Reproducing top-k experiment from ARSIA paper

```
library(BayesMallows)
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.4.4     v tibble    3.2.1
v lubridate 1.9.3     v tidyr     1.3.1
v purrr     1.0.2
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becor
```

```
library(patchwork)
library(furrr)
```

```
Loading required package: future
```

```
library(extraDistr)
```

```
Attaching package: 'extraDistr'

The following object is masked from 'package:purrr':

    rdunif
```

```
sessionInfo()
```

```
R version 4.3.2 (2023-10-31)
Platform: aarch64-apple-darwin20 (64-bit)
Running under: macOS Sonoma 14.3.1

Matrix products: default
BLAS:   /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/lib/libRblas.0.dylib
LAPACK: /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/lib/libRlapack.dylib;  L

locale:
[1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8

time zone: Europe/Oslo
tzcode source: internal

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
 [1] extraDistr_1.10.0       furrr_0.3.1             future_1.33.1
 [4] patchwork_1.2.0         lubridate_1.9.3         forcats_1.0.0
 [7] stringr_1.5.1           dplyr_1.1.4             purrr_1.0.2
[10] readr_2.1.5             tidyr_1.3.1             tibble_3.2.1
[13] ggplot2_3.4.4           tidyverse_2.0.0         BayesMallows_2.0.1.9003

loaded via a namespace (and not attached):
 [1] utf8_1.2.4        generics_0.1.3    stringi_1.8.3     listenv_0.9.1
 [5] hms_1.1.3         digest_0.6.34     magrittr_2.0.3    evaluate_0.23
 [9] grid_4.3.2        timechange_0.3.0  fastmap_1.1.1     jsonlite_1.8.8
[13] fansi_1.0.6       scales_1.3.0      codetools_0.2-19  Rdpack_2.6
[17] cli_3.6.2         rlang_1.1.3       rbibutils_2.2.16  parallelly_1.36.0
[21] munsell_0.5.0     withr_3.0.0       yaml_2.3.8        tools_4.3.2
[25] parallel_4.3.2    tzdb_0.4.0        colorspace_2.1-0  globals_0.16.2
[29] vctrs_0.6.5       R6_2.5.1          lifecycle_1.0.4   pkgconfig_2.0.3
[33] pillar_1.9.0      gtable_0.3.4      glue_1.7.0        Rcpp_1.0.12
[37] xfun_0.42         tidyselect_1.2.0  rstudioapi_0.15.0 knitr_1.45
[41] htmltools_0.5.7   rmarkdown_2.25    compiler_4.3.2
```

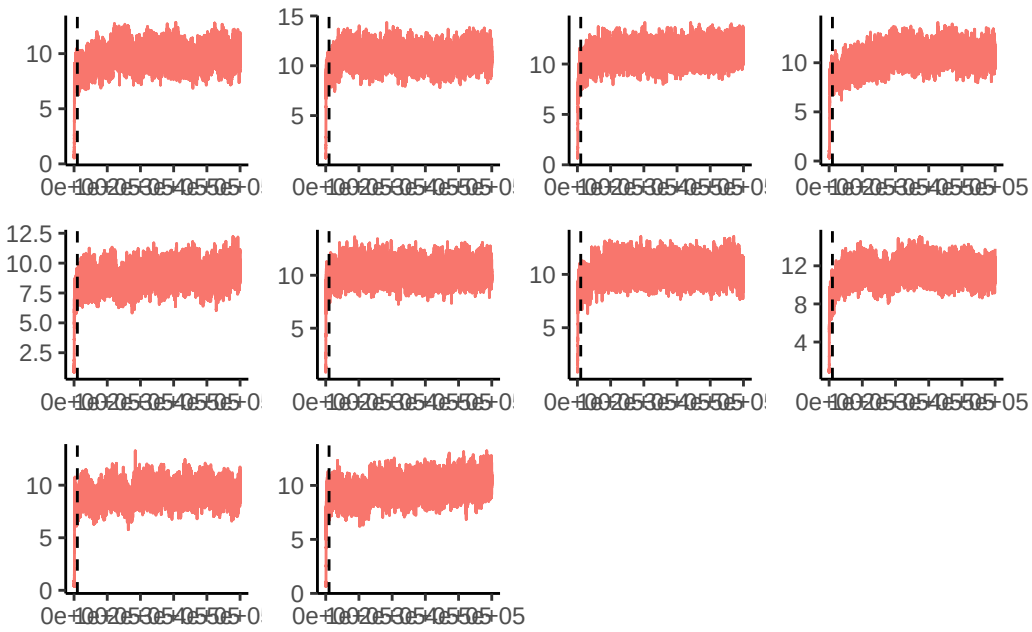
Simulations

```
plan(multisession)
models <- future_map(1:10, function(i) {
  k <- rtpois(12, lambda = 7, a = 1, b = 17)
  dat <- potato_visual
  colnames(dat) <- LETTERS[seq_len(ncol(dat))]
  for(i in seq_along(k)) {
    dat[i, ][dat[i, ] > k[[i]]] <- NA
  }
  compute_mallows(
    data = setup_rank_data(rankings = dat),
    compute_options = set_compute_options(
      nmc = 5e5, burnin = 1e4, alpha_jump = 10, rho_thinning = 10,
      aug_thinning = 10),
    priors = set_priors(lambda = .1)
  )
}, .options = furrr_options(seed = 123L))
plan("default")
```

