Midterm 2, STATS 531/631 W26

In class on 4/15

This document produces different random tests each time the source code generating it is run. The actual midterm will be a realization generated by this random process, or something similar.

This version lists all the questions currently in the test generator. The actual test will have one question sampled from each of the 6 question categories.

Instructions. The test is closed book, and you are not allowed access to any notes. Any electronic devices in your possession must be turned off and remain in a bag on the floor.

For each question, circle one letter answer and provide some supporting reasoning.

Q1. Foundations of POMP models

Q1-01.

Consider a model $Y_{1:N}$ for data $y^*_{1:N}$, with a latent variable $X_{0:N}$, and a statistical model defined by a joint density $f_{X_{0:N},Y_{1:N}}(x_{0:N}, y_{1:N};\theta)$. The likelihood function is

$$L(\theta) = f_{Y_{1:N}}(y^*_{1:N};\theta).$$

Are the following identities (A) true for all statistical models; (B) true for general POMP models but not all models; (C) true for linear Gaussian POMP models but not general POMP models; (D) generally false? Give a letter answer for each identity.

$$L(\theta) = \int f_{Y_{1:N}|X_{0:N}}(y^*_{1:N}|x_{0:N};\theta)\, f_{X_{0:N}}(x_{0:N};\theta)\, dx_{0:N} \tag{1}$$

$$L(\theta) = \prod_{n=1}^{N} f_{Y_n|Y_{1:n-1}}(y^*_n|y^*_{1:n-1};\theta) \tag{2}$$

$$\mathrm{Var}\{X_{n+1}\,|\,Y_{1:n}\} = E\big[\mathrm{Var}\{X_{n+1}\,|\,X_n\}\,\big|\,Y_{1:n}\big] + \mathrm{Var}\{E[X_{n+1}\,|\,X_n]\,\big|\,Y_{1:n}\} \tag{3}$$

$$L(\theta) = \int \left[\prod_{n=1}^{N} f_{Y_n|X_n}(y_n^*|x_n;\theta) \right] f_{X_{0:N}}(x_{0:N};\theta)\, dx_{0:N} \tag{4}$$

Q1-02.

Which of the following linear Gaussian POMP model have an observable variable Y_n with distribution matching an ARMA(1,1) model? Here, ϵ_n and η_n are Gaussian white noise. X_n is 1-dimensional in (1) and 2-dimensional in (2) and (3).

(A). Only (M3)
(B). (M1) and (M2) but not (M3)
(C). (M2) and (M3) but not (M1)
(D). (M1) and (M3) but not (M2)
(E). (M1), (M2) and (M3)

$$\left.\begin{array}{rcl} X_n &=& aX_{n-1}+\epsilon_n \\ Y_n &=& X_n + \eta_n \end{array}\right\} \qquad (M1)$$

$$\left.\begin{array}{rcl} X_n &=& \begin{pmatrix} a & 1 \\ 0 & 0 \end{pmatrix} X_{n-1} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} \epsilon_n \\ Y_n &=& (1,0)\, X_n + \eta_n \end{array}\right\} \qquad (M2)$$

$$\left.\begin{array}{rcl} X_n &=& \begin{pmatrix} a & 1 \\ 0 & 0 \end{pmatrix} X_{n-1} + \begin{pmatrix} 1 \\ b \end{pmatrix} \epsilon_n \\ Y_n &=& (1,0)\, X_n \end{array}\right\} \qquad (M3)$$

Q1-03.

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.

Q1-04.

Let V_n be a Markov process and let $W_n = h(V_n)$ for some function h. Let (X_n, Y_n) be a POMP with latent process X_n and observed process Y_n. 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 $V_n = (X_n, Y_n)$ and $h(X_n, Y_n) = X_n$ then W_n is a Markov process.

v. If $V_n = (X_n, Y_n)$ 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). Some combination other than those listed above

Q2. Likelihood evaluation; the particle filter

Q2-01.

Suppose that 10 replications of a particle filter, each using 10^3 particles, runs in 15 minutes with no parallelization. To look for a more precise likelihood evaulation, you consider running 20 replicates, each with 10^4 particles. Approximately how many minutes will this take, if you distribute the calculation across 4 cores?

(A). 50
(B). 60
(C). 75

(D). 120
(E). 300

Q2-02.

