

# Testing different imputation methods on PUMS (MCAR) - Random imputation

---

```
# load dataset: df
load('../Datasets/ordinalPUMS.Rdata')

# take 10,000 samples: df
set.seed(0)
n = 10000
sample <- sample(nrow(df), size = 10000)
df <- df[sample,]

# create MCAR scenario with 30% chance of missing: df_observed
missing_prob = 0.3
df_observed <- df
missing_col = colnames(df)[c(1,3,5,7,9,11)]
for (col in missing_col) {
  missing_ind <- rbernoulli(n,p = missing_prob)
  df_observed[missing_ind, col] <- NA
}
```

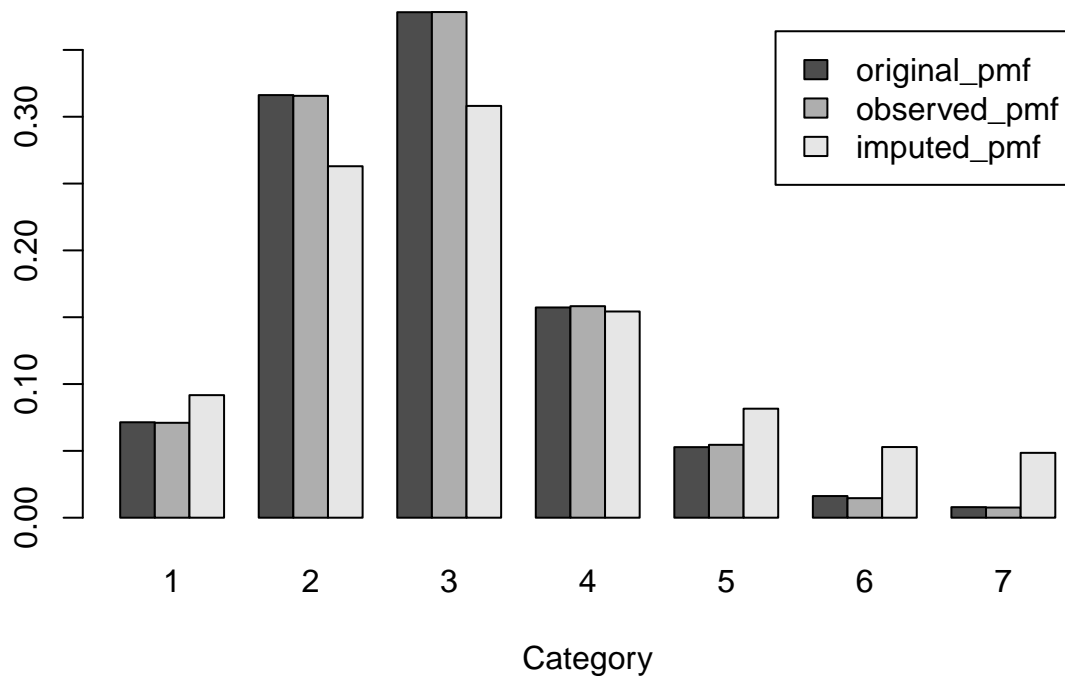
## Random imputation

Using uniform pmf for missing data imputation

```
random_imputation <- function(data){
  # Missing data imputation with uniform pmf
  # data: dataframe with missing values
  d = data
  missing_matrix = is.na(d)
  for (col in 1:dim(d)[2]) {
    # impute with random value
    pmf = rep(1/length(table(d[,col])),length(table(d[,col])))
    missing_indicator = missing_matrix[,col]

    n_missing = sum(missing_indicator)
    sample = rmultinom(n_missing,1,prob = pmf)
    d[missing_indicator,col] = apply(sample*(1:nrow(sample)),
                                     MARGIN = 2, FUN = sum)
  }
  # return imputed dataset
  return(d)
}
```

```
# do random imputation 5 times
d1 = random_imputation(df_observed)
d2 = random_imputation(df_observed)
d3 = random_imputation(df_observed)
d4 = random_imputation(df_observed)
d5 = random_imputation(df_observed)
imputed_df = rbind(d1, d2, d3, d4, d5)
```