Trace plots

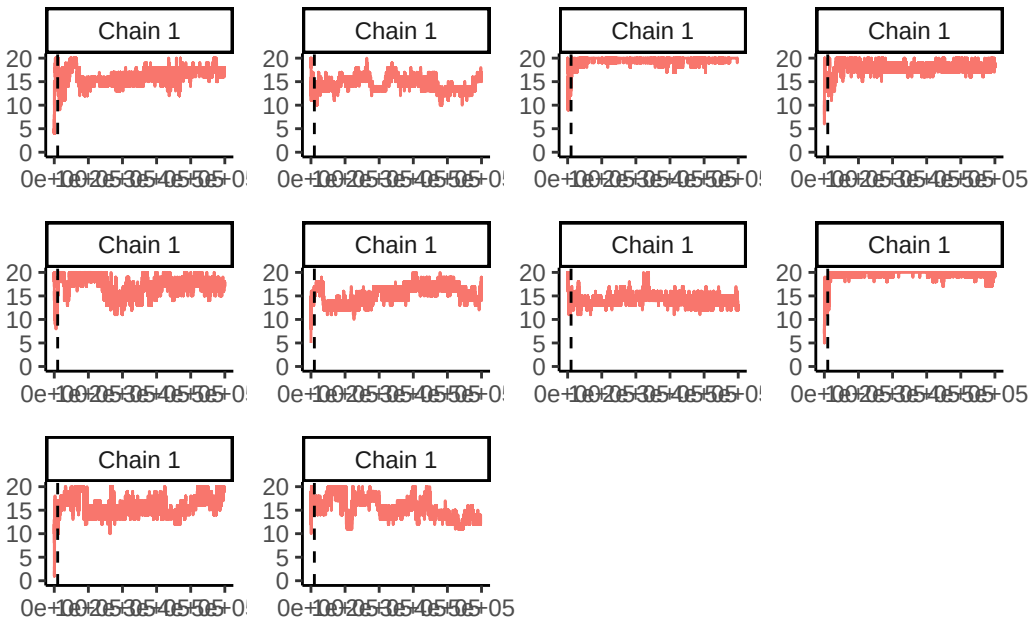
The dashed black lines shows the burn-in. This looks like it converges well in the α.

```
map(models, function(m) {
  assess_convergence(m) +
    geom_vline(xintercept = 1e4, linetype = "dashed") +
    theme_classic() +
    theme(legend.position = "none", axis.title.x = element_blank(),
          axis.title.y = element_blank())
}) %>%
  wrap_plots()
```



Potato 8 is ranked as the 20th by everyone except assessor 8, who ranks it 17th. However, this does not mean that potato 8 has to be ranked last, since in each simulated dataset there are many other potatoes that may also never be ranked, depending on the k_j. Anyhow, these trace plots look reasonable to me.

