A conversation with John (Jack) Ferguson about Lloyd Welch and the history of the Baum-Welch algorithm

Background: I wanted to learn more about the history of the Baum-Welch algorithm. Leornard Baum died in 2017, but I learned from Alfred Hales, a close friend of Lloyd Welch, that Lloyd was alive (age 96) and living in Southern California. He told me that Lloyd is in fairly good health but has lost his memory. Alfred suggested that I talk with Jack Ferguson, a friend of Lloyd's who worked along side of Lloyd when Baum and Welch did their work. Unfortunately Jack has lost much of his hearing, and so was unable to do a zoom interview with me. So he kindly suggested that we have an email conversation.

Sadly, Lloyd passed away on December 28, 2023.

What follows is an (edited) version of my conversation with Jack Ferguson.

Oct 31-Nov 12, 2023

Jack: I am pleased to be able to explain some of the history of Hidden Markov Models and, in particular, the Baum-Welch Algorithm.

Rob: Can you tell me the history of the development of the Baum-Welch algorithm?

Jack: I will explain what Welch did, in 1963, when he was working at the Communications Research Division of the Institute for Defense Analyses, in Princeton, New Jersey. This was abbreviated CRD, at that time, and renamed as the Center for Communications Research, abbreviated CCR, much later.

I began employment there in June of 1962, just in time to be able to attend Welch's talks on his ideas. I blame Yale (my school) for my not being well prepared in statistics, probability, or computer programming, none of which were taught in the math department at Yale, as far as I knew.

Welch started at CRD in 1958 or maybe 1959, not sure which. Baum in 1959, I believe.

At that time, CRD had, as a Director, an eminent mathematician, for a one or two year period. First, J. Barkley Rosser, President of SIAM. Second, A. Adrian Albert, President of the AMS. Third, my professor, Gustav A. Hedlund, Vice President of the AMS, who was Director in 1963 and 1963. I may be off by a year on that.

In any case, after Hedlund, CRD was unable to obtain the services of such eminences, and resorted instead, to elevating the Deputy Director, Richard A. Leibler, of Kullback-Leibler fame, to Acting Director, and then, later, to Director.

This sets the stage for the period when Welch created the reestimation algorithm that is now called the Baum-Welch Algorithm. In brief, Welch created an algorithm with a heuristic justification, and Baum formalized the algorithm in mathematical terms, and replaced the heuristic justification by a proof that reestimation always produced a more likely model. In other words, Welch's algorithm was actually a 'growth transformation.'

Rob: can you tell me more about Lloyd Welch and his accomplishments?

About Welch and his power. I admit that I am quite biased in his favor, having worked directly with him, and, later, intermittently with him, and from his papers.

When Welch started at CCR, there was interest in the idea of a function on a shift register. This was an unclassified topic, and several mathematicians were invited to come to CCR for summer projects on this, including Andrew Gleason, and, my PhD advisor, Gus Hedlund. At some point, I suppose 1958, Welch joined the research staff and proceeded to prove most of the important results in the field. In fact, in 1960, Welch proved, using combinatorial methods, the same result that Gleason proved, independently, using ideals in semi-groups of matrices, about the possible structures of all inverses of sequences under an onto shift register function. For reference, see Hedlund's lengthy paper in 1968.

Not long after this, he invented the idea of, and wrote programs for, an iterative method to estimate the parameters of a hidden Markov Model. About the history of this: the idea of an HMM had been invented before. Specifically, the well-known Russian engineer R. Stratonovitch, wrote about Conditional Markov Processes, in the signal processing field. He meant an HMM, but he viewed the Markov process (unseen) was conditional, given the observed data sequence. Stratonovitch described the forward, or alpha calculation, for these models. It was equivalent to Welch's later formulation. Stratonovitch also described the backward, or beta calculation, which yields the impact of all future observations on the state at a time t. Same as Welch's beta calculation. Stratonovitch also noted that a normalized product of an alpha vector at time t, with a beta vector at the same time, yielded a gamma vector, which gives the posterior probability of each state at that time t, conditional on observing the entire observation sequence. Same as Welch's gamma calculation.

But Stratonovitch assumed that the model parameters were known, exactly, and had no notion of a learning algorithm when those parameters were unknown. That is what Welch invented. Welch had never seen Stratonovitch's paper.