**MICE: VEH**

Assess bivariate joint distribution

Assess bivariate joint distribution

```
# calculate rmse
numeric_df = sapply(df, as.numeric)
normalized_df = t(t(numeric_df-1)/(apply(numeric_df, MARGIN = 2, FUN = max)-1))
numeric_impute = sapply(d1, as.numeric)
normalized_impute = t(t(numeric_impute-1)/(apply(numeric_df, MARGIN = 2, FUN = max)-1))
```

```
missing_matrix = is.na(df_observed)
```

```
rmse = sqrt(sum((normalized_df[missing_matrix] - normalized_impute[missing_matrix])^2)/sum(missing_matrix))
rmse
```

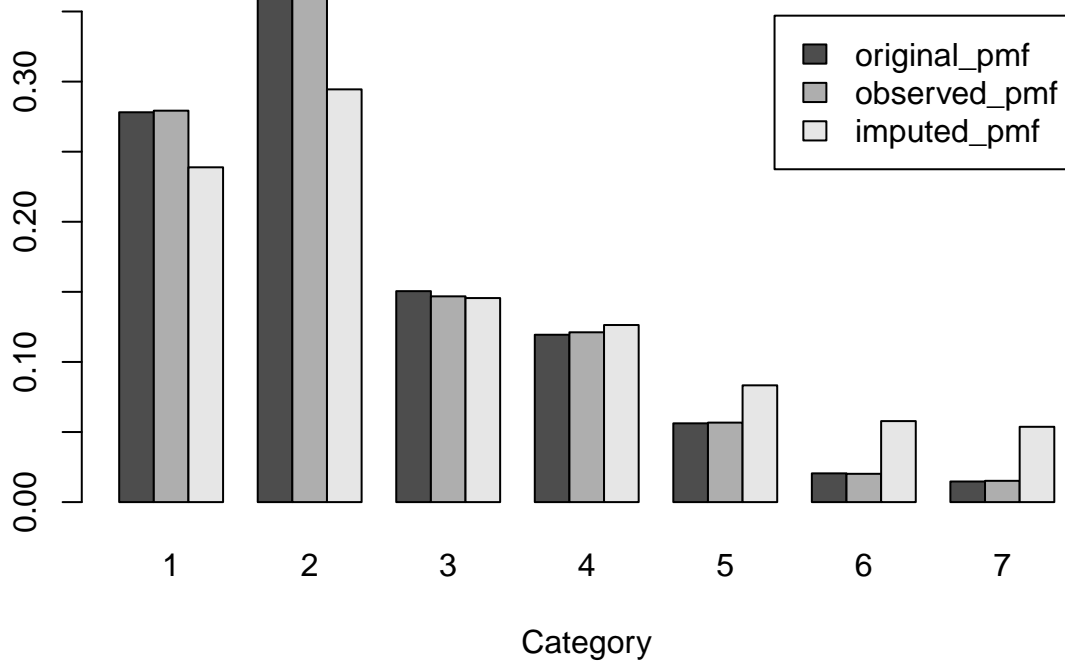
```
## [1] 0.4511768
```

```
# accuracy
```