A particle filter is repeated 5 times to evaluate the likelihood at a proposed maximum likelihood estimate, each time with 10^4 particles. Suppose the log likelihood estimates are -2446.0, -2444.0, -2443.0, -2442.0, -2440.0. Which of the following is an appropriate estimate for the log likelihood at this parameter value and its standard error.

(A). Estimate = -2443.0, with standard error 1.0
(B). Estimate = -2443.0, with standard error 2.2
(C). Estimate = -2443.0, with standard error 5.0
(D). Estimate = -2441.4, with standard error 2.2
(E). Estimate = -2441.4, with standard error 1.4

Q2-03.

Effective sample size (ESS) is one of the main tools for diagnosing the success of a particle filter. If you plot an object of class `pfilterd_pomp` (created by applying `pfilter` to a pomp object), the ESS is displayed. Suppose one or more time points have low ESS (say, less than 10) even when using a fairly large number of particles (say, 10^4). What is the proper interpretation?

(A). There is a problem with data, perhaps an error recording an observation.
(B). There is a problem with the model which means that it cannot explain something in the data.
(C). The model and data have no major problems, but the model happens to be problematic for the particle filter algorithm.
(D). At least one of A, B, and C.
(E). Either A or B or both, but not C. If the model fits the data well, the particle filter is guaranteed to work well.

Q2-04.

In a particle filter, a particle that is resampled k times is said to be the parent of these k children, The complete ancestry graph of all the particles is an evolutionary tree for the population of particles. Each filtering iteration corresponds to a generation of this population process. Darwinian evolution on populations occurs when individuals with the fittest phenotype reproduce more offspring for the next generation, and these children inherit the parent's genotype subject to mutation.

Particle filter	Darwinian evolution
(1) Prediction step: simulation	(a) Genotype
(2) Filtering step: resampling	(b) Fitness
(3) State value, $X_{n,j}$	(c) Mutation

Particle filter	Darwinian evolution
(4) Measurement density, $f_{Y_n\vert X_n}(y_n^*\vert X_{n,j})$	(d) Reproduction

What is the pairing between the particle filter concepts (numbers 1–4) and the analogous evolutionary concepts (letters a–d).

(A). (1a) (2b) (3c) (4d)
(B). (1b) (2a) (3c) (4d)
(C). (1c) (2d) (3a) (4b)
(D). (1d) (2c) (3a) (4b)
(E). A combination not listed above.

Q2-05.

The particle filter provides a log-likelihood estimate that is negatively biased. Is this a consequence of:

(A). Jensen's inequality.
(B). Fubini's theorem.
(C). Wilks' theorem.
(D). Chebyshev's inequality.
(E). Consistency of the Metropolis-Hastings algorithm.
F. Unbiasedness of the likelihood estimate for sequential Monte Carlo.
G. Two or more of the above.

Q3. Likelihood maximization; iterated filtering

Q3-01.

When carrying out inference by iterated particle filtering, the likelihood increases for the first 10 iterations or so, and then steadily decreases. Testing the inference procedure on simulated data, this does not happen and the likelihood increases steadily toward convergence. Which of the following is the best explanation for this?

(A). One or more random walk standard deviations is too large.

(B). One or more random walk standard deviations is too small.

(C). The model is misspecified, so it does not fit the data adequately.

(D). A combination of the parameters is weakly identified, leading to a ridge in the likelihood surface.

(E). Too few particles are being used.

Q3-02.

People sometimes confuse likelihood profiles with likelihood slices. When you read a report claiming to have computed a profile it can be worth checking whether it is actually computed as a slice. Suppose you read a figure which claims to construct a profile confidence interval for a parameter ρ in a POMP model with four unknown parameters. Which of the following confirms that the plot is, or is not, a properly constructed profile confidence interval.

(A). The CI is constructed by obtaining the interval of rho values whose log likelihood is within 1.92 of the maximum on a smoothed curve of likelihood values plotted against ρ.
(B). The code (made available to you by the authors as an Rmarkdown file) involves evaluation of the likelihood but not maximization.
(C). The points along the ρ axis are not equally spaced.
(D). The smoothed line shown in the plot is close to quadratic.
(E). A and D together.

Q3-03.