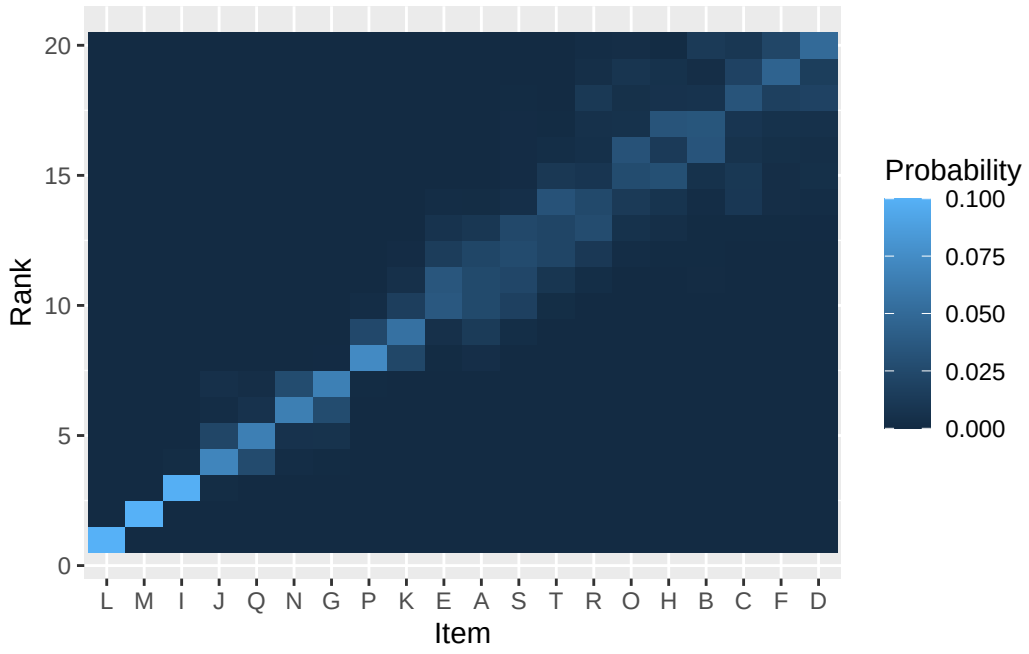
```
map(models, function(m) {
  assess_convergence(m, parameter = "rho", items = 8) +
    geom_vline(xintercept = 1e4, linetype = "dashed") +
    theme_classic() +
    ylim(0, 20) +
    theme(legend.position = "none", axis.title.x = element_blank(),
          axis.title.y = element_blank())
}) %>%
  wrap_plots()
```