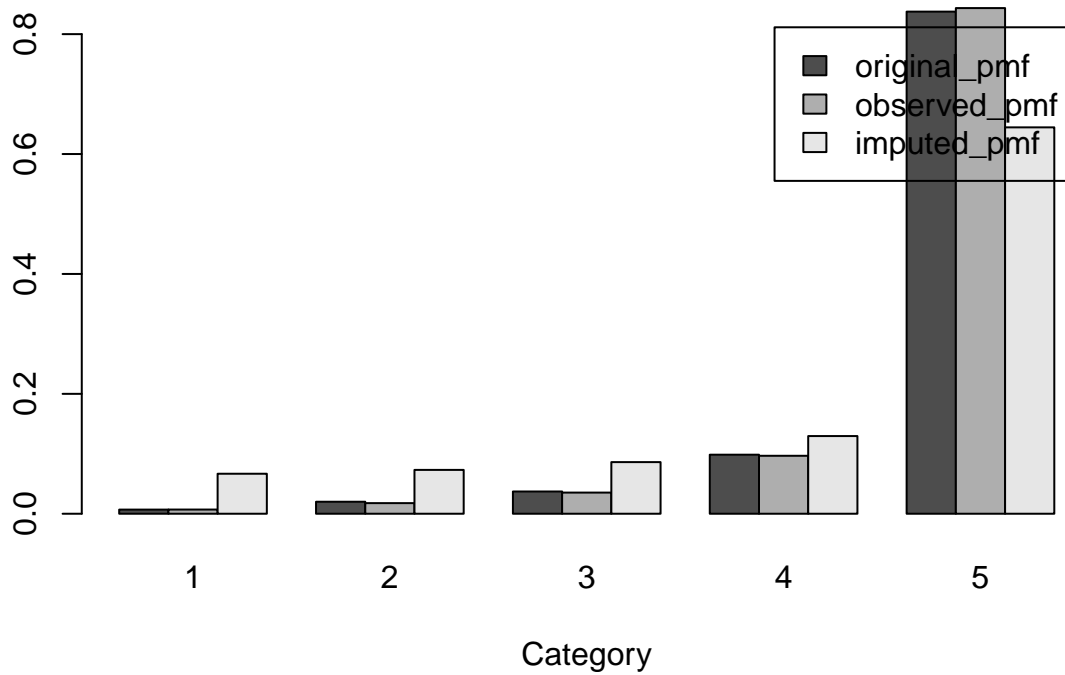
```
acc = sum(numeric_df[missing_matrix] == numeric_impute[missing_matrix])/sum(missing_matrix)
acc
```

```
## [1] 0.1302115
```