After inventing this algorithm, Lloyd decided to pull up roots and head for California. His wife, June, liked California lots more than Princeton. (There are East Coast people and West Coast people!). He went to JPL, and, I think, simultaneously joined the faculty at USC, in the EE department. Welch was trained in electrical engineering. I am unsure about which came first, JPL, or USC, because I continued on at CCR for nearly twenty-five more years, before joining CCR-La Jolla, a sister organization to CCR-P, so we were a continent apart.

At JPL, Lloyd worked with Solomon Golomb on linear feedback shift registers. See Golomb's book called Shift Register Sequences for Welch's many contributions to that field.

Meanwhile, at USC and JPL, Welch took up information theory and produced outstanding results, culminating in his selection for the Shannon Award for his contributions. I am not versed in what Lloyd did, exactly, in this area, and can only mention an important paper with Howard Rumsey and Eugene Rodemich which contained a famous inequality. You can, I hope, consult Welch's acceptance speech on receiving the Shannon Award. He talked, VAGUELY, about how he developed the Baum-Welch algorithm, and how Baum got involved.
If that is unavailable to you, I know that Al Hales has a printed copy, as do I.

I would also suggest consulting Al Hales on the actual information theory results Lloyd obtained.

At one point, USC voted on the most important scientific, or perhaps computational, results obtained by USC people. Ranked first was Cooley-Tukey's Fast Fourier Transform. Ranked second was the Baum-Welch Algorithm. I talked to Lloyd about this and he said: I figured out the FFT at the same time, and wrote it up, and was ready to submit it, but Cooley-Tukey appeared. Of course, others such as I.J. Good (also, for a time, a staff member at CCRP), produced versions of the FFT.

Last point for today: I once asked Baum about his estimate for Welch's ability. He gave a one-word answer: Kolmogorov. (I assume you are old enough to know what that meant! When I tried that on some whipper-snappers a few years ago at CCR-La Jolla, I had to explain who Kolmogorov was!)

Anyway, enjoyed writing to you about one of my heroes. Next time, about Baum, his contributions to HMM theory and implementation, and your other questions.

One final remark though. Welch and Baum never produced a joint unclassified paper. I say no more about OTHER joint papers!

Jack: Hi, again

I neglected to mention one other advance that Baum made. He showed that the BW reestimate could be expressed in terms of the gradient of the P function, with respect to the individual parameters.

Thus, for example, the reestimate for a_{i,j}, which we denote as â, can be written as:
â = a_{i,j} partial P/ partial a_{i,j}, normalized to be a probability distribution. Briefly, we often write x-hat = x dP/dx, normalized, where x is a catchall for 'any parameter.'

Finally, you will have noticed that work was often done at CCRP, but not published. Each published result had to pass government censors. It was fortunate that Baum succeeded in that.

Rob: who coined the term "Baum-Welch algorithm"?

Jack: I do not know who coined it. Likely, it was Lee Neuwirth, who coined the expression Hidden Markov Model.

As mentioned, Stratonovitch used the term Conditional Markov Process. The IBM automatic Speech Recognition group used the expression Markov Source Model. Baum called it: Probabilistic Function of a Finite State Markov Process. Neuwirth was an early researcher at CCRP, and originally a knot theorist under Ralph Fox at Princeton. He was later Deputy Director under Leibler at CCRP, then Director himself. He had a cover article about knots for the Scientific American Magazine, at one point.

Later, after discussing Baum, and his contributions, I will talk about the spread of use of HMMs and BW, and wind up with a discussion of EM.

Now Baum. Leonard Esau Baum was Phi Beta Kappa and Summa Cum Laude at Harvard, graduating about 1954. My Yale Class was 1957, so Baum was perhaps three years older. Baum was an Honorable Mention in the Putnam Prize examination one year, as was I.

After Welch had developed his algorithm to estimate the parameters of an HMM for a particular problem, Baum (and I and others) started working on another, different, problem. We found that we could apply the same method to our problem too. This was reported in Welch's Shannon Award Lecture. But still, there was only a heuristic argument as to why the algorithm was a good idea. I will discuss that.