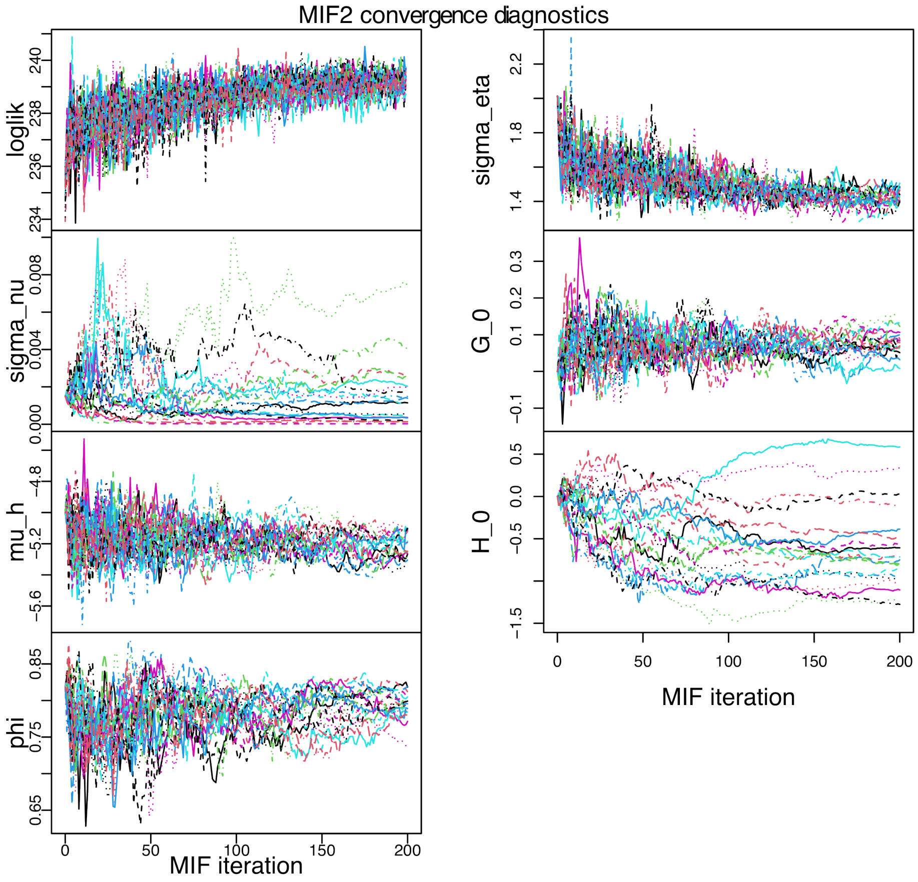

Figure 1: Iterated filtering diagnostic plot

The iterated filtering convergence diagnostics in figure 1 come from a student project investigating the market value of Gamestop. What is the best interpretation?

(A). Everything seems to be working fine. The likelihood is climbing. The replicated searches are giving consistent runs. The spread of convergence points for σ_ν and H_0 indicates weak identifability, which is a statistical fact worth noticing but not a weakness of the model.

(B). The consistently climbing likelihood is promising, but the failure of σ_ν and H_0 to converge needs attention. Additional searching is needed, experimenting with **larger** values of the random walk perturbation standard deviation for these parameters to make sure the parameter space is properly searched.

(C). The consistently climbing likelihood is promising, but the failure of σ_ν and H_0 to converge needs attention. Additional searching is needed, experimenting with **smaller** values of the random walk perturbation standard deviation for these parameters to make sure the parameter space is properly searched.

(D). The consistently climbing likelihood is promising, but the failure of σ_ν and H_0 to converge needs attention. This indicates weak identifiability which cannot be solved by improving the searching algorithm. Instead, we should change the model, or fix one or more parameters at scientifically plausible values, to resolve the identifiability issue before proceeding.

(E). Although the log likelihood seems to be climbing during the search, until the convergence problems with σ_ν and H_0 have been addressed we should not be confident about the successful optimization of the likelihood function or the other parameter estimates.

Q3-04.