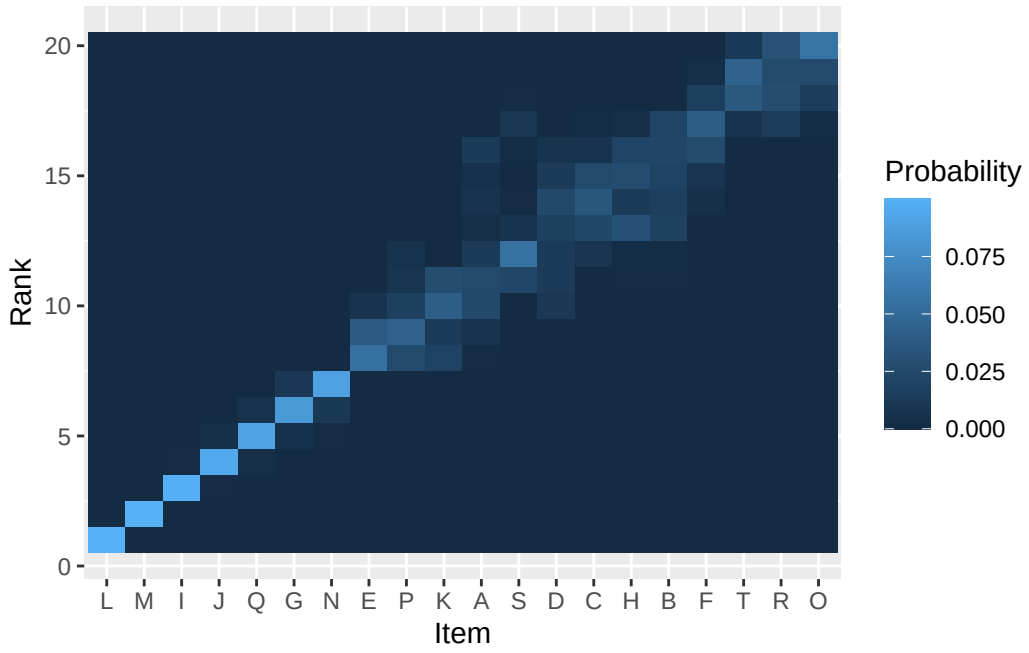


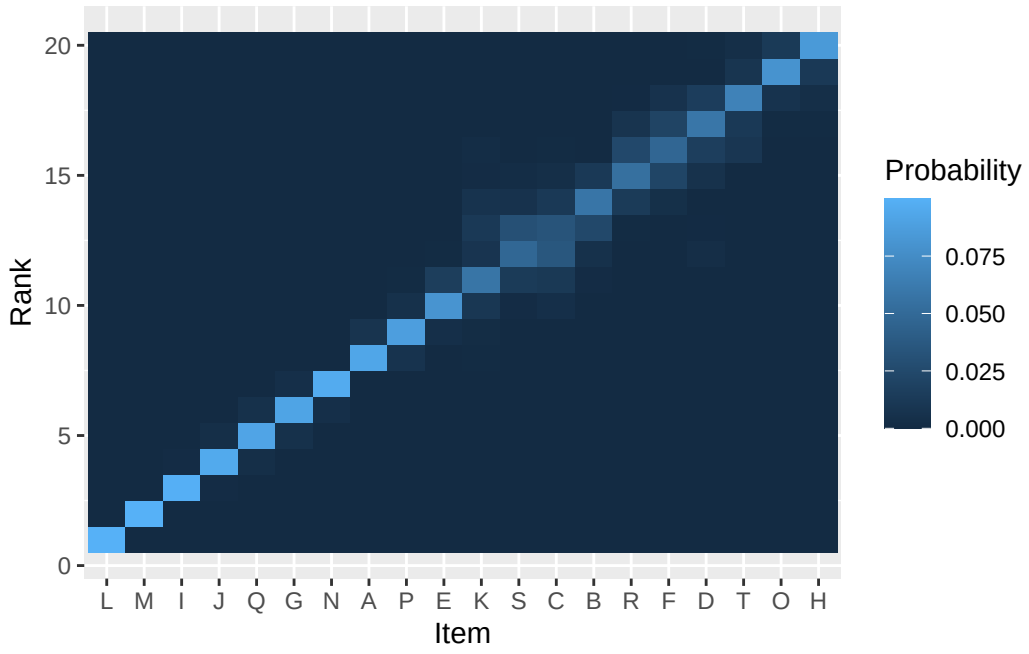
Heat plots

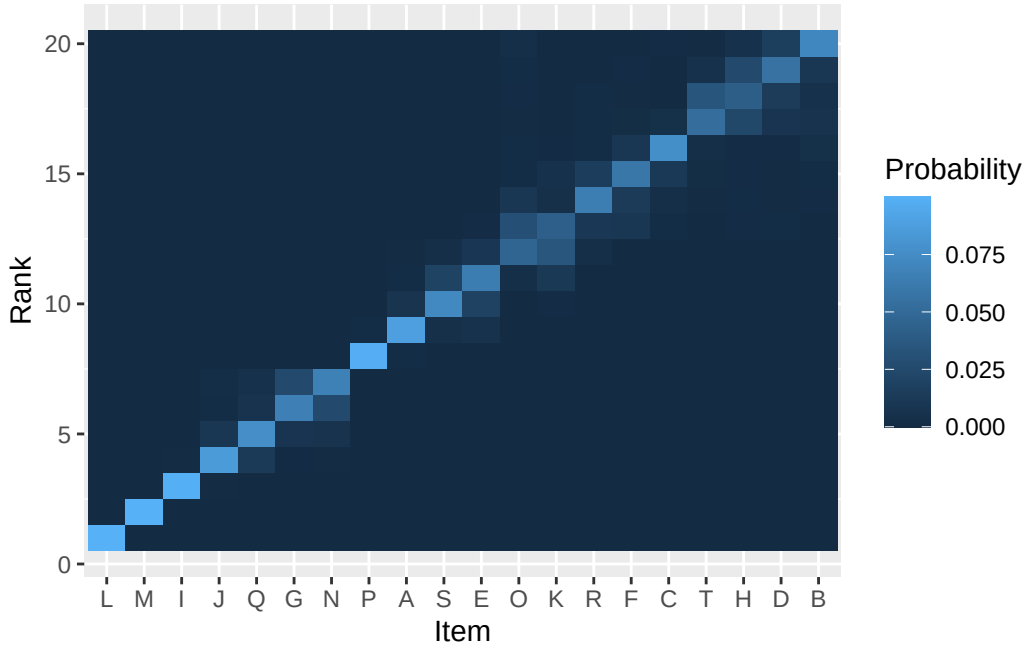
I'll show the heat plots one-by-one. They all look reasonable to me. We cannot hope to exactly reproduce