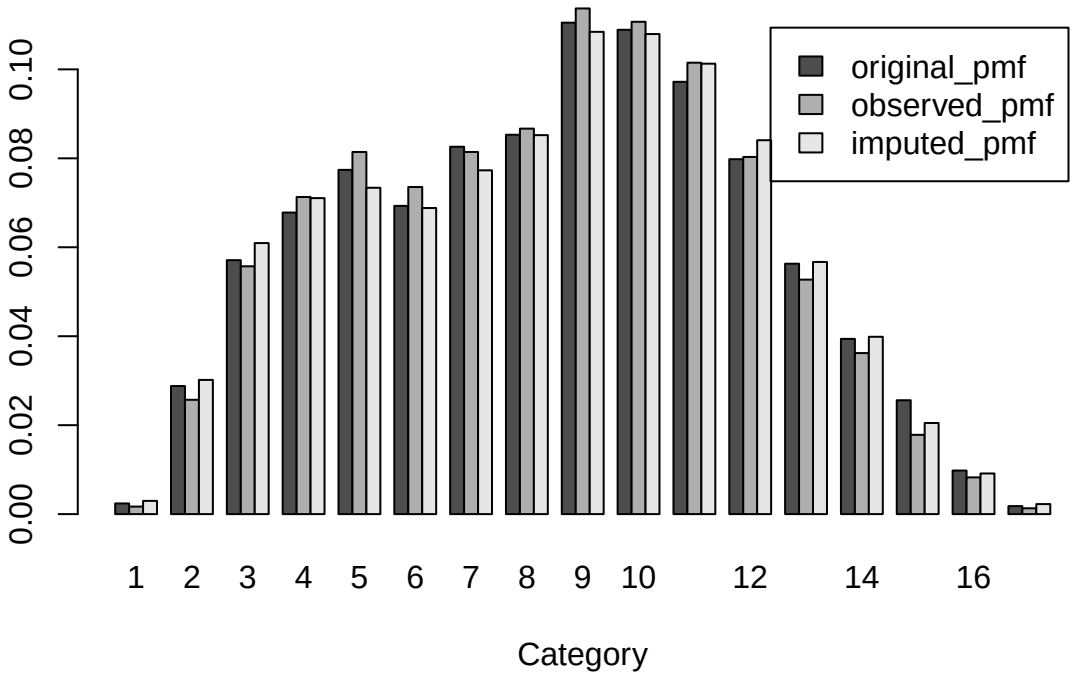


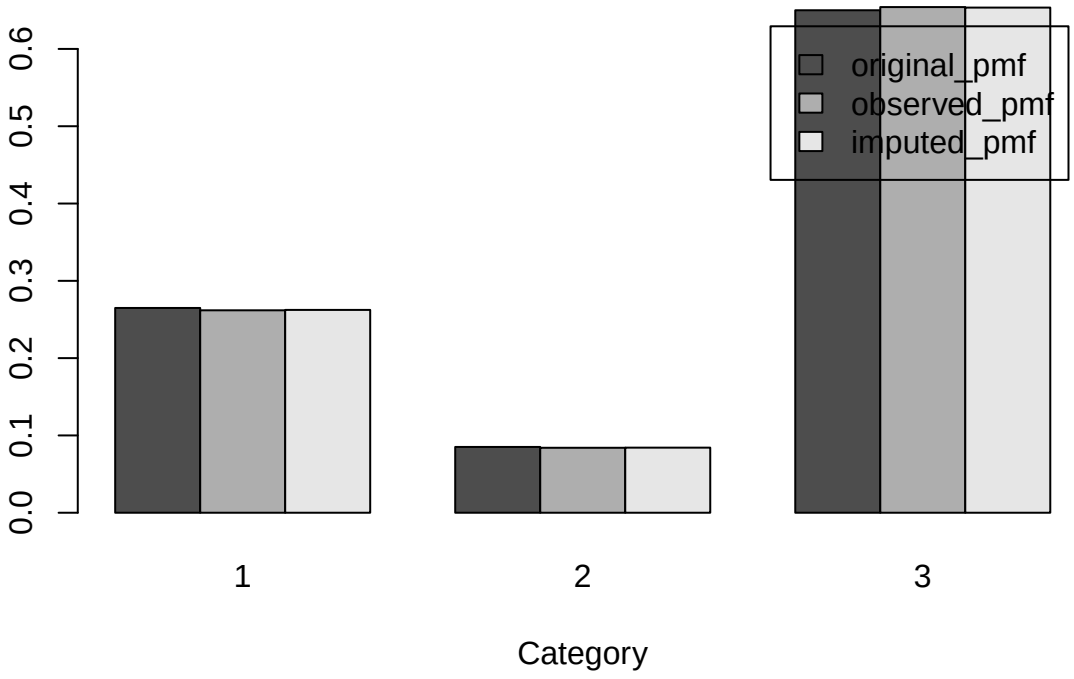
Assess bivariate joint distribution

