PURDUE UNIVERSITY

Weldon School of Biomedical Engineering

Biomedical Signal Processing
BME 595 - Fall 2021
Problem Set 3

This problem set accounts for 6% of your final grade. Please submit your solutions through **Brightspace** as a single .ipynb file or .pdf file containing typed explanations, plots, and any code that has been requested explicitly. Keep the text as brief as possible. Each problem lists the plots and results that need to be included in your solutions report. Please label the axes of all plots with appropriate units. You need not include any code that is not explicitly requested, but no problem if you do.

1. Here, we will gain more familiarity with using random processes to model signals of various "smoothness" levels and see how the smoothness affects the statistical properties of commonly evaluated features such as peak-to-peak or cluster sizes. Because statistical modeling is often more important under the *null hypothesis* where there isn't really a signal waveform of interest present, we will focus on zero-mean stationary signals. For instance, in the newborn hearing screening example discussed in class, when the baby has no hearing, there isn't any notable feature in the response and we are just averaging over a number of noise measurements. Thus, stationarity and the zero-mean property are reasonable assumptions to make.

   One way to generate zero-mean stationary signals is using LTI filtered versions of white noise. If the filter used is an FIR filter, then the resulting random process is called a *moving average* (MA) process because FIR filtering, as you know, generates an output sample by doing weighted averaging of a certain number of past *input* samples (the weighted come from the FIR impulse response). That is,

   $$x(t) = \sum_{k=0}^{p-1} h(k)w(t-k) \tag{1}$$

   Another way of generating zero-mean stationary random signals is to allow the signal to "evolve" to new values that are weighted sums of past value of the *signal itself*, but with some purely random "innovation" (i.e., white noise samples) added at each step such. Intuitively, if the innovation is small compared to the dependence on past values, the signal will evolve smoothly. On the other hand, if the innovation is large and dependence on current/past values small, the signal will look more like white noise. A random process which can be thought of being generated this way is called an*autoregressive* process (AR):

   $$x(t) = \sum_{k=0}^{p-1} a(k)x(t-k) + w(t) \tag{2}$$

   It can be shown mathematically that AR processes can also be thought of as filtered versions of white noise, but here the filter would be IIR. Here, we will use a very simple AR process to simulate stationary gaussian random processes with various smoothness levels.

   (a) The simplest AR process one can think of is something that just has 1 parameter, sometimes called an *AR(1) process*

   $$x(t) = ax(t-1) + w(t) \tag{3}$$

where $w(t)$ is white noise (i.e., gaussian random variables that are independently realized for each time point) with zero mean and some variance (let's call that σ_w^2), and a is a parameter of the AR process[1]. Different values of a will yield different autocorrelation functions for the resulting $x(t)$. Our first step is to calculate analytically, how the autocorrelation function of $x(t)$, i.e., $R_{xx}(\tau)$ depends on a. Recognizing that

$$R_{xx}(\tau) = E[x(t)x(t+\tau)] \tag{4}$$

where $E[.]$ denotes the mean value or *expectation* operator, calculate an expression for $R_{xx}(\tau)$ as a function of a, τ and σ_w^2. **Hint:** By applying the basic definition of $R_{xx}(\tau)$ in Equation 4 above, you can quickly show that $R_{xx}(\tau) = aR_{xx}(\tau - 1)$. Then you can work the recursion all the way back to $R_{xx}(0)$, which is nothing but the variance of $x(t)$. The variance can easily be calculated assuming stationarity, and by using the fact that the variance of the sum of two random variables is the sum of the variances as long as the variables are uncorrelated.

(b) Now, use MATLAB or python to simulate 2000 samples of an AR(1) process for $a = 0.99$ and $\sigma_w^2 = 1$. To do so, create 2000 samples of white noise using `randn` in MATLAB or `numpy.random.randn` in python and apply Equation 3 directly from the second sample going forward. Then repeat the whole process ten times to get ten random examples of your AR(1) process. Plot them all on top of each other with different colors. Remembering that the IIR filter will take a long time to settle, show only the last 1000 samples (perhaps using `xlim` in MATLAB or `pylab`). Does it simulate a somewhat smooth stationary noise (i.e., smoother than white noise)?

(c) Calculate the across-sample variance of each of the ten AR(1) examples you generated. They will be different because of inherent randomness. However, the numbers you get should be spread around the true value of variance, i.e., $R_{xx}(0)$ that you calculated in part (a). Does that happen?

(d) Convert your code from Part (b) into a a function called `ar1` that takes the value of the AR(1) parameter `a`, the length of the simulation (`Nsamps`, previously 2000 samples), and the number of example signals to generate (`Nexamples`, previously 10) as inputs, and returns `x` which is a `Nexamples` $\times$ `Nsamps` matrix. This let's you get any length of signal and any number of examples of such signals. Also make the function such that the returned signal has an average variance of 1. You can do this by dividing the value of `x` by the **standard deviation** you expect for the particular value of `a` from your analytical calculations. Include the code for this function in the document you turn in.