ARSIA, since Figure 3a in the paper shows the result of just one random simulation. Note that the color scale differs between the papers.

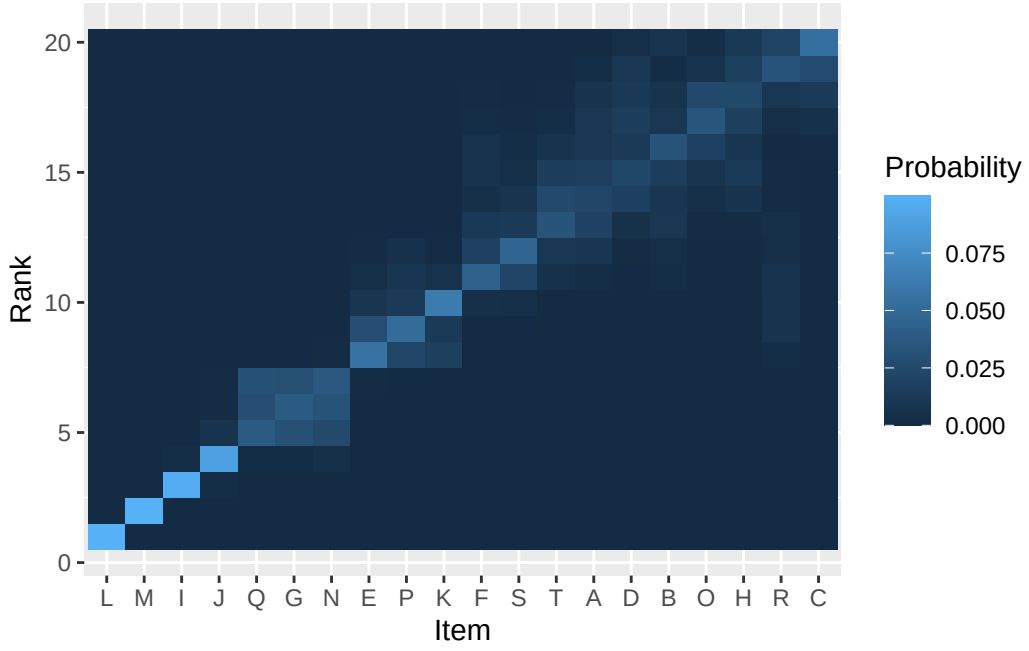
```
for(i in seq_along(models)) {
  print(heat_plot(models[[i]]))
}
```