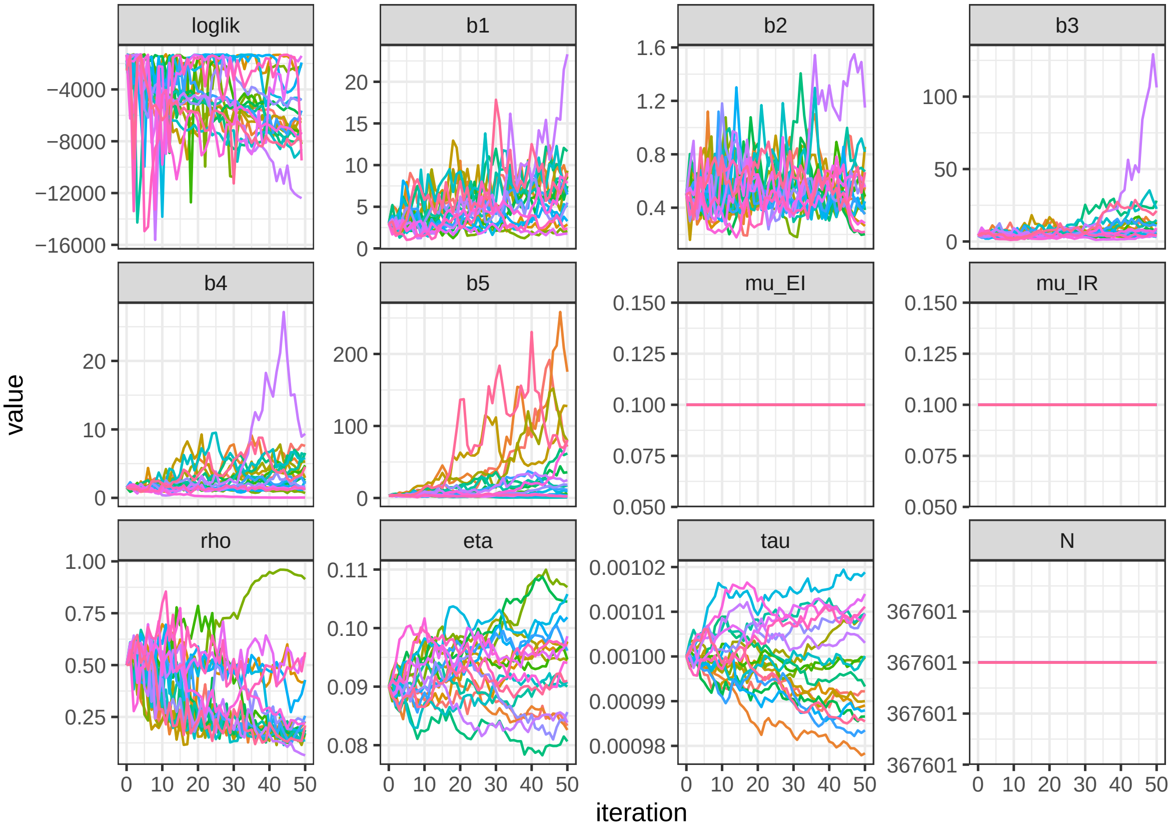

Figure 2: Diagnostic plot for a COVID-19 model

The iterated filtering convergence diagnostics plot in figure 2 comes from a 2021 student project investigating COVID-19. The calculation used 10^3 particles. What is the best interpretation?

(A). Everything seems to be working fine. There is a clear consensus from the different searches concerning the highest likelihood that can be found. Therefore, the search is doing a good job of maximization. Occasional searches get lost, such as the purple line with a low likelihood, but that is not a problem.

(B). The searches obtain likelihood values spread over thousands of log units. We would like to see consistent convergence within a few log units. We should use more particles and/or more iterations to achieve this.

(C). The searches obtain likelihood values spread over thousands of log units. We would like to see consistent convergence within a few log units. We should compare the best likelihoods obtained with simple statistical models, such as an auto-regressive moving average model, to look for evidence of model misspecification.

(D). The searches obtain likelihood values spread over thousands of log units. We would like to see consistent convergence within a few log units. We should look at the effective sample size plot for the best fit we have found yet, to see whether there are problems with the particle filtering.

(E). All of B, C, and D.

Q4. Data analysis: epidemiological models

Q4-01.

Two models are fitted to case counts on an epidemic. Model 1 is an SIR POMP model with a negative binomial measurement model, and model 2 is a linear regression model estimating a cubic trend. The log likelihoods are $\ell_1 = -2037.91$ and $\ell_2 = -2031.28$ respectively. Which of the following do you agree with most?