As mentioned, Stratonovitch assumed that he had an exact model— all parameters known. In our cases, and in many others later, the parameters were either partially known or even completely unknown. So, people often initialize the iterative search for the most appropriate model with a fully random assignment of the parameter values: the initial probabilities of each state, the transition probabilities for the Markov chain, and the output probabilities for each state. Then we run the first iteration of the BW algorithm. This provides an initial score, that is, the log probability of the observed data, over random, based on the initial, hopelessly Inaccurate assignment of parameters. But BW also gives, for each parameter, via the Gammas, an estimate of how often this parameter was used during the production of the observed data. Take, for instance, a transition parameter, a_{i,j}, the probability that state i will transit to state j. We have, for each time t, the gamma estimate, Gamma_t (i, j), the posterior probability, given the data, and the faulty model, that, at time t, state i actually transited to state j. Welch argued, heuristically, that replacing the current value of a_{i,j} by the value sum_t Gamma_t (i,j) / sum_t sum_j Gamma_t (i,j) should produce a better model, since it took into account the entire observed data.

Baum agreed, heuristically, but wanted a proof. He devoted the next couple of years to that. First, he formalized the technique in terms of a hidden Markov Model, with parameters a_i, a_{i,j}, b_j (k), where k is one of the finite list of possible output symbols. This was Baum's notation.

He observed that the probability of the observed sequence of data could be expressed as a polynomial, in fact, a homogeneous polynomial (all terms having the same degree). Of course, for evaluation, this is a hopelessly inefficient way to calculate the probability, because it would require work that is exponential in T, the length of the data sequence. The alpha calculation, or the beta, require work that is LINEAR in T, namely, about TxSxS, where S is the size of the state space.

This suggested to Baum that, if this algorithm was going to work, it should work for any homogeneous polynomial with non-negative coefficients. This was tested. Baum never programmed, so he used one of our designated programmers to generate many homogeneous polynomials of various degrees etc. Sure enough, for all cases, and for every iteration, the score increased. A word was coined: a 'Posynomial' is a polynomial with non-negative coefficients. Of course, we never write the terms with zero coefficients, so all the written terms will have positive coefficients. This word never caught on, I think.

This raised a theoretical issue: computer says it works, but exactly why does it work?

Actually, this idea that BW is a growth transformation is not true, when implemented with finite precision computers. The limiting behavior of a computer implementation with finite precision is a limit cycle, with small ups and downs in the score. The goal of the theorem would be — with infinite precision, BW is a growth transformation.

Baum proposed this problem to our summer visiting mathematicians for a year or two. Finally, a proof was found by Jack Eagon, who was from Illinois, if I recall. This proof was complicated, and non-intuitive, utilizing, if I recall, the geometric/arithmetic inequality, as well as Jensen's inequality. The result was published in the open literature. So —BW is a growth transformation.

The restriction to homogeneous polynomials was seen to be irrelevant, via some sort of slack variable argument, details of which I don't recall.

In any case, later proofs were much clearer. The best is probably Baum, Petrie, Soules, Weiss, 1968, I believe. That paper introduces the auxiliary Q function and shows that maximizing the Q-function (which is what BW does) always increases P, the probability.
This function was used in Dempster-Laird-Rubin, in their EM paper.

Enough of Baum and his contributions. The bottom line is Welch — heuristics, Baum, formalization and provable effectiveness.

Rob: what the reaction of Baum and Welch to the EM paper of Dempster, Laird and Rubin? Baum's discussion of that paper seems to imply that Baum and Welch had already done much of what appeared in the EM paper

Jack: As for EM, I can answer directly. Dempster, Laird, and Rubin used personal terms, amounting, perhaps, to insult, to suggest that Baum and company had missed the boat. In any case, Baum felt insulted, so responded with some severity.

As for the technical difference between them, it is probably true that EM covers more territory, but recall, from my last, that Baum proved the result needed for arbitrary posynomials. EM seems to cover arbitrary probability models. Is it true that every posynomial has a corresponding probability distribution? I do not know.

Moreover, Baum, and company, were concerned about modeling sequences of data, hence the emphasis on HMMs. On the other hand, as far as I know, every single example cited in Em paper involves a mixture problem. We considered mixture problems as a trivial special case of HMMs, where the Markov order is 0, so that observations are independent.

Finally, yes, as mentioned in previous, much work went on to generalize the applicability of BW, but was not necessarily published. Much of it did extend in the direction of arbitrary probability models, as long as one could maximize the Q function, to obtain a reestimare.

One simple result, mentioned by Welch in his Shannon lecture, and not found in the EM paper, is that if one can INCREASE the Q function (without necessarily maximizing it) the reestimate will increase P.

Rob: Dear Jack, thank you so much for sharing your memories and insights with me.