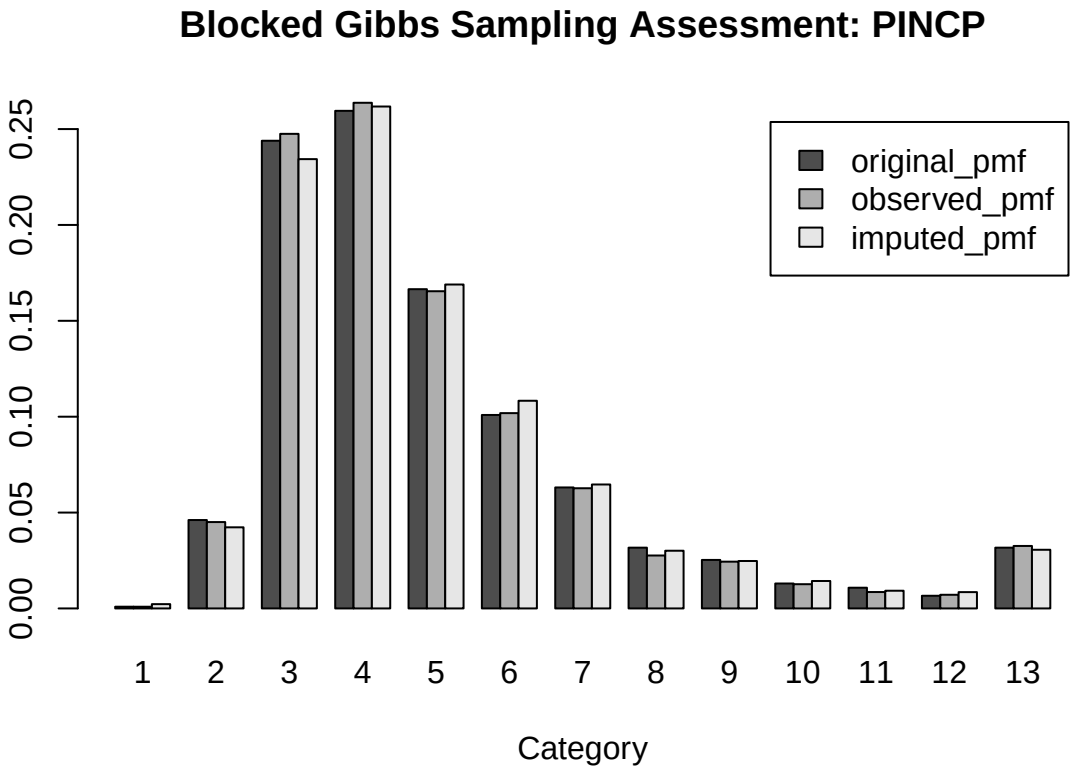
Assess trivariate joint distribution

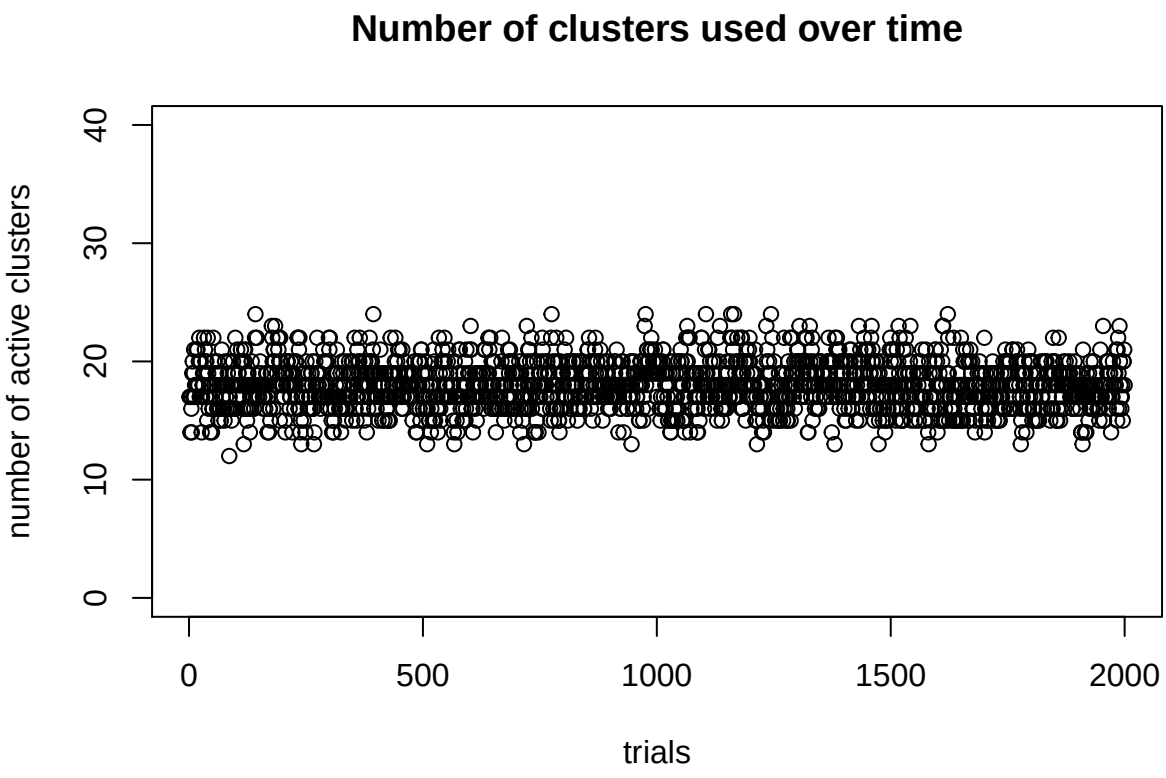

Blocked Gibbs Sampling Assessment: NP
original_pmf
observed_pmf
imputed_pmf
0.00
0.10
0.20
0.30
1
2
3
4
5
6
7
Category