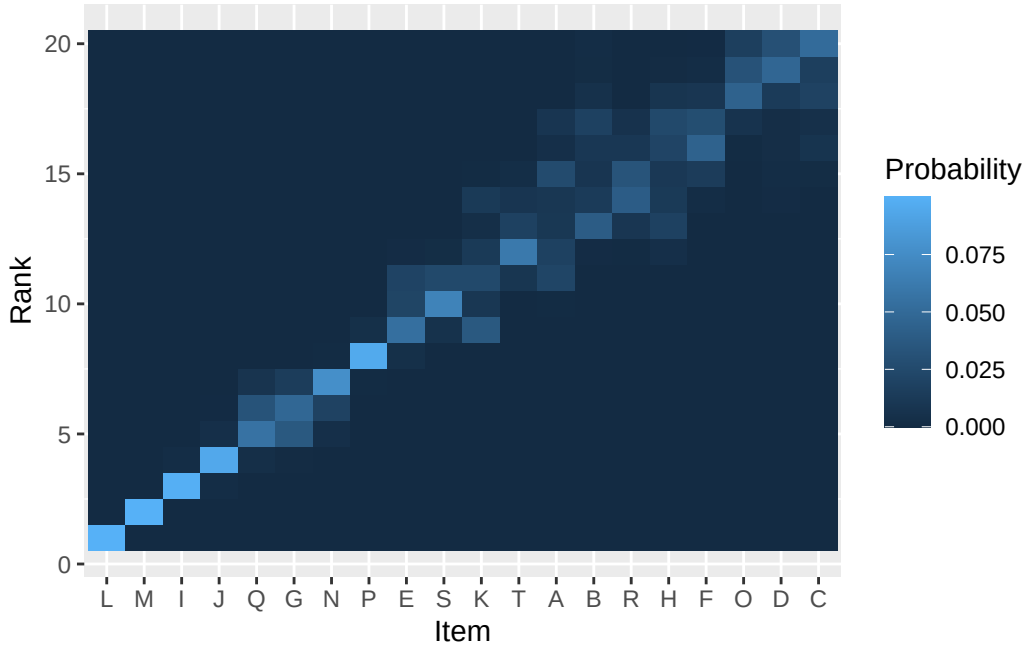

Rank
20
15
10
5
0
L M I J Q N G P K E A S T R O H B C F D
Item
Probability
0.100
0.075
0.050
0.025
0.000

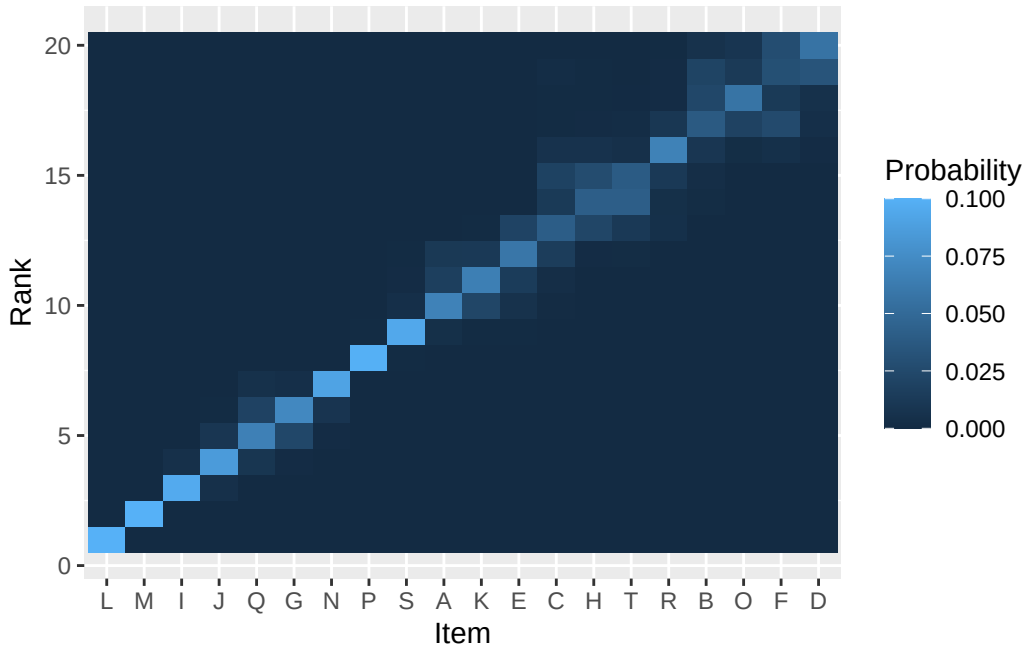

Rank
20
15
10
5
0
L M I J Q G N E P K A S D C H B F T R O
Item
Probability
0.075
0.050
0.025
0.000

