How to install and run WinBUGS from Stata on a Mac under Wine

Tom Palmer, 19 March 2014

I am running Mac OS X 10.9.2 but these instructions should work for other versions as long as your Mac has an Intel processor.

The approach we will take is to use the Wine (http://www.winehq.org) compatibility layer that allows many applications originally designed to run under Windows to work on Linux or Mac OS X.

- To be on the safe side start by doing a Time Machine backup of your Mac.
- Install XQuartz.app from https://xquartz.macosforge.org/landing/ . Once installed launch it.
- Install Xcode from the App Store.
- Once Xcode is installed open it and go to Preferences. On the Locations tab install Xcode Command Line Tools.
- Install Macports from http://www.macports.org/install.php
- Once Macports is installed install the wine-devel port using the Terminal command, note you will need an Admin password:
  ```
  sudo port install wine-devel
  ```
  This takes a very long time (possibly as long as 1hr on slow Macs)
- Download the WinBUGS installer from
  http://www.mrc-bsu.cam.ac.uk/bugs/winbugs/WinBUGS14.exe
  Download the version 1.4.3 patch from
  http://www.mrc-bsu.cam.ac.uk/bugs/winbugs/WinBUGS14_cumulative_patch_No3_06_08_07_RELEASE.txt
  Download the immortality key from
  http://www.mrc-bsu.cam.ac.uk/bugs/winbugs/WinBUGS14_immortality_key.txt
- Now we start the installation of WinBUGS into our Wine system with the Terminal command (assuming you are cd'd into the directory where the WinBUGS14.exe installer is saved). XQuartz must be open for this to work.
  ```
  wine WinBUGS14.exe
  ```
  Follow the onscreen installer instructions to install WinBUGS.
- Now we must install the key and then patch into WinBUGS. Start by launching WinBUGS using the following Terminal command (if you installed WinBUGS into the default location)
  ```
  wine "C:/Program Files/WinBUGS14/WinBUGS14.exe"
  ```
  At this point you may be wondering how that command worked since Mac OS X uses Unix file path conventions (i.e. there is no C:/ drive on a Mac). Well the C:/ drive is really a pointer inside our Wine installation. When installing Wine if you used the default location the C:/ drive is at (replace username with your username):
  ```
  /Users/username/.wine/drive_c
  ```
  so you could also issue the following to launch WinBUGS
  ```
  wine "/Users/tompalmer/.wine/drive_c/Program
  Files/WinBUGS14/WinBUGS14.exe"
  ```

Now install the WinBUGS key as you would usually. Note the rest of your Mac harddrive is assigned as the Z:/ drive under Wine. This is done by opening the key file in WinBUGS then clicking: Tools | Decode All

After installing the key relaunch WinBUGS and install the patch file in the same way.

- Now we are ready to use WinBUGS. The important command when using the winbugsfromstata commands is to replace the wbrun command with our own shell command. You will typically need a shell command that looks like the following (assuming your WinBUGS script file is saved on your Desktop):

```
shell /opt/local/bin/wine "C:/Program
Files/WinBUGS14/WinBUGS14.exe" /par
"Z:/Users/username/Desktop/script.txt"
```

It is very important to specify the full filepath to the script file, i.e. usual shortenings such as ~/Desktop/script.txt do not work.

Also within your script file you must code the full filepath to all files referred to such as the model file, initial values file, data file, coda file, etc. i.e. code things like:

```
check('Z:/Users/username/Desktop/model.txt')
```

If you want to suppress the Wine error message simply prefix the above with `quietly` or `capture`.

- And that should be it. Here's a picture for proof:
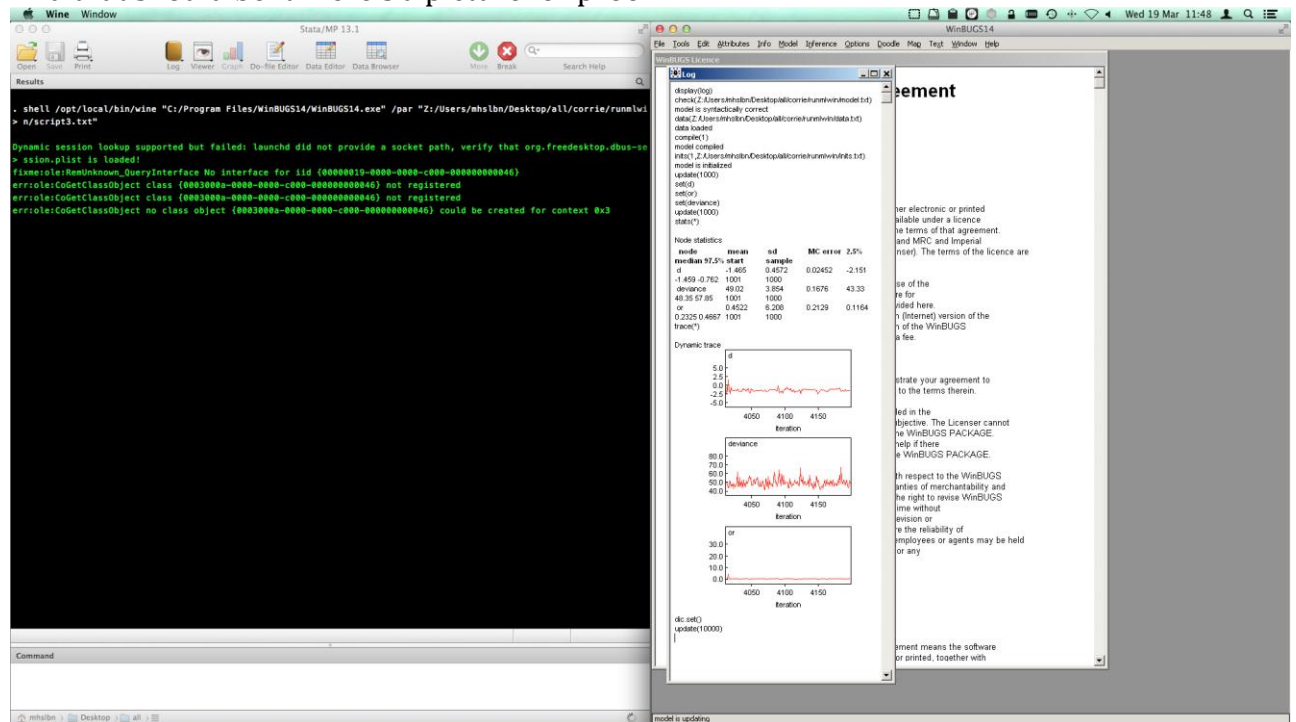
- The same principle should work on Linux, except you would install Wine as appropriate for your distribution (but I haven't actually checked this).