

B | (i)#1

 To download and install docker, visit the official Docker website

(https://docs.docker.com/get-docker/).

- The windows user could use wsl (https://docs.docker.com/desktop/wsl/)
- To test the docker installation by command line: docker run hello-world

(i)#2

 To pull and run dockerHDDM (details in section 2) require terminal line basics

(https://www.freecodecamp.org/news/command-line-forbeginners/)

and basic docker commands

(https://dev.to/roselinebassey/docker-for-beginners-basic-docker-commands-2n89)

①#3

- If you are not familiar with the jupyter IDE and python code, you can try it online without installing it beforehand: https://jupyter.org/try-jupyter/retro/ notebooks/?path=notebooks/Intro.ipynb
- If you successfully run the dockerHDDM container and go to jupyter, you'll be greeted with the sight of figure5.

(i)#4

If you are not familiar with the basic code of HDDM, such as specifying and fitting model, you can refer to the jupyter notebook provided in the HDDM official tutorial folder:

- HDDM_basic_tutorial.ipynb: provides basic code examples on how to build and fitting the DDM in hierarchical structure.
- HDDM_regression_stimcoding.ipynb: shows how to allow parameters to vary with experimental conditions
- Posterior_Predictive_Checks.ipynb: demonstrates how to evaluate the predictive performance of a model.
- LAN_Tutorial.ipynb: exhibits the use of neural network methods to deal with complex models with free likelihood.

(i)#5

New features in DockerHDDM about how to use Arviz tools to implement analysis workflow, code examples stored in dockerHDDM tutorial folder:

- dockerHDDM_quick_view.ipynb: shows how to parallelize the computation of multiple MCMC chains, model loglikelihood and ppc, and return inference data for Arviz analysis (see section 3).
- dockerHDDM_workflow.ipynb: include the workflow analyzed in the article (see section 4), which serves as a reference for the DDM analysis process, mainly for analyzing the inference data returned by the fitted model.