

## Quiz on Lesson 2

July 25, 2022

**Question 1.** Scientifically, our conclusions should not depend on the units of measurement we use, but we can make errors if we don't get the details right. Suppose our data are two years of weekly aggregated case reports of a disease and we have a continuous-time model solved numerically by an Euler timestep of size  $dt$ . Which of the following is a correct explanation of our options for properly implementing this in a `pomp` object called `po`?

- (A) The measurement times, `time(po)`, should be in units of weeks, such as  $1, 2, \dots, 104$ . The latent process can be modeled using arbitrary time units, say days or weeks or years. The units of  $dt$  should match the time units of the **latent** process.
- (B) The measurement times, `time(po)`, should be in units of weeks, such as  $1, 2, \dots, 104$ . The latent process can be modeled using arbitrary time units, say days or weeks or years. The units of  $dt$  should be in weeks (in practice, usually a fraction of a week) to match the units of the **measurement** times.
- (C) The measurement times do not have to be in units of weeks. For example, we could use `time(po)=1/52, 2/52, ..., 2`. The latent process and  $dt$  should use the same units of time as the measurement times.
- (D) The measurement times do not have to be in units of weeks. For example, we could use `time(po)=1/52, 2/52, ..., 2`. The latent process can also use arbitrary units of time, which do not necessarily match the units of the measurement times. The units of  $dt$  should match the units used for the **latent** process.
- (E) The measurement times do not have to be in units of weeks. For example, we could use `time(po)=1/52, 2/52, ..., 2`. The latent process can also use arbitrary units of time, which do not necessarily match the units of the measurement times. The units of  $dt$  should match the units used for the **measurement** times.

**Solution.** C. For scientific calculations, you generally have to pick an arbitrary set of units and use it consistently. In `pomp`, this means that you have to use the same units for measurement times and within the latent process. For example, if your measurement times are in days  $(7, 14, \dots)$  then rate parameters should have units  $\text{day}^{-1}$ . A latent transition with mean duration 1 week would have corresponding rate  $1/7\text{day}^{-1}$ .

**Question 2.** Suppose you obtain the following error message when you build your pomp model using C snippets.

```
Error: error in building shared-object library from C snippets: in 'Cbuilder': compilation error:
cannot compile shared-object library
'/tmp/RtmpFkkeCQ/24104/pomp_068eedfcaf62b1e391363bbdd99fbe8c.so': status = 1
compiler messages:
gcc -I"/usr/local/apps/R/ubuntu_20.04/4.2.1/lib64/R/include" -DNDEBUG
-I'/home/kingaa/R/x86_64-pc-linux-gnu-library/4.2/pomp/include' -I'/home/kingaa/teach/sbied'
-I/usr/local/include -fpic -g -O2 -Wall -pedantic
-c /tmp/RtmpFkkeCQ/24104/pomp_068eedfcaf62b1e391363bbdd99fbe8c.c
-o /tmp/RtmpFkkeCQ/24104/pomp_068eedfcaf62b1e391363bbdd99fbe8c.o
/tmp/RtmpFkkeCQ/24104/pomp_068eedfcaf62b1e391363bbdd99fbe8c.c:
In function '__pomp_rinit':
/tmp/RtmpFkkeCQ/24104/pomp_068eedfcaf62b1e391363bbdd99fbe8c.c:38:13:
error: called object is not a function or function pointer
  38 |         cases = 0
      |         ~
make: *** [/usr/local/apps/R/ubuntu_20.04/4.2.1/lib64/R/etc/Makeconf:168:
/tmp/RtmpFkkeCQ/24104/pomp_068eedfcaf62b1e391363bbdd99fbe8c.o] Error 1
In addition: Warning message:
In system2(command = R.home("bin/R"), args = c("CMD", "SHLIB", "-c", :
running command 'PKG_CPPFLAGS=-I'/home/kingaa/R/x86_64-pc-linux-gnu-library/4.2/pomp/include'
-I'/home/kingaa/teach/sbied' " '/usr/local/apps/R/ubuntu_20.04/4.2.1/lib64/R/bin/R' CMD SHLIB -c
-o /tmp/RtmpFkkeCQ/24104/pomp_068eedfcaf62b1e391363bbdd99fbe8c.so
/tmp/RtmpFkkeCQ/24104/pomp_068eedfcaf62b1e391363bbdd99fbe8c.c 2>&1' had status 1
```

Which of the following is a plausible cause for this error?

