Containers for reproducibility

Karl Broman

Biostatistics & Medical Informatics, UW–Madison

kbroman.org
github.com/kbroman
@kwbroman

Course web: kbroman.org/AdvData

Reproducible research

organize the data and code in a way
that you can hand them to someone else
and they can re-run the code
and get the same results
(the same figures and tables)

Dependency Hell

- What software does your project depend on?
  - operating system
  - system libraries
  - R or python
  - packages or modules
  - other tools (e.g. pandoc and LaTeX)
- Can you install all necessary dependencies?
- Have dependencies changed? Do you need particular versions?
- How much time does it take to set things up?

Capturing dependencies

- R: renv

  ```
  renv::init()
  renv::snapshot()
  renv::restore()
  ```

  Also see MRAN

- Python: conda

  ```
  conda create
  conda install
  conda activate
  conda env list --explicit
  ```

  Also the built-in venv

Or create package/module

- R package
  - dependencies in `DESCRIPTION` file
  - data in `inst/ext_data`
  - analyses as vignettes
- Python package
  - multiple modules, plus `__init__.py` and `setup.py`
  - define dependencies with `setuptools.setup`

Docker containers

- Light-weight virtual machine
  - Uses the host machine’s linux kernel
  - On Mac/Windows, containers run within a separate virtual machine
- Capture all dependencies, down to the OS
- Binary image with everything pre-installed, including data
- Text-based recipes for creating the image
- Can build recipe starting from some previous one

Getting started with Docker

- Download and install docker, from `docker.com`
- Get an account at `hub.docker.com`

Docker stuff

- Container

  A running docker thing
- Image

  A binary file with a snapshot of a container
- `Dockerfile`

  Text file with recipe to create a new container

Rocker images

- Docker containers for R
- Can run locally, and have RStudio in the web browser
- Poke around:
  - hub.docker.com/u/rocker
  - rocker-project.org
  - github.com/rocker-org

```
docker pull rocker/rstudio

docker run -e PASSWORD=[blah] -p 8787:8787 rocker/rstudio

  -v $(pwd):/home/rstudio
```

Jupyter images

- Docker containers set up for Jupyter notebooks
- Look at hub.docker.com/u/jupyter

```
docker pull jupyter/minimal-notebook

docker run -v $(pwd):/home/jovyan -p 8888:8888 jupyter/minimal-notebook
```

Creating a docker image

- Start from some previous image
- Use a `Dockerfile`
  - explicit
  - human-readable
  - an often-small script
- Create a container interactively and then write it to an image
  - `docker cp` to copy stuff into the container
  - `docker commit` to save a container to an image file

Creating a new docker image

```
docker run -d -e PASSWORD=rqtl --name rqtl -p 8787:8787 rocker/rstudio

install.packages("qtl")
download.file("https://rqtl.org/sug.csv", "sug.csv")

docker commit rqtl rstudio_rqtl

docker tag e3ae59d1443f kbroman/rstudio_rqtl:firsttry
docker login
docker push kbroman/rstudio_rqtl
```

Example Dockerfile

```
FROM java
MAINTAINER daroczig@rapporter.net

## Prepare folder for the Minecraft stuff
RUN mkdir -p /minecraft

## Download Spigot build tools
RUN wget https://hub.spigotmc.org/jenkins/job/BuildTools/[clip]/target/BuildTools.jar -P /minecraft/

## Build the Spigot server
RUN cd /minecraft && java -jar BuildTools.jar

## Symlink for the built Spigot server
RUN ln -s /minecraft/spigot*.jar /minecraft/spigot.jar

## Accept EULA
RUN echo "eula=true" > /minecraft/eula.txt

## Download and install the RaspberryJuice plugin for API access
RUN mkdir -p /minecraft/plugins \
    && wget https://github.com/zhuowei/RaspberryJuice/raw/master/jars/raspberryjuice-1.11.jar \
    && mv raspberryjuice-1.11.jar /minecraft/plugins/

## Open up API port
EXPOSE 4711
## Open up Game port
EXPOSE 25565

## Start the server
CMD cd /minecraft; java -Xms512M -Xmx1G -XX:MaxPermSize=128M -XX:+UseConcMarkSweepGC -jar spigot.jar
```

Another example

`github.com/rocker-org/rocker-versioned`

`/rstudio/latest.Dockerfile`

Managing Docker stuff

```
docker images
docker image ls

docker ps -a
docker container ls -a

docker container stop adoring_hamilton
docker container start adoring_hamilton

docker rm adoring_hamilton
docker image rm alpine
docker rm $(docker ps -a -q)
```

binder

- mybinder.org
- add two files to a github repo → docker container in the cloud
  - runtime.txt telling date of R
  - install.R with install.packages() calls
  - special url with ?urlpath=rstudio
- examples:
  - kbroman.org/blog/2019/02/18/omg_binder
  - github.com/kbroman/Teaching_CTC2019

Summary

- Want to capture the full environment for a project
  - code + data
  - dependent packages, libraries
- Want to lower the barrier to the set-up of this stuff
- Docker containers
  - portable
  - shareable
  - extendable
  - `Dockerfile` script to define
- `mybinder.org`
  - github → docker in the cloud
  - magical set-up