

Collaboratively Setting Daily Step Goals with a Virtual Coach: Using Reinforcement Learning to Personalize Initial Proposals

This repository is meant to show and reproduce our analyses for the paper "Collaboratively Setting Daily Step Goals with a Virtual Coach: Using Reinforcement Learning to Personalize Initial Proposals". The third author checked the analysis code and data and filled in a checklist (ChecklistDataRepositoryReview.pdf).

Authored by Martin Dierikx, Nele Albers, Bouke L. Scheltinga, and Willem-Paul Brinkman.

Contents of this Repository

```
| ChecklistDataRepositoryReview.pdf
| Dockerfile
| Dockerfile_r
| JAGS-4.3.2.tar.gz
| README.md/README.pdf
|
└──analysis_code
|   | Database data preprocessing.ipynb
|   | Results, values, and figures.ipynb
|
└──bayesian_analysis_steps
|   | bayesian_analysis_steps.pdf
|   | bayesian_analysis_steps.Rmd
|   | preprocessed_step_data.csv
|   | references.bib
|
└──data
|   | cleaned_database_data.csv
|   | preprocessed_database_data.csv
|   | preprocessed_demographic_data.csv
|   | preprocessed_post_questionnaire_data.csv
|   | preprocessed_pre_screening_data.csv
|   | README.md
|
└──results
|   | Database data preprocessing.html
|   | Results, values, and figures.html
|
└──scripts
|   | Pre-processing.py
```

File/Folder name	Content
ChecklistDataRepositoryReview.pdf	Checklist reviewing the analysis code and data.

Dockerfile	File containing instructions to build Docker image for analyses in jupyter notebook.
Dockerfile_r	File containing instructions to build Docker image for Bayesian analysis of steps in R.
JAGS-4.3.2.tar.gz	JAGS needed for building the Docker image for the Bayesian analysis of steps in R.
README.md/README.pdf	The file you are currently reading - provides an overview of the repository and how to reproduce our analyses.
analysis_code/	Folder containing the jupyter notebooks with our analysis code.
- Database data preprocessing.ipynb	Code for reproducing the preprocessing of the database data file and step data file.
- Results, values, and figures.ipynb	Code for reproducing all the results, values and figures in our paper.
bayesian_analysis_steps/	Folder containing the files needed for the Bayesian analysis of steps in R.
- bayesian_analysis_steps.pdf	Knitted .Rmd-file for the Bayesian analysis of steps in R.
- bayesian_analysis_steps.Rmd	.Rmd-file for running our Bayesian analysis of steps in R.
- preprocessed_step_data.csv	Preprocessed data needed for the Bayesian analysis of steps.
- references.bib	References used in the .Rmd-file, needed for knitting the .Rmd-file.
data/	Folder containing preprocessed data from our observational study used in our analyses.
- cleaned_database_data.csv	Cleaned and anonymized data from the database of the conversation sessions from the study.
- preprocessed_database_data.csv	Preprocessed data from the database of the conversation sessions from the study.
- preprocessed_demographic_data.csv	Preprocessed data from Prolific on demographics from the participants of the study.
-	Preprocessed data from the post-

preprocessed_post_questionnaire_data.csv	questionnaire of the study.
- preprocessed_pre_screening_data.csv	Preprocessed data from the pre-screening of the study.
- README.md	File with an overview and explanation of all the cells in the data files.
results/	Folder containing html files created from the jupyter notebooks with the results after running them.
- Database data preprocessing.html	Html file from running Database data preprocessing.ipynb.
- Results, values, and figures.html	Html file from running Results, values, and figures.ipynb.
scripts/	Folder containing Python scripts to preprocess the raw data files.
- Pre-processing.py	Preprocessing script to clean up, anonymize and organize data.

Reproducing our Analyses

Initial Setup

Make sure that you have Docker installed. You can check whether you do by running `docker -v`. In case you do not have Docker installed, you can follow the instructions [here](#).

Python-based Analysis

All our results except for the Bayesian analysis of steps can be reproduced using Python.

Obtaining the Docker Image

You can build the Docker image in the directory of this README-file. Build the Docker image via `docker build . -t <DOCKER_IMAGE_NAME>`.