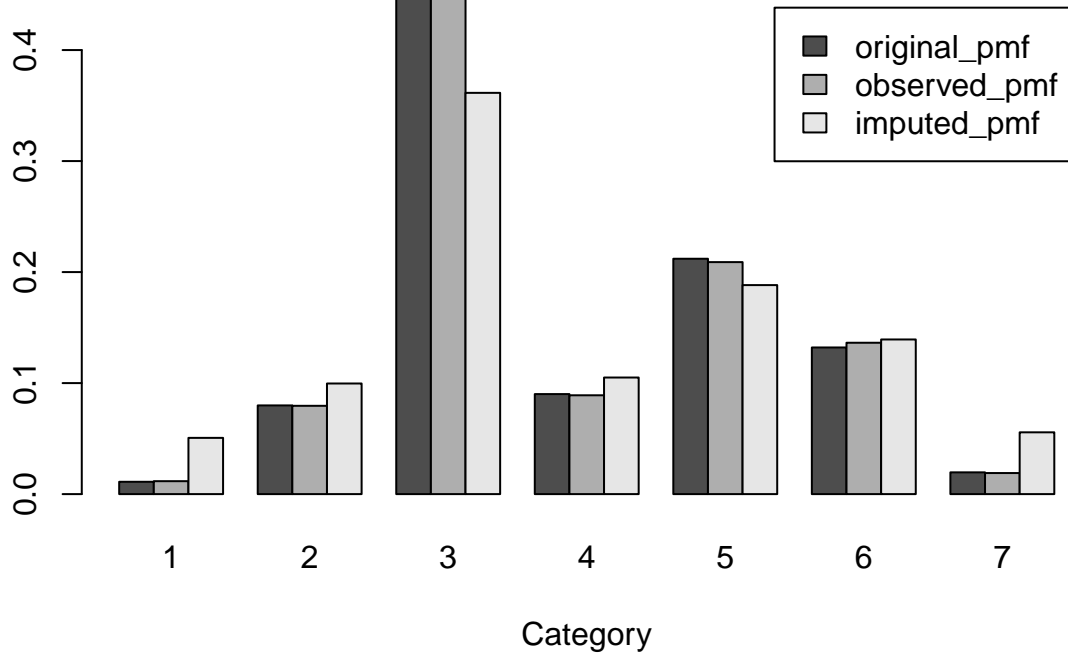
### MICE: NP



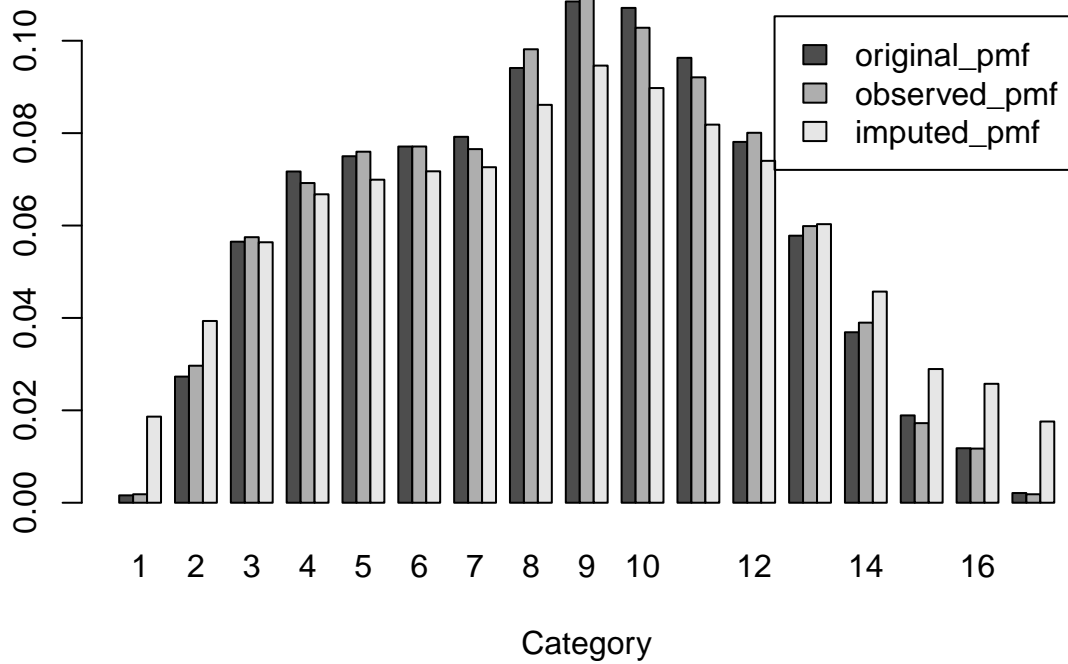
### MICE: ENG



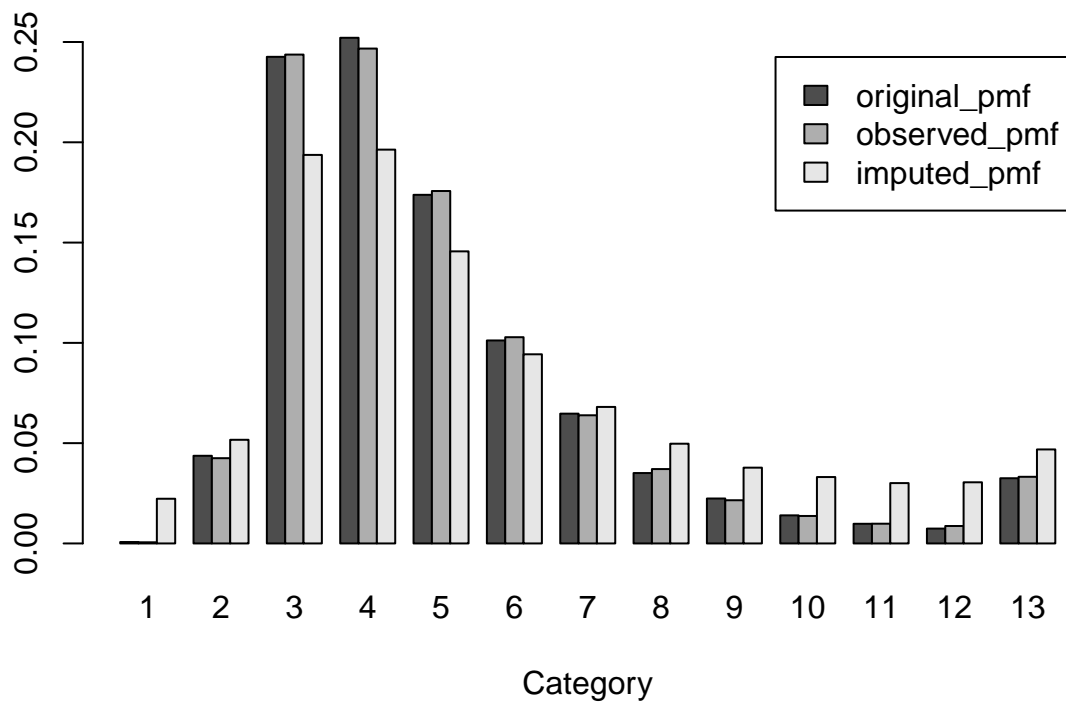