(A). We should not compare the models using these likelihoods. They correspond to different model structures, so it is an apples-to-oranges comparison.

(B). We can compare them, but the difference is in the 4th significant figure, so the likelihoods are statistically indistinguishable.

(C). The linear model has a noticeably higher likelihood. Our mechanistic model needs to be updated to beat this benchmark before we can responsibly interpret the fitted model. If a simple regression model has higher likelihood than a more complex mechanistic model, one should prefer the simpler model.

(D). The linear model has a noticeably higher likelihood. The mechanistic model is somewhat validated by being not too far behind the simple regression model. We are justified in cautiously interpreting the mechanistic model, while continuing to look for further improvements.

(E). The log likelihoods cannot properly be compared as presented, but could be if we used a Gaussian measurement model for the POMP (or a negative binomial generalized linear model instead of least squares for the regression).

Q4-02.

A compartment model is first implemented as a system of ordinary differential equations (ODEs). This leads to qualitatively reasonable trajectories, but poor likelihood values. The researchers add stochasticity in an attempt to improve the fit of the model by interpreting the ODEs as rates of a Markov chain. The likelihood, maximized by iterated particle filtering, remains poor compared to ARMA benchmarks. In addition, the effective sample size for the particle filtering is low at many time points despite even using as many as 10^4 particles. Which of the following is the most promising next step?

(A). Increase to 10^5 particles, moving the computations to a cluster if necessary.
(B). Add noise to one or more rates to allow for overdispersion.

(C). Try adding extra features to the model to capture scientific details not present in the original model.
(D). Experiment with variations in the iterated filtering procedure; maybe more iterations, or a different cooling schedule.
(E). To address the possibility of reporting errors, see if the model fits better when the most problematic data points are removed.

Q4-03.

You fit an SEIR model to case reports of an immunizing disease from a city. The resulting confidence interval for the mean latent period is 12–21 days, but clinical evidence points to a latent period averaging about 7 days. Which of the following is the most appropriate response to this discrepancy?

(A). The latent period may be confounded with some unmodeled aspect of the system, such as spatial or age structure. The model estimates an effective latent period at the population level, which may not perfectly match what is happening at the scale of individuals.

(B). The discrepancy shows that something is substantially wrong with the model. Extra biological detail must be introduced with the goal of bringing the estimated parameter back in line with the known biology of the system.

(C). The discrepancy is problematic, but fortunately can readily be fixed. Since we know the clinical value of this parameter with reasonable accuracy, we should simply use this value in the model rather than estimating it.

(D). If the model fits the data statistically better than any known alternative model, then we have to take the estimated parameter at face value. It is certainly possible that the estimates in the literature correspond to some different population, or different strain, or have some other measurement bias such as corresponding to severe cases resulting in hospitalization. The discrepancy does not show that our model was wrong.

(E). This discrepancy suggests that we should take advantage of both C and D above by putting a Bayesian prior on the latent period. By quantifying the degree of our skepticism about the previously established clinical value of 7 days, we can optimally combine that uncertainty with the evidence from this dataset.

Q4-04.

Let $X(t) = (S(t), I(t), R(t))$ be a discrete-valued SIR model, and suppose that the counting process $N_{SI}(t)$ has per-capita rate $\mu_{SI}(t) = \beta I(t)$. Consider four Euler approximations:

$$
\begin{aligned}
N_{SI}(t+\delta) - N_{SI}(t) &\sim \text{Poisson}\big(\beta I(t) S(t) \delta\big) && (5)\\
N_{SI}(t+\delta) - N_{SI}(t) &\sim \text{Binomial}\big(S(t), \beta I(t) \delta\big) && (6)\\
N_{SI}(t+\delta) - N_{SI}(t) &\sim \text{Binomial}\big(S(t), 1 - \exp\big\{-\beta I(t)\delta\big\}\big) && (7)\\
N_{SI}(t+\delta) - N_{SI}(t) &\sim \text{Binomial}\big(S(t), \beta I(t)\delta \big/ (1 + \beta I(t)\delta)\big) && (8)
\end{aligned}
$$

Which (if any) of these choices is preferable, and why?

(A). (1) because it is the simplest, quickest to simulate, and most easily understood. All continuous time, discrete state, compartment models have approximately Poisson increments on small time intervals.