Replace the `<DOCKER_IMAGE_NAME>` with your chosen name of the docker image. For example, if you want to name your docker image `data_analysis`, your docker command would be `docker build . -t data_analysis`.

More information regarding Dockerfile and how to build images can be found [here](#).

Running the Docker Container

Run the Docker container via `docker run -p 8888:8888 -e JUPYTER_ENABLE_LAB=yes -v <PATH_TO_THIS_DIRECTORY>:/home/jovyan/work <DOCKER_IMAGE_NAME>`. Replace `<PATH_TO_THIS_DIRECTORY>` with the path to the directory that this README-file is in and replace with the name you chose for the image in the previous step.

If you are running a container where the actual path to your current folder is, for instance, `/home/users/documents`, and the name for the docker image you chose was, for

example, `data_analysis`, the command should look something like this: `docker run -p 8888:8888 -e JUPYTER_ENABLE_LAB=yes -v /home/users/documents:/home/jovyan/work data_analysis`

Running the Analyses

When the container is running:

1. Go to one of the links presented in the terminal to access Jupyter Notebook.
2. Open the `work` -folder in Jupyter Notebook
3. Open the `analysis_code` folder in Jupyter Notebook.
4. Choose the `.ipynb` file that you want to run and reproduce the analysis.

If the `work` -folder is empty, check that you have used the correct path in the command for running the Docker container. Also, sharing the folder content may not work if you use a drive other than the C drive.

Results

The `results` -folder contains the `.html` files from running the Jupyter Notebook files. These can be obtained by running one of the `.ipynb` files and downloading them as `.html` file.

Tables/Figures and Corresponding Files

All the results, values, and figures in the paper, except for the Bayesian analysis of steps, can be reproduced using the `Results, values, and figures.ipynb` file. The file contains information on which results can be found where in the file and comments on what the code is doing.

R-based Analysis of Steps

The Bayesian analysis of steps reported in the `Data Collection and Model Training - Collected data` -section in the paper can be reproduced using R.

The reproduction of our code is based on Docker and R Studio. Take the following steps:

1. Navigate to the folder this README-file is in.
2. Now you have 2 options:
 - Build the Docker image via `docker build . -f Dockerfile_r -t gbna4/rl_daily_goals_r`, or
 - Pull the Docker image from Dockerhub via `docker pull gbna4/rl_daily_goals_r`.
3. Run the Docker container via `docker run -d -p 8787:8787 -v <path_to_this_directory>:/home/rstudio/analysis -e PASSWORD=<some_password> gbna4/rl_daily_goals_r`.
4. Go to `localhost:8787`.
5. Login with username 'rstudio' and the password chosen in step 3.
6. Navigate to the `analysis/bayesian_analysis_steps` -folder in R Studio. If the `analysis` -folder is empty, check that you have used the correct path in the command for running the Docker container. Also, sharing the folder content may not work if you use a drive other than the C drive.
7. Now you can run the analysis using the `bayesian_analysis_steps.Rmd` -workbook. To knit a pdf-file from the Rmd-file, you can use the `Knit` -button in R Studio.

Data Preprocessing

The `scripts` -folder contains the Python scripts used to preprocess the data files. Since the raw data contained Prolific ids and other fields we were not allowed to publish, we could not publish those files. Therefore we only published the preprocessed data files. The Python scripts contain notes on how the raw data files looked like and what steps were taken to anonymize, clean, and order the data. For the database data, however, we published the cleaned and anonymized data file called `cleaned_database_data.csv` . The Jupyter notebook file called `Database data preprocessing.ipynb` processes the cleaned database data and creates the `preprocessed_database_data_reproduced.csv` in the `results` folder. Therefore, it can be used to reproduce the processed database data file and can be compared with our provided `preprocessed_database_data.csv` to check the reproducibility. Also, for the data used in the R script to calculate the differences in steps before and after the intervention, the data file `preprocessed_step_data.csv` can be reproduced using the code in the Jupyter notebook file called `Database data preprocessing.ipynb` . This creates the `preprocessed_step_data_reproduced.csv` in the `results` -folder.