(e) Use the `ar1` function to simulate 10 examples as before but with $a = 0.999$ and plot the second half as before. Include this plot in your report. Comment on the relative smoothness of this signal compared to when you had $a = 0.99$. Does that agree with your intuition about the autocorrelation expression that you calculated in the first part?

(f) Let's say you have determined from some baseline data for newborn hearing screening that an AR(1) process is a good model for the measurements when there is no response. LEt's also say that you have a way of estimate what the a value is from some data from deaf babies (We'll talk more about estimation in latter half of the course). Let's say our estimate is $a = 0.99$. Now let's say that a baby that comes in for screening produces a response whose peak value is

[1] Incidentally, the absolute value of a should be less than 1 for the process to be stationary, which you might recognize based on your calculations. For simpilicity, assume that $|a| < 1$

4 (in suitable units). Use your `ar1` function to simulate 1000 examples of $x(t)$ when $a = 0.99$. For how many of those examples is the maximum value of x going beyond a value of 4? This kind of a calculation can be used to estimate p-values for the measured response when using AR(1) type models for the underlying noise.

2. Let's consider a simple example of modeling a phenomenon and then asking if the data is consistent with the model. In the second half of this course, a multiple choice problem set will posted with about 50 questions, each with four options. However, let's say that the options are such that it is easy to eliminate one or two for most questions, leaving really 3 viable choices on average. Under the scenario of 50 questions, each with 3 options:

   (a) If a student were to simply take a random guess at an answer for each question, what would be their average/expected score out of 50? Note that it is reasonable to assume that the guess for each question is independent.

   (b) Under the conventional p-value criterion of 5% to reject the "null" hypothesis $\mathcal{H}_0$, what is the minimum number of questions that a given student would have to get correct for rejecting the hypothesis $\mathcal{H}_0$ that they were simply guessing at an answer for each question independently? To answer this, first construct a suitable model for what distribution for the number of correct answers will be. Then use this model to calculate the score for which the p-value is 0.05.

   Caveat: If the student were merely guessing, both very low scores and very high scores would be unlikely. This raises the issue of "one-sided" vs. "two-sided" statistical testing. We will ignore that issue for now and for the purposes of this question, focus just on *high scores* that would be unlikely under $\mathcal{H}_0$. That is, let's define p-value for a particular score as the "probability under $\mathcal{H}_0$ of getting *that score or higher*".

   (c) Let's say this multiple choice assignment was worth 15 points. If points were allocated for this multiple choice assignment such that p=0.5 or higher received a zero, and then for every order-of-magnitude reduction in p-value, an extra point was allocated, how many questions would a student need to get correct to get the full 15 points? That is, they get 1 point when they hit p=0.05, 2 if they hit 0.005, 3 for 0.0005, and so on. **PS:** The multiple choice problem set in this course will actually be graded along these lines, although the exact number of questions and total points will be slightly different.

3. Here, let us take a closer look at basic probabilistic notions like *false-alarm rate*, *hit rate*, etc. that are applicable to detection problems.

   (a) Let's say a disease occurs in 50% of all people. This is sometimes called the *prevalance* of the disease. Now let's say someone designs a test that takes a blood sample of each person being tested for the disease and analyzes the biomarkers in the sample to return a binary decision, i.e., "positive" or "negative" for the disease. Let's say that the test has a false-alarm rate of 10% (i.e., specificity is 90%), and a hit-rate (i.e., sensitivity) of 90%.

   Note that the false alarm rate is a number that makes sense to talk about when you are testing people who don't actually have the disease – it is the proportion of people that test positive when they all in fact should test negative. Similarly hit rate is a number that makes sense to talk about when you are testing people with the disease – it is the proportion of people that test positive when they all in fact have the disease.

   For a random individual, what is the probability that they actually have the disease if they test positive for it?

(b) Check your calculations in part (a) by simulating this scenario with 10,000 total people. That is,

- First divide the 10,000 people up into two groups, one actually having the disease and the other not, in proportion with the prevalence of the disease
- Give each person in each group a test result that is in proportion with the sensitivity and specificity values of the test. Remember that when a person actually having the disease is being tested, the relevant metric is hit-rate. Similarly, when a person without the disease is being tested, the relevant metric is false-alarm rate.
- Now take all the people that ended up with a positive test result in this simulation, and see what proportion actually have the disease. This should match your calculation in part (a).

(c) Repeat the calculation similar to part (a), and the simulation similar to part (b), but now assuming that the prevalence of the disease is just 1%.

(d) One technique that is sometimes used to make any test more useful is repeat-testing a certain number of times. When testing a random individual from the general population when the prevalence is 1%, what is the probability that they actually have the disease if they test positive twice? Similarly, what is the probability that they actually have the disease if they test positive thrice? For the purposes of this problem assume that repeat tests are statistically independent (this is not true in practice, however).