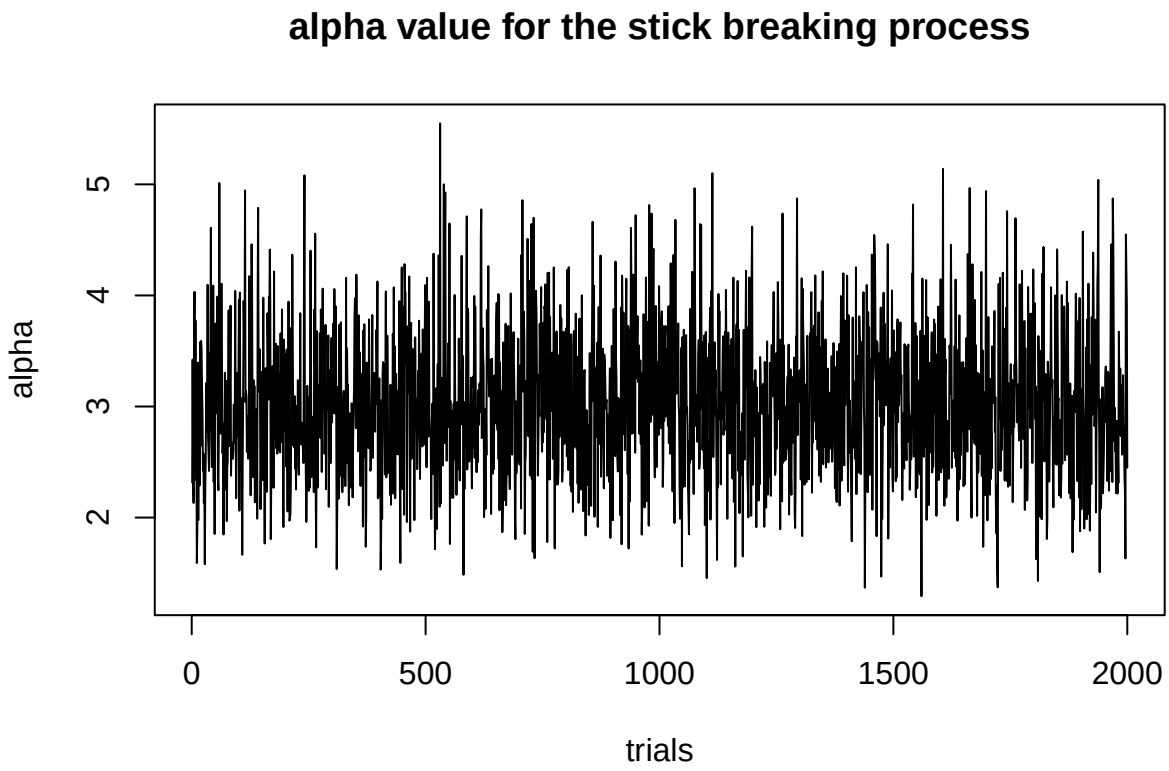

Blocked Gibbs Sampling Assessment: SCHL
original_pmf
observed_pmf
imputed_pmf
0.0
0.1
0.2
0.3
0.4
1
2
3
4
5
6
7
Category