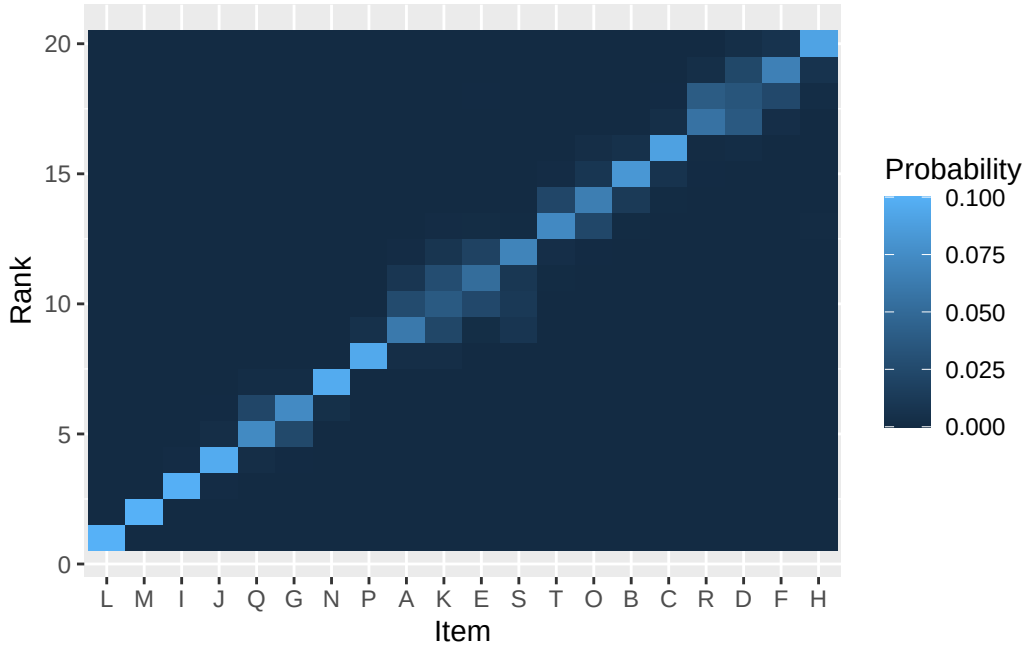

Rank
20
15
10
5
0
L M I J Q G N A P E K S C B R F D T O H
Item
Probability
0.075
0.050
0.025
0.000