### MICE: SCHL



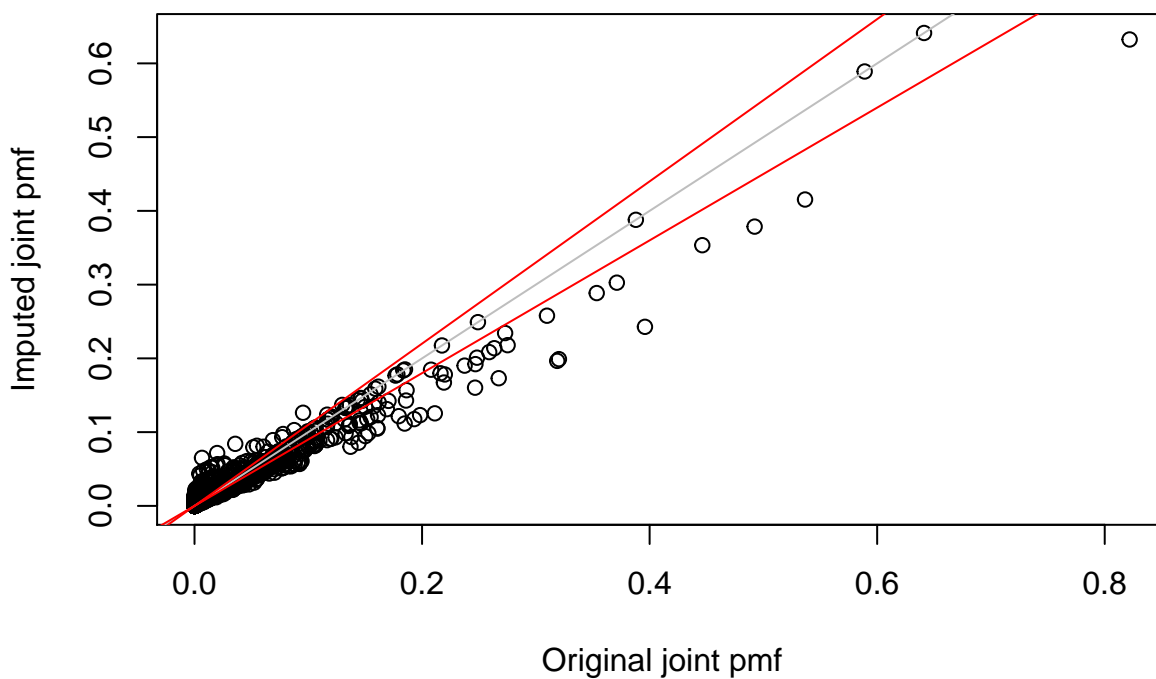
### MICE: AGEP



### MICE: PINCP



### Bivariate pmf , r square: 0.979



**Trivariate pmf , r square: 0.969**