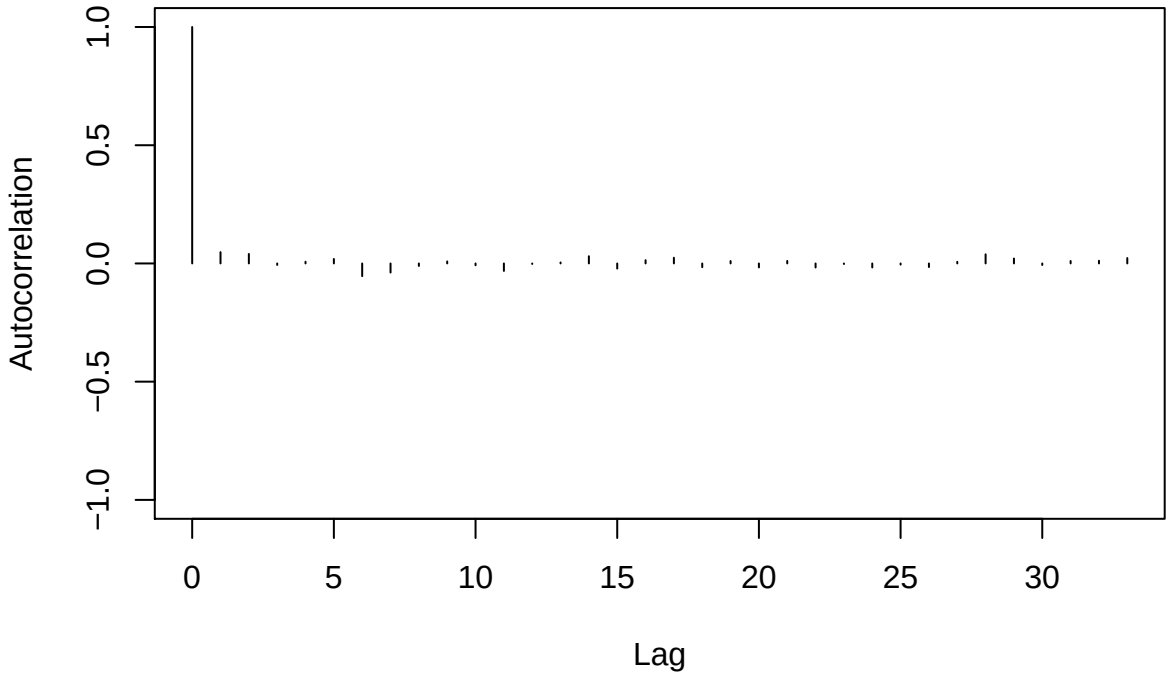

Blocked Gibbs Sampling Assessment: AGEP
original_pmf
observed_pmf
imputed_pmf
0.00
0.02
0.04
0.06
0.08
0.10
1
2
3
4
5
6
7
8
9
10
12
14
16
Category

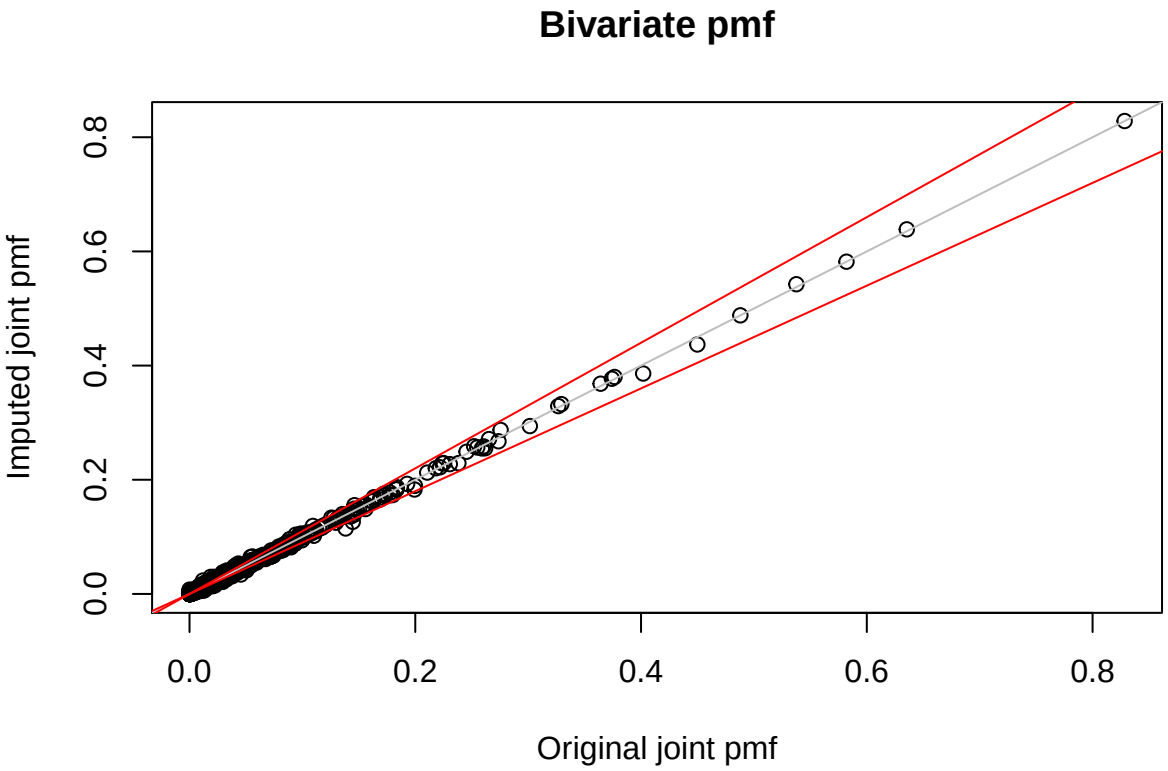

Blocked Gibbs Sampling Assessment: WKL
original_pmf
observed_pmf
imputed_pmf
0.0
0.1
0.2
0.3
0.4
0.5
0.6
1
2
3
Category