(B). (2) because each individual in S has approximately an independent chance $\beta I(t)\delta$ of transitioning to I.

(C). (3) because each individual in S has approximately an independent hazard rate $\beta I(t)\delta$ of transitioning to I.

(D). (4) because it is asymptotically equivalent to (3), up to order δ^2, but quicker to compute.

(E). It should not matter which you choose. All the methods are correct Euler schemes, with the same justification in a limit when δ is small. Therefore, there is no theoretical reason to prefer one over another.

Q5. Data analysis: financial models

Q5-01.

A generalized autoregressive conditional heteroskedasticity (GARCH) model has $Y_n = \sigma_n Z_n$ where $Z_n \sim$ i.i.d.$N(0,1)$ and

$$\sigma_n^2 = \alpha_0 + \sum_{i=1}^{p} \alpha_i Y_{n-i}^2 + \sum_{j=1}^{q} \beta_j \sigma_{n-j}^2.$$

For data $y_{1:N}^*$, residuals may be defined by $r_n = Y_n/\hat{\sigma}_n$ where $\hat{\sigma}_n$ is an estimate of σ_n. Suppose that we fit a GARCH model to the log-returns of a financial time series, and we find that the sample ACF of $r_{1:N}$ is consistent with white noise (e.g., 531W24 final project #7). What is the best inference from the residual ACF about the success of the GARCH model for these data?

(A). This supports the use of GARCH over ARMA. That is not especially surprising, since it is true for essentially all financial time series, but it is good to check.
(B). A fitted ARMA model is also anticipated to have a residual ACF consistent with white noise. The problem with the ARMA model for financial data is not residual autocorrelation.
(C). We should also make a normal quantile plot of the residuals. If the residuals are approximately normal then the ACF plot becomes more trustworthy as a test for lack of correlation. If the residuals are far from normal, we should not draw conclusions from the sample ACF.
(D). GARCH aims to fix the problem of conditional heteroskedasticity in financial data that ARMA cannot explain. However, fixing this might break the negligible autocorrelation that is critical for the efficient market hypothesis. It is good to see that we can fix conditional heteroskedasticity while remaining compatible with the efficient market hypothesis.

Q5-02.

A generalized autoregressive conditional heteroskedasticity (GARCH) model has $Y_n = \sigma_n Z_n$ where $Z_n \sim$ i.i.d.$N(0,1)$ and $\sigma_n^2 = \alpha_0 + \sum_{i=1}^p \alpha_i Y_{n-i}^2 + \sum_{j=1}^q \beta_j \sigma_{n-j}^2$. There are many extensions to GARCH implemented by various R packages. When comparing models by likelihood or AIC, care is required since packages do not always use standard definitions. This table was produced in R with the widely used `tseries::logLik.garch` function. What is the most reasonable interpretation of this table?

	q1	q2	q3	q4
p1	2646.277	2642.919	2620.280	2616.151
p2	2644.417	2625.417	2622.460	2616.427
p3	2641.804	2637.538	2625.953	2625.740
p4	2639.728	2629.869	2629.969	2628.345

(A). The positive values of the log-likelihood are implausible. Perhaps the software actually reports the negative log-likelihood since many optimizers are designed to minimize rather than maximize.

(B). The models are nested and so a larger model should mathematically have a larger likelihood. In this table, the larger model usually has lower likelihood, so optimization is problematic.

(C). This table would make more sense if `logLik` in fact returns an AIC value. The preferred model is $(p, q) = (1, 4)$.

(D). The preferred model is $(p, q) = (1, 1)$ since it is both the simplest model and the one with the highest log-likelihood.

(E). `tseries::logLik.garch` produces something that is not the log-likelihood of $y_{1:N}$, so we cannot readily compare it between models.

Q5-03.

The Heston model for volatility, V_n, is a stochastic volatility (SV) model with

$$V_n = (1-\phi)\theta + \phi V_{n-1} + \sqrt{V_{n-1}}\,\omega_n,$$

for $\omega_n \sim N[0, \sigma_\omega^2]$. The log return is $Y_n \sim N[0, V_n]$, conditional on V_n. A previous 531 project (W22, #14) fitted the Heston model to investment in Ethereum, a crypto currency. They obtained a log-likelihood of 34975.3, compared to 28587.4 for GARCH and 28977 for the SV model with leverage presented in class. Their iterated filtering convergence diagnostics are shown in figure 3. What is the best conclusion from this information?

