Jim Paper: PhL 10.18.04
Gaussian Quadrature for Two Classes of Log Weight Functions

Section 2. Derivation of the Method

Start with (2.1) with the strange α-dependent weight function shown and quoted in (2.2). Then differentiate the integral wrt α and you get the first log integral of interest as in (2.3). If we could do the original integral by GQ, we get the result in (2.4). Then we can diff that result wrt α to get the result for the log integral. This sounds like a lot of work! The end result so far is (2.5).

Now back up. Why not just do the integral shown left of 2.5 directly using GQ? I guess the concern is that maybe - x_0 will be in the range (a,b) and you will be integrating over a branch point and then GQ does not work well. You would think that otherwise normal GQ would work OK. Well in fact Jim is interested in integrals where x_0 collides with one of the endpoints, as in the example of (1.1) where x_0 = 0. He has to keep x_0 along for the ride to get clean answers I guess. OK, I see the problem.

So here we are with (2.5) for our integral. The f and df/dx at the nodes we can compute presumably for any f(x). Our problem is to put in the right stuff for W and dW/dα and dx/dα. The rest of this long section computes these items.

He starts with W, and the computational form is 2.9 (the sum part). But how is Jim going to evaluate the $p_n(x)$ if he does not know what these polys even are? He does know the eigenvalues x_i. Oh yeah, the matrix TLQ thing returns the eigenvalues and eigenvectors, so yes, he has both. So W can be computed just by adding up the squares of the p's as shown in 2.9. Remember, we don't evaluate polys, we really don't know what the polys are (but I do), we just use the numeric numbers that come from the TLQ.

The next ingredient we need is dx/dα and he gets an expression in 2.14. This requires derivatives of the A and B coefficients, where are those going to come from? In fact, where are the original A and B going to come from? Seems to me he has to know these things in full detail (as functions of α) in order to proceed.

Next, he needs dW/dα and this comes from (2.15), but where is he going to get the dP/dα from? I think you have to set up a whole new Jacobi Matrix problem from 2.19, but there is an extra term at the end that has to be dealt with. No details are given. Similarly, I guess you have to set up a system from 2.20 so you can get numbers for the ϕ_n. I will assume this has been done. THEN Jim has all the numbers he needs to feed into the result 2.27. Lots of errors in this paper!

Section 3 on Generalized Laguerre. In this section, we select a particular g(x) and set $x_0 = 0$ and we can then directly apply the results of the previous section. However, Jim never really states this result anywhere, but we know that it is just (2.27).

Now, his main issue is that (2.27) requires evaluation of the derivatives of f(x). If you have perhaps experimental data (that is the only situation I can think of), then you won't know the derivative f'(x), and numeric differentiation always makes errors. In this case, Jim is going to develop a new method.

What he does for this new method is use the weight function 3.9. Yes, this is still singular at x = 0 which is one of the endpoints, but he claims that he can do GQ directly with this weight function (no need for all the derivative stuff). So, assume that this new weight function x - 1 - lnx has a set of ortho polys Λ. They will have some recursion as in 3.12. As a side track, he presents a method of actually finding the recursion coefficients using a Stieltjes method, but this seems a bit off key. The idea is to continue and find the new nodes y_i with weights Z_i and for the new problem with the new w(x). Since this is direct GQ, you don't end up with that df/dx stuff in your result. The result is now 3.15. Notice that you really have to solve two problems here, one with nodes x_i and the other with nodes y_i. The first is with the

function (x-1) inside the integrand, and you do regular GQ on that to get the x_i. Then the y_i come from the messy problem. It is not really clear to me how he knows that the new singular weight won't have some kind of log endpoint problem when GQ is attempted. He cannot just use ln(x) as part of the weight function since it is not pos def.

To summarize, for the gen Laguerre's, Jim has two methods: (1) use the results of section 2 with $x_0 = 0$ and with $g(x) = \exp(-x)$; (2) use the fancier weight function and this avoids df/dx knowledge.

Section 4 on Jacobi. The integral shown here can also use Section 2 results, this time with $x_0 = 1$ and with $g(x) = (1-x)^{\alpha}$. Jim has told us to now think of β as the d/dβ variable, to maintain standard notation. Fine. You can then use the result 2.27, and the supporting info is computed.

As in the previous case, to avoid needing df/dx he does a new weight function with direct GQ on it. The result is now 4.11, analogous to 3.15.

Now lets read the "other text" -- abstract, intro and conclusion.

In the abstract he notes that he never needs to actually write down the polys, though he does that in Sections 3 and 4 anyway. He suggests that using the ugly log weight functions you can generate some "non-classical" ortho polys which I guess means the Λ things. Not sure which def of non-classical he means, having read Erdelyi.

In the introduction he does point out that log singularities at endpoints is the problem. He then states the two integrals he is interested in.

Conclusion speculates on higher powers of log.

Refs are: Wilf, Stieltjes and Gautchi book on ortho polys 1990.

Comments: The paper would be worth more I think if Jim actually used his method to compute something and could quote some results, the way Gautshci does. These results make us more believers. As is, the paper is just a proposal for a method that may or may not be numerically robust. I don't think this paper has been published, it seems to be in a preliminary state with all the spelling errors. I will have to ask.