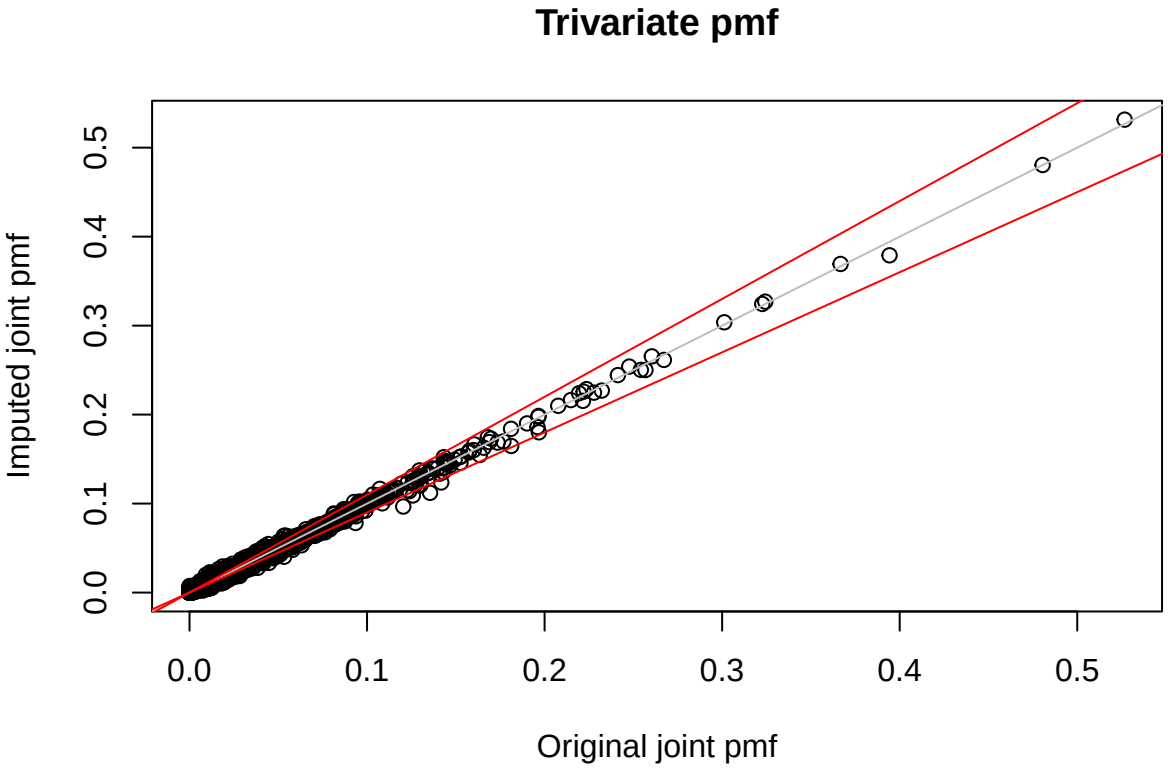

Blocked Gibbs Sampling Assessment: PINCP
original_pmf
observed_pmf
imputed_pmf
0.00
0.05
0.10
0.15
0.20
0.25
1
2
3
4
5
6
7
8
9
10
11
12
13
Category


Number of clusters used over time
number of active clusters
0
10
20
30
40
0
500
1000
1500
2000
trials


alpha value for the stick breaking process
alpha
2
3
4
5
0
500
1000
1500
2000
trials


Autocorrelation
−1.0
−0.5
0.0
0.5
1.0
0
5
10
15
20
25
30
Lag


Bivariate pmf
Imputed joint pmf
0.0
0.2
0.4
0.6
0.8
Original joint pmf


Trivariate pmf
Imputed joint pmf
0.0
0.1
0.2
0.3
0.4
0.5
Original joint pmf