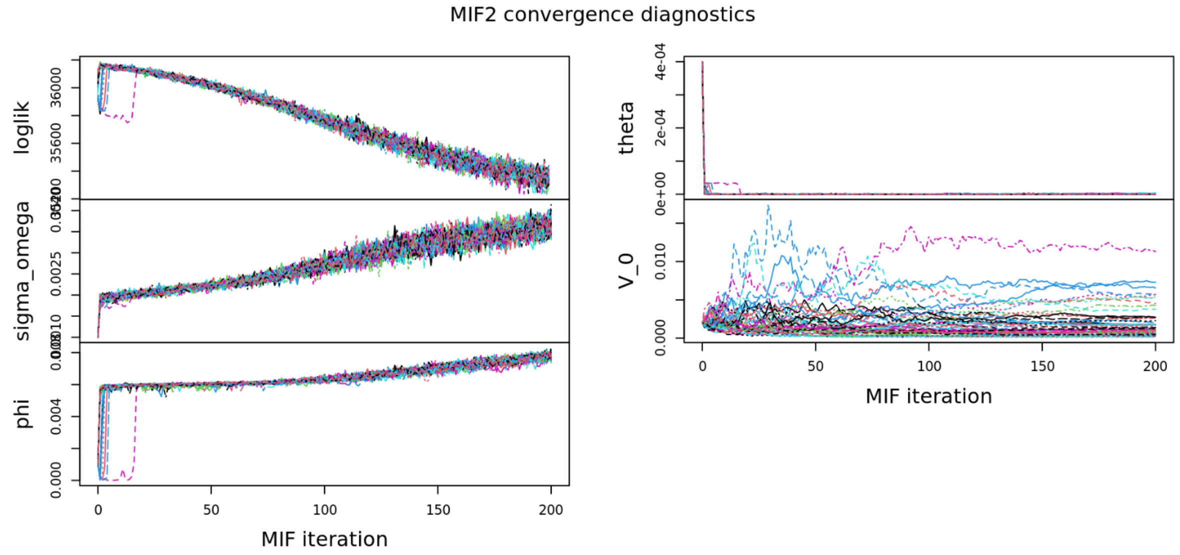

Figure 3: Diagnostic plot for fitting the Heston model

(A). The high likelihood shows this is a promising model despite the convergence problems identified in the figure. Attention to the diagnostics may lead to additional improvements.

(B). The most important diagnostic feature is the observation that the log-likelihood trace plot peaks and then declines. From the y-axis scale we see the decline is of order 1000 log units. This is evidence of substantial model misspecification which should be addressed.

(C). The most important dignostic feature is that the `theta` traces all drop quickly to zero. Since that is not a scientifically plausible value for the parameter, we can deduce that the model is unsuccessful despite its high likelihood.

(D). The most important diagnostic feature is that `phi` is close to zero and well identified. This shows that the volatility is close to constant, and is supported by the high likelihood.

(E). The decreasing likelihood and other convergence diagnostics problems show there is a problem with the model. Likely, there is a bug and the high likelihood obtained is simply an error.

Q6. Diagnostics and debugging for POMP models

Q6-01.

A useful way to check statistical methodology is to apply an inference method to a collection of simulated datasets from the fitted model with the estimated parameter values (say, the maximum likelihood estimate, MLE). This is sometimes called a “parametric bootstrap”. Suppose that we carry out this check for a POMP data analysis, using plug-and-play inference methodology such as iterated filtering, and we find that the re-estimated parameters from inference on the simulated data are close to the MLE. What can we infer about the correctness of our inference.

(A). This is a strong check that both the model and the methodology are correctly implemented. Except for some rare special cases, an error in either one of these will lead the check to fail.

(B). This checks the implementation of the inference methodology but not the model. Even if the model is implemented wrongly, the check will still show us whether the inference methodology is correct.

(C). This checks the implementation of the model but not the inference methodology. As long as the model is implemented correctly, any reasonable inference methodology should pass the check successfully.

(D). This is not a strong check of either the model or the methodology. It shows self-consistency but that is different from showing accuracy.