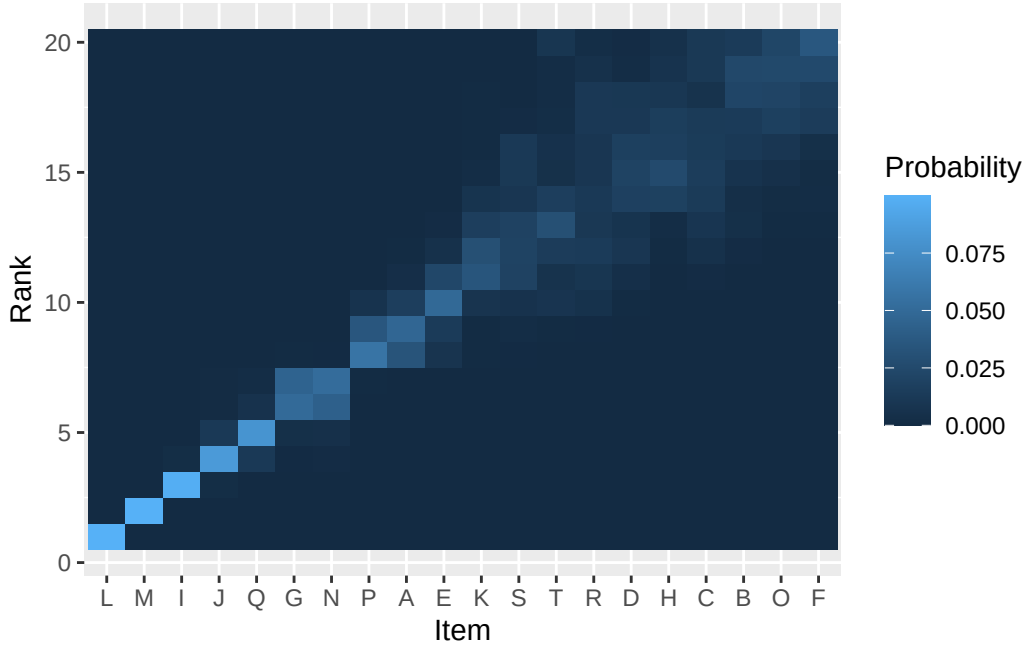

Rank
20
15
10
5
0
L M I J Q G N P A S E O K R F C T H D B
Item
Probability
0.075
0.050
0.025
0.000

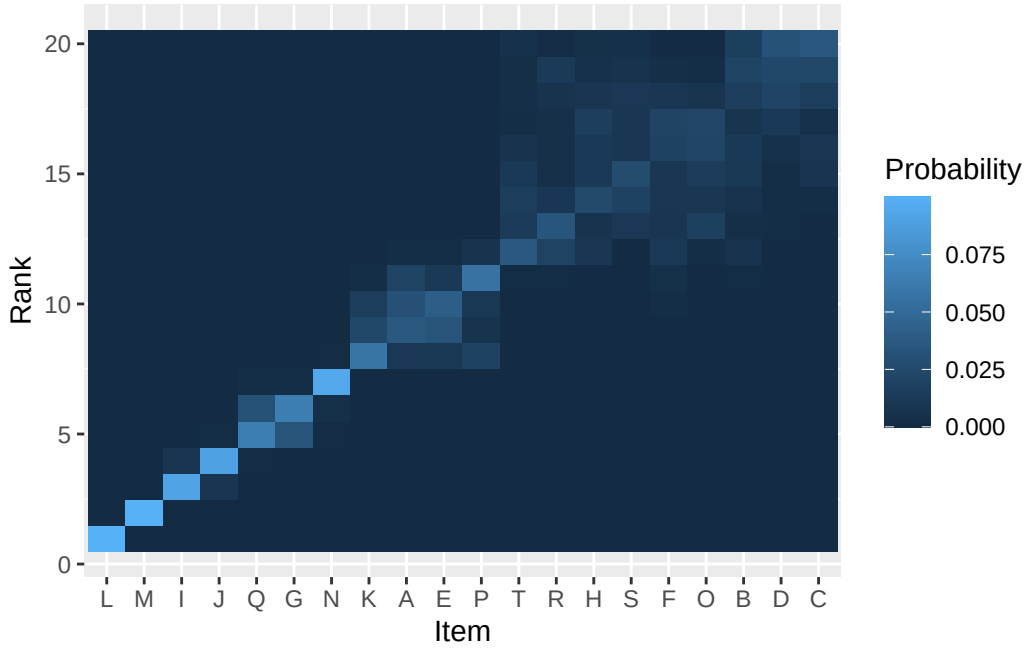

Rank
20
15
10
5
0
L M I J Q G N E P K F S T A D B O H R C
Item
Probability
0.075
0.050
0.025
0.000


Rank
20
15
10
5
0
L M I J Q G N P E S K T A B R H F O D C
Item
Probability
0.075
0.050
0.025
0.000


Rank
20
15
10
5
0
L M I J Q G N P S A K E C H T R B O F D
Item
Probability
0.100
0.075
0.050
0.025
0.000


Rank
20
15
10
5
0
L M I J Q G N P A K E S T O B C R D F H
Item
Probability
0.100
0.075
0.050
0.025
0.000





Mean Squared Error

Again I cannot exactly reproduce the random number seeds etc used in the ARSIA paper, but the order of magnitude here is very close to Figure 3c.

```r
mse <- map_dbl(models, function(m) {
  mean((potato_true_ranking - create_ranking(match(compute_consensus(m)$item, LETTERS)))^2
})

plot(mse, xlab = "run", ylab = "MSE", ylim = c(0, max(mse)))
```