- (A) Using R syntax within a C function that has the same name as an R function.
- (B) A parameter is missing from the `paramnames` argument to `pomp`.
- (C) Indexing past the end of an array because C labels indices starting at 0.
- (D) Using `beta` as a parameter name when it is a declared C function.
- (E) A missing semicolon at the end of a line.

**Solution.** E. The error message was produced by the code below. `pomp` passes on the C compiler error message for you to inspect. Note the missing semicolon in the next-to-last line.

```
sir1 <- sir()
sir2 <- pomp(
  sir1,
  statenames=c("S","I","R","cases","W"),
  paramnames=c(
    "gamma","mu","iota",
    "beta1","beta_sd","pop","rho",
    "S_0","I_0","R_0"
  ),
  rinit=Csnippet("
double m = pop/(S_0+I_0+R_0);
S = nearbyint(m*S_0);
I = nearbyint(m*I_0);
R = nearbyint(m*R_0);
```

```
cases = 0
W = 0;"
)
)
```

**Question 3.** Suppose you obtain the following error message when you build your pomp model using C snippets.

```
Error: error in building shared-object library from C snippets: in 'Cbuilder': compilation error:
cannot compile shared-object library '/var/folders/fv/pt62sh2d6_gf9fp3t7b466vr0000gr/T//RtmpD16GmG/
5324/pomp_b675d99e691eda865610f570058ea3be.so': status = 1
compiler messages:
clang -mmacosx-version-min=10.13 -I"/Library/Frameworks/R.framework/Resources/include" -DNDEBUG
-I'/Users/ionides/Library/R/x86_64/4.1/library/pomp/include' -I'/Users/ionides/sbied/questions'
-I/usr/local/include -fPIC -Wall -g -O2
-c /var/folders/fv/pt62sh2d6_gf9fp3t7b466vr0000gr/T//RtmpD16GmG/
5324/pomp_b675d99e691eda865610f570058ea3be.c
-o /var/folders/fv/pt62sh2d6_gf9fp3t7b466vr0000gr/T//RtmpD16GmG/
5324/pomp_b675d99e691eda865610f570058ea3be.o
/var/folders/fv/pt62sh2d6_gf9fp3t7b466vr0000gr/T//RtmpD16GmG/
5324/pomp_b675d99e691eda865610f570058ea3be.c:33:16:
error: use of undeclared identifier 'pop'; did you mean 'pow'?
    double m = pop/(S_0+I_0+R_0);
                ^~~
                pow
/Applications/
In addition: Warning message:
In system2(command = R.home("bin/R"), args = c("CMD", "SHLIB", "-c", :
  running command 'PKG_CPPFLAGS="-I'/Users/ionides/Library/R/x86_64/4.1/library/pomp/include'
-I'/Users/ionides/sbied/questions' " '/Library/Frameworks/R.framework/Resources/bin/R' CMD SHLIB
-c -o /var/folders/fv/pt62sh2d6_gf9fp3t7b466vr0000gr/T//RtmpD16GmG/
5324/pomp_b675d99e691eda865610f570058ea3be.so
/var/folders/fv/pt62sh2d6_gf9fp3t7b466vr0000gr/T//RtmpD16GmG/
5324/pomp_b675d99e691eda865610f570058ea3be.c 2>&1' had status 1
```

Which of the following is a plausible cause for this error?

- (A) Using R syntax within a C function that has the same name as an R function.
- (B) A parameter is missing from the `paramnames` argument to `pomp`.
- (C) Indexing past the end of an array because C labels indices starting at 0.
- (D) Using `beta` as a parameter name when it is a declared C function.
- (E) A missing semicolon at the end of a line.

**Solution.** B. The code generating this error is below. Here, `pop` is intended to be passed as a parameter, but it is missing from the `paramnames` argument. It could alternatively be defined as a global variable using the `globals` argument to `pomp`.

```
sir3 <- pomp(
  sir1,
  statenames=c("S","I","R","cases","W"),
  paramnames=c(
    "gamma","mu","iota",
    "beta1","beta_sd","rho",
    "S_0","I_0","R_0"
  ),
  rinit=Csnippet("
double m = pop/(S_0+I_0+R_0);
```

```
S = nearbyint(m*S_0);  
I = nearbyint(m*I_0);  
R = nearbyint(m*R_0);  
cases = 0  
W = 0;"  
)  
)
```

