**Time Series 2**

Here’s some solved examples, to aid with intuition and such. Note we’ll repeatedly use:



So proceeding,

**AM(1) model**

This guy looks like this:



Instead of xn depending on the previous x, it depends on the previous noise. I guess there is no need to solve the equation; the solution is right there. What would be the statistics of this model?



The covariance is:



So,



At least as far as the mean and variance are concerned, this model is no different than white noise, with a different white noise amplitude. But the correlation thing is different; now a point is correlated with both of its neighbors. We can see that these also satisfy the time-independent requirements for stationarity. For instance, we can see <xm+txn+t> = <xmxn>. Finally, let’s do the ACF:



assuming k is positive. So this is also stationary. Can see ACF(0) is 1, as it should be. And unlike pure white noise, now succesive points are correlated, with sign depending on θ. The probability distribution of xn should be:



since it is, like all the others, just the sum of Gaussian variables. What about the conditional distribution? Well P(xn|x0, x1, …, xn-1) would seem to depend on all those preceding coordinates. Consider the first three points:



where we presume ΔW0 = 0. Consider P(x3|x0, x1, x2). x3 depends on ΔW3 and ΔW2. But if we specify x1 and x2, then we’ve specified ΔW1 and ΔW2, and so the only variable left unspecified for x3 is ΔW3. Moreover, we can write ΔW2 in terms of x0, x1, x2. We have: ΔW2 = x2 – βΔt – θΔW1 = (x2 – βΔt) – θ(x1 – βΔt). So since xn = βΔt + ΔWn + θΔWn-1, P(xn|x0, x1, …, xn) should be given by a normal distribution:



where we implicitly note that ΔWn-1 can be determined from the preceeding points: x0, x1, …, xn-  Can we work out a formula for this? Here’s the equations out to n = 5:



So in general we can say:



We can get the ΔW’s by finding the inverse of K1. We can find the inverse of the matrix by doing row operations on the identity matrix – that thing from Linear Algebra. Let’s use our n = 5 example for simplicity.



So we can see the pattern:



So it follows that:



And now we could write the conditional distribution as:



So here’s a question. xn seems to be correlated with all the other points that come before it, due to the presence of all the other xj<n within ΔWn-1. But when we do calculate correlations between points, we only see a correlation between nearest neighbors. Well, a variable can be conditionally dependent on another, even if their covariance is zero – think A = X, and B = X2, over an even interval. Would an RNN need to know all previous points to make a prediction? I hope not somehow. Well, we can see that if we’re making predictions on xn>>1, and if θ << 1, then the contributions of the terms far away from n make little contribution, because according to the ΔWn formula, contributions from xj go as (-θ)n-j. So only for j close to n will we get a decent contribution. But this window is definitely θ-dependent. So if θ is close to 1 then we might need a lot of preceding xj’s.

What about the joint probability distribution?



So there. But since ΔWn-1 is a linear combination of the xj’s, this formula works out to a normal distribution for x1, x2, …, xn over all. And since we know the average and covariance of the distribution, we can just say:



The covariance matrix σ is a banded matrix with the constant value of (1+θ2)DΔt down the diagonal, and θDΔt along the two diagonals on either side. As an example of such a series, we can consider white noise, and take the difference.



The white noise difference follows an MA(1) model. To be more specific, we could say: x0 = 0, x1 = ΔW1, x2 = ΔW2, x3 = ΔW3, etc. And then Dx1 = ΔW1, Dx2 = ΔW2 – ΔW1, Dx3 = ΔW3 – ΔW2, etc. Seems weird that P(xn|xj) depends only on j = n, while P(Dxn|Dxj) depends on all j < n. I guess this makes sense. The probability distribution of Dxn = xn – xn-1 depends on xn-1, since if we know xn-1, then we know P(Dxn) to the maximum degree possible. But to know xn-1, we would need to know x0 and all the Dxj<n. So I guess it does depend on all of those previous differences?

**AM(2) model**

This guy looks like this:



Again, we have the solution right here.