Q6-02.

Pypomp gains power from JAX, but JAX is a sophisticated tool that has its sharp bits. Keeping alert for these can help address subtle bugs that could arise in your Pypomp code.

```python
import jax
key = jax.random.key(42)
def get_random_normal(key, shape):
    new_key, subkey = jax.random.split(key)
    x = jax.random.normal(subkey, shape)
    return new_key, x

get_random_normal_jit = jax.jit(get_random_normal)
```

We can generate 10 standard normal random variables as follows

```python
get_random_normal(key,10)
```

But the jitted code gives a hard-to-interpret error message, `TypeError: iteration over a 0-d array`

```python
get_random_normal_jit(key,10)
```

What is the problem?

(A). The code is okay, so there is some deeper problem. Perhaps we are working with an old version of Python (say, 3.8) and using a recent JAX version which is not fully backward-compatible, leading to a compilation error.

(B). Arguments to compiled functions must be JAX arrays. Replacing `10` by `jax.numpy.array([10])` will fix it.

(C). We cannot expect JAX to compile code for a variable value of random numbers. Set

```
get_random_normal_jit = jax.jit(get_random_normal, static_argnames=['shape'])
```

(D). We cannot reasonably expect JAX to compile for a variable `key`. We should set

```
get_random_normal_jit = jax.jit(get_random_normal, static_argnames=['key'])
```

(E). The problem will be solved if we use the `@jit` decorator to compile the code while keeping the same name as the original function. For example,

```
@jax.jit
def get_random_normal(key, shape):
    new_key, subkey = jax.random.split(key)
    x = jax.random.normal(subkey, shape)
    return new_key, x

get_random_normal(key,10)
```

will run the compiled version, without error. Note that

```
@jax.jit
get_random_normal
```

is equivalent to

```
get_random_normal = jax.jit(get_random_normal)
```

Q6-03.

The following defines the dynamics of a certain SIR model in Pypomp:

```
def rproc(X_, theta_, key, covars, t, dt):
    S, I, R = X_["S"], X_["I"], X_["R"]
    Beta = theta_["Beta"]
    mu_IR = theta_["mu_IR"]
    N = theta_["N"]
    dN_SI = jax.random.binomial(key, n=jnp.int32(S), p=Beta * I / N * dt)
    dN_IR = jax.random.binomial(key, n=jnp.int32(I), p=mu_IR * dt)
    S_new = S - dN_SI
    I_new = I + dN_SI - dN_IR
    R_new = R + dN_IR
    return {"S": S_new, "I": I_new, "R": R_new}
```

What is the most critical flaw in this code? Or is the code valid?

(A). The random number stream `key` is not passed correctly.

(B). The binomial probability should be exponentiated, say $1 - \exp\{-\mu_{IR} dt\}$.

(C). The code is missing an accumulator state, H, to keep track of the number of new cases.

(D). This model has only demographic noise, represented by the binomial distribution. For statistically matching the variability in data, it can be important to include extra-demographic variability, such as noise on the transmission rate, β.

(E). The model may be fine. Without seeing the data or knowing the scientific context, we can't tell for sure whether there is a problem.

Q6-04.

A basic diagnostic plot is to simulate from the fitted model POMP model at the parameter value with the highest likelihood found so far in the search of the parameter space. If this simulation is qualitatively similar to the data, this suggests the data are consistent with the model. If the simulation is qualitatively different from the data, which of the following is the best next step?

(A). We should do additional global and local searches to make sure we have accurately maximized likelihood estimate. If the problem remains, we can try other approaches with more confidence.

(B). Visual comparisons of trajectories can be misleading. For example, there is no reason why a simulation from a model should match the timing of events in the dataset (e.g., recessions in economic data; outbreaks in epidemic data). Maybe our model is good despite the mismatching simulation.

(C). We should think hard about why the simulations don't match the data. There could easily be a missing scientific mechanism that needs to be included to explain the data. Unless we think about the science, we won't work this out.

(D). We should compare the log-likelihood against non-mechanistic benchmarks, such as ARMA models. If we can find a simple statistical model that has better AIC than our mechanistic model, that would strengthen the inference that there might be a problem with the data.

(E). We should inspect residuals (the difference between observed data and the one-step prediction mean) to help us identify which data points are explained poorly by the model.