**Question 4.** Suppose you obtain the following error message when you build your pomp model using C snippets.

```
Error: in 'simulate': error in building shared-object library from C snippets: in 'Cbuilder':
  compilation error: cannot compile shared-object library
  '/tmp/RtmpFkkeCQ/24104/pomp_4fc43714a7a9ebddf896bbc51635d211.so': status = 1
compiler messages:
gcc -I"/usr/local/apps/R/ubuntu_20.04/4.2.1/lib64/R/include" -DNDEBUG
  -I'/home/kingaa/R/x86_64-pc-linux-gnu-library/4.2/pomp/include' -I'/home/kingaa/teach/sbied'
  -I/usr/local/include -fpic -g -O2 -Wall -pedantic -c
/tmp/RtmpFkkeCQ/24104/pomp_4fc43714a7a9ebddf896bbc51635d211.c
-o /tmp/RtmpFkkeCQ/24104/pomp_4fc43714a7a9ebddf896bbc51635d211.o
In file included from /home/kingaa/R/x86_64-pc-linux-gnu-library/4.2/pomp/include/pomp.h:9,
                 from /tmp/RtmpFkkeCQ/24104/pomp_4fc43714a7a9ebddf896bbc51635d211.c:5:
/tmp/RtmpFkkeCQ/24104/pomp_4fc43714a7a9ebddf896bbc51635d211.c: In function '__pomp_rmeasure':
/usr/local/apps/R/ubuntu_20.04/4.2.1/lib64/R/include/Rmath.h:333:16: error:
  too many arguments to function 'Rf_rnorm
In addition: Warning message:
In system2(command = R.home("bin/R"), args = c("CMD", "SHLIB", "-c",
  running command 'PKG_CPPFLAGS="-I'/home/kingaa/R/x86_64-pc-linux-gnu-library/4.2/pomp/include'
  -I'/home/kingaa/teach/sbied'" '/usr/local/apps/R/ubuntu_20.04/4.2.1/lib64/R/bin/R' CMD SHLIB -c
  -o /tmp/RtmpFkkeCQ/24104/pomp_4fc43714a7a9ebddf896bbc51635d211.so
  /tmp/RtmpFkkeCQ/24104/pomp_4fc43714a7a9ebddf896bbc51635d211.c 2>&1' had status 1
```

Which of the following is a plausible cause for this error?

- (A) Using R syntax within a C function that has the same name as an R function.
- (B) A parameter is missing from the `paramnames` argument to `pomp`.
- (C) Indexing past the end of an array because C labels indices starting at 0.
- (D) Using `beta` as a parameter name when it is a declared C function.
- (E) A missing semicolon at the end of a line.

**Solution.** A. The code producing the error is below. Within C snippets, the C versions of R distribution functions are available but they have slightly different syntax from their more familiar R children. A complete reference guide to R's C interface [can be found here](#). In particular, the C form of R's distribution functions is [fully described here](#).

```
sir4 <- simulate(
  sir1,
  statenames=c("S","I","R","cases","W"),
  paramnames=c(
    "gamma","mu","iota",
    "beta1","beta_sd","pop","rho",
    "S_0","I_0","R_0"
  ),
  rmeasure=Csnippet("
    double mean, sd;
    double rep;
    mean = cases*rho;
    sd = sqrt(cases*rho*(1-rho));
    rep = nearbyint(rnorm(1,mean,sd));
    reports = (rep > 0) ? rep : 0;"
  )
)
```

**Question 5.** Let  $V_n$  be a Markov process and let  $W_n = h(V_n)$  for some function  $h$ . Which of the following statements are true?

- (i)  $W_n$  is a Markov process for all choices of  $h$ .
  - (ii)  $W_n$  is a Markov process for some choices of  $h$ .
  - (iii)  $W_n$  is not a Markov process for any choice of  $h$ .
  - (iv) If  $V_n = (X_n, Y_n)$  where  $X_n$  and  $Y_n$  are a POMP, and  $h(X_n, Y_n) = X_n$  then  $W_n$  is a Markov process.
  - (v) If  $V_n = (X_n, Y_n)$  where  $X_n$  and  $Y_n$  are a POMP, and  $h(X_n, Y_n) = Y_n$  then  $W_n$  is a Markov process.
- (A) i,iv,v  
(B) ii,iv  
(C) ii,v  
(D) iii  
(E) None of them

**Solution. B.**