We can work out the covariance,



So,



This is a matrix with 1 + θ12 + θ22 down the diagonal, θ1(1 + θ2) on either of the diagonal, and θ2 on either side of *that*. So this is also stationary. Can see ACF(0) is 1, as it should be. The probability distribution of xn should be:



What about the conditional distribution? Well P(xn|x0, x1, …, xn-1) would seem to depend on all those preceding coordinates, like the MA(1) guy. Consider the first four points, and presume the first two white noises are zero, so ΔW0 = ΔW1 = 0:



Can see that if we presume knowledge of x2, then we get ΔW2. Specification of x3 gives us ΔW3. Specification of x4 gives us ΔW4. And so by the time we get to x5, only ΔW5 is unknown. So generally, specification of x2, x3, …, xn-1 will have ΔW0, ΔW1, …, ΔWn-1 all specified, leaving only ΔWn unspecified. So xn will be completely specified up to the unknown ΔWn (which has variance DΔt). So we would say:



where we implicitly note that ΔWn-1 and ΔWn-2 can be/would be determined from the preceeding points: x0, x1, …, xn-1. Can we work out a formula, again?



Here’s the equations out to n = 5:



We can find the inverse of the matrix by doing row operations on the identity matrix – that thing from Linear Algebra. But it’s complicated; I don’t think it simplifies nicely. But it does look like ΔWn involves all previous xj’s. So we will just define K as:



but generalized to n-dimensions, and write the conditional distribution as:



It does look like the contributions from xj<<n will get smaller and smaller if θ1,2<< 1. The joint probability distribution would be given by:



And as in the previous model, we can write this as a Gaussian with the afore-calculated mean and covariance:



Still going,

**ARMA(1,1) model**

This is a combination of an AR and MA model! Order 1 would look like,



Borrowing from the AR(1) model, the solution would be:



with stipulation that ΔW0 = 0. Let’s work out some expectations. The average is:



And the covariance,



So I think I have:



And the variance is:



The probability distribution of xn should be:



What about the conditional distribution? Well P(xn|x0, x1, …, xn-1) would seem to depend on all those preceding coordinates, like the MA guys. And knowledge of the x1, x2, …, xn-1 would also determine ΔW1, ΔW2, …, ΔWn-1. So we could say:



It does look like the contributions from xj<<n will get smaller and smaller if θ1,2<< 1. The joint probability distribution would be given by:



And as in the previus model, we can write this as a Gaussian with the afore-calculated mean and covariance:



**ARMA(2,2) model**

This will look like:



The solution ought to be:



where,



and,



I’m too lazy to work out the mean (well that’s easy) and covariance matrix. But it’s straightforward as before. And from that we can get the probability distributions.

We could similarly find the exact solution to all ARMA(n) models, but it would be increasingly laborious. Statsmodels has all this anyway.

**Simplest Exogeneous Series**

A simple regression model would an example of an exogeneous series,



where fn is the exogeneous series. What is the autocorrelation function for such a series:



So yeah.

**Not as Simple Exogeneous Series**

For example, consider the following first order, linear, inhomogeneous series.



where pn and fn are known auxiliary series (I don’t know if either of them can alternatively be white noise increments, maybe both can?). Let’s solve for xn. In order to write this as a discrete derivative/difference we need an integrating factor, In. To figure out what that would be, we can use brute force. We want to multiply both sides of the equation by In so that we can express the LHS as the difference of something:



so it must be the case that:



which together implies that:



We can solve this equation:



I guess we’d want to take I0 = 1/p0. Then,



and furthermore,



And note g0 = 1. And now we’ll go back to our equation:



which is:



Could just say:



Let’s make sure:



So that’s looking good, and I think we can write:



And explicitly in terms of pj,



Maybe we’ll write this as:



So that’s cool! If fn were a source term + white noise, say fn = qn + ΔWn (we could also easily do weighted noise, with an n-dependent weight factor). Then our difference equation would be:



And the solution would be:



What are average and variance?



Generally, it looks like for stationarity we need |pn| < 1, and |qn| < 1. If we only care about the variance not growing with time, then it seems we just need |pn| < 1. This is the same conclusion we reached when looking at the AR(1) model, though, for that model we needed pn = p for all n. Let’s look at the difference.



Yeah, just differencing won’t necessarily help reduce the noise if it’s a complicated summation. A general *linear* exogeneous series could be obtained from the ARMA(n) model. We’d just replace all the φ’s and θ’s with exogeneous series: φ1n, φ2n, …, θ1n, θ2n, …, etc. That would not be explicitly solvable of course, except in very rare, contrived, instances.