Handwriting Analysis with Toy Data - Empty cluster

2022-09-28

Toy Datasets

Let’s use handwritten documents from the CSAFE Handwriting Database to perform handwriting analysis with the handwriter package. The handwriter package contains three toy datasets: one for creating a clustering template; one for training the Bayesian hierarchical model; and one for questioned documents.

- The template training images are scans of handwritten prompts from 10 randomly selected writers from the CSAFE Handwriting Database. The prompt from each writer is the London Letter prompt from the first session and the first repetition.
- The model training images are scans of handwritten prompts from 5 randomly selected CSAFE writers. For each writer, the prompts are the 3 Wizard of Oz prompts from the first session. These 5 writers are distinct from the 10 template training writers.
- The questioned images are scans of the Wizard of Oz prompt from the first session and the first repetition from the 5 model training writers.

These images are located in the handwriter package in the following folders:

- /Library/Frameworks/R.framework/Versions/4.1-arm64/Resources/library/handwriter/extdata/example_images/tem
- /Library/Frameworks/R.framework/Versions/4.1-arm64/Resources/library/handwriter/extdata/example_images/moc
- /Library/Frameworks/R.framework/Versions/4.1-arm64/Resources/library/handwriter/extdata/example_images/tem

Create a Clustering Template

Start by creating a new directory that will hold the handwriting analysis data.

```
# path to main directory
main_dir <- "/Users/stephanie/Documents/non_version_control/CSAFE_toy_datasets_empty_cluster"

# create folder if it doesn't already exist
if (!dir.exists(main_dir)){dir.create(main_dir)}
```

Process the handwriting images with [`process_batch_dir()`]. Specify where to save the processed handwriting. I like to save them in a subfolder of the main directory. The [`process_batch_dir()`] function will create this folder automatically if it doesn’t already exist.

```
# get the path to the handwriter image directory
template_images_dir <- system.file("extdata/example_images/template_training_images", package = "handwri

# choose where to save the processed
template_graphs_dir <- file.path(main_dir, "data", "template_graphs")
```

```
# process the handwriting
# process_batch_dir(image_batch = template_images_dir,
#                   batch_output_dir = template_graphs_dir,
#                   transform_output = 'document')
```

Create a new clustering template from the processed template training images. The new template will be saved in a subdirectory of the main directory.

```
# template <- make_clustering_templates(template_dir = main_dir,
#                                       K = 5,
#                                       num_dist_cores = 4,
#                                       max_iters = 3,
#                                       num_graphs = 1000,
#                                       num_runs = 1,
#                                       starting_seed = 200)
```

Fit the Full Bayesian Hierarchical Model

Process the handwriting images with [`process_batch_dir()`]. Specify where to save the processed handwriting. I like to save them in a subfolder of the main directory. The [`process_batch_dir()`] function will create this folder automatically if it doesn't already exist.

```
# get the path to the handwriter image directory
model_images_dir <- system.file("extdata/example_images/model_training_images", package = "handwriter")

# choose where to save the processed handwriting
model_graphs_dir <- file.path(main_dir, "data", "model_graphs")
```

```
# process the handwriting
# process_batch_dir(image_batch = model_images_dir,
#                   batch_output_dir = model_graphs_dir,
#                   transform_output = 'document')
```

Assign each graph from the processed handwriting to a cluster in the clustering template. The output of [`make_clustering_templates()`] is a list of template(s). We access the first, and only in this case, template in the list with template[[1]].

```
template <- readRDS(file.path(main_dir, "template_seed200/data/all_templates.rds"))

m_proc_list <- get_clusterassignment(clustertemplate = template[[1]], input_dir = model_graphs_dir)
```

```
## 0.07
## 0.13
## 0.2
## 0.27
## 0.33
## 0.4
## 0.47
## 0.53
## 0.6
## 0.67
## 0.73
## 0.8
## 0.87
## 0.93
## 1
```

```
saveRDS(m_proc_list, file.path(main_dir, "template_seed200/seed200_run1/data/model_cluster_proc_list.rd
```

Get the data model training data ready for the model. The model needs to know which writer wrote which document. We get this information for the image file names. Each file name has the format w0001_s01_pLND_r01. The writer ID starts at character 2 and ends at character 5 so we input writer_indices=c(2,5). The model also asks for a name for each document to distinguish between documents written by the same writer. We use s01_pLND_r01 for the document name so we enter doc_indices=c(7,18). We could use the entire file name as the document name if we want. Lastly, we choose values for the model parameters: a=2, b=0.25, c=2, d=2, and e=0.5.

```
md <- format_model_data(proc_list=m_proc_list,
                        writer_indices=c(2,5),
                        doc_indices=c(7,18),
                        a=2, b=0.25, c=2, d=2, e=0.5)
```

Fit the Bayesian hierarchical model and drop burn-in.

```
draws <- fit_model(md, num_iters = 4000)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 2910
##    Unobserved stochastic nodes: 53
##    Total graph size: 37057
##
## Initializing model
```

```
draws <- drop_burnin(draws, 1000)
```

Analyze Questioned Documents

Process the questioned documents.

```
# get the path to the handwriter image directory
questioned_images_dir <- system.file("extdata/example_images/questioned_images", package = "handwriter"

# choose where to save the processed handwriting
questioned_graphs_dir <- file.path(main_dir, "data", "questioned_graphs")

# process the handwriting
# process_batch_dir(image_batch = questioned_images_dir,
#                   batch_output_dir = questioned_graphs_dir,
#                   transform_output = 'document')
```

Get the cluster assignments for the graphs in the questioned documents.

```
q_proc_list <- get_clusterassignment(clustertemplate = template[[1]], input_dir = questioned_graphs_dir
```

```
## 0.2
## 0.4
## 0.6
## 0.8
## 1
```

```
saveRDS(q_proc_list, file.path(main_dir, "template_seed200/seed200_run1/data/questioned_cluster_proc_li
```

Format the questioned document data so that it can be used with the model.

```
qd <- format_questioned_data(proc_list = example_questioned_proc_list,
                             writer_indices=c(2,5),
                             doc_indices=c(7,18))
```

Use the fitted model to analyze the questioned documents and plot the posterior probabilities of writership.

```
# analysis <- analyze_questioned_documents(md, draws, qd, num_cores = 4)

# plot_posterior_probabilities(analysis)
```