October 12, 2012

I don't like that I'm comparing predicted z coordinates to other predicted z coordinates. I found an experiment that *measured* orientation, so we have something to check Daniel's algorithm against, but I don't understand it. I trust Daniel and Vik, but I can remove the need to trust them. Just do what Vik said: instead of zdiff, just measure *fraction predicted correctly*. Vik successfully looked at my results and saw that being one residue off pretty much makes something purple anyway: we were really just looking at where the mis-predicted residues were.

Mis-predicted? What's the *right* answer?

So here's the new basis for the testing. The procedure is dot renaming. If you know both of the structures, you can approximate some property of interest of one protein pretty well by dot-renaming of the other. Knowing both structures, you can find the *optimal dot-renaming* that minimizes the error in your approximation of that property. In my case, this was z-coordinates.

But let's get a little more general. The optimal dot-renamings for different properties are probably the same. The correspondence between the structures of related β-barrel membrane proteins is deep. I don't exactly understand the reasons for it. But big pieces of the structure are actually pretty much exactly the same. In these regions, the optimal dot-renaming doesn't vary much as a function of any geometric property that I can think of, though you may be able to construct exotic ones. So I can say that there's a dot-renaming that's *correct*, in the sense that it looks right and it's the optimal one for most properties. So, it doesn't really matter what property I use, and that frees me from needing to know the real z-coordinates.

(case study: 1a0s and 1af6. Sequence identity like 27% in a structural alignment according to TopMatch. But if you look at the alignment they follow each other very very closely - although there are insertions in the interior loops. But... what's the sequence identity in the *strands*, specifically? How much can those really change? What do they converge on if they maintian their shape? What's it take for them to *change* their shape? With Ezβ you can model the changes, and find the sequence identity of two randomly changing strands with the same structure but no common ancestor. What properties of beta strands does such a model find *unlikeliest*, and are they interesting?)

But what am I doing? I'm trying to approximate the dot-renaming that you get from a *structural alignment*, and renaming the dots based on what's closest, because I can see, visually, that you get something pretty much like the dot pattern where you got the names from (the "target structure", or "prediction target"). But what property is being predicted?

Okay. Then *that's* what I should start with. The observation that if you do a structural alignment and dot-rename based on closeness, the structure doesn't change much; this is the exact sense of the "structural similarity" that I say we observe. So, I can get something close to the original structure after structure alignment just by dot renaming; something with similar coordinates. It's easy to do this if I know both structures, and it doesn't matter *exactly* what I optimize for, RMSD or zdiff or whatever other measure; if the coordinates are similar, the structures are similar in many different ways. I can use a lot of different similarity measures, and maybe get slightly different results, but really, they are similar shapes.

Now, the question becomes: whatever I define as the best dot renaming, can I also get that dot renaming *without* already knowing the target structure?

For my own project, I'd rather not do any more work refining these alignmnets now that the zdiff's establish that they more or less work. If it weren't for Daniel's project I'd do that *after* I get my moment results. It's the advice "fail quickly". If it doesn't pretty much work with these alignments, it won't work with any alignments. If it was just my own project I was worrying about, I'd see if I get any interesting moment results with these mediocre alignments before I bothered improving them. Anyway I think they're good enough at least for that.

But it's not just the zdiff's that establish this, is it? We did this because we had a high prior probability that the alignments would make good dot renamings. And the comparison on five different structures is... well it's a lot of evidence I think, but it's not *so* much is it? (I love the relationship Ronny told me between the prior and the likelihood ratio of something you believe. If I think there's a fifty percent chance these alignments are good, then my likelihood ratio is the inverse of my prior - so the weaker the evidence is, the stronger the prior has to be to support the same results.) If two different alignment methods give different results, but not so different, then I'd go based on the priors, though I don't have a derivation that hsows that it's best it's cerainly what I'd do.

It's also important to determine if, if the alignments do *not* produce the right dot renamings, if they would still look like they do in the 5 comparisons, conditional upon the stuff that made us think the alignments might wokr in the first place. If whatever made us think the alignments would work also implies that they will work in our tests *even if* we're wrong, then the tests are worthless. THat's not the case, is it? What are our prior reasons for thinking the alignments wil be close to the correct dot renamings?

I don't really want to refine these alignments. I'll only make refinements that I know will help. Especially because if I *don't*know that they're going to help, then the 5 comparisons won't be *terribly* useful in establishing that they do, no matter what I see I mean there are only a few misalignments in the current test set. It's not too hard to eliminate those by luck, while introducing the same number of misalignments in proteins *outside* the test set, is it? (Is it? There're so many residues in a protein, this might actually be very unlikely - three proteins might be fairly representative)

Proteins with similar structures have common ancestors. That's what we think: that two versions of the same protein will mutate, but the accepted mutations will keep the structure the same. I don't know why. It confuses me. Why would their function be so delicate? Why's the tilt of the strands really matter? Do *all* mutations maintain the structure or just accepted ones?

You know, if there were changes in the tilt of the axis relative to the membrane normal, but no changes in the tilt of the strands relative to the axis, we'd never notice. The strands should get a little longer or shorter, but we can't really tell where they end. Especially not *before* we derive Ezβ.

It's pretty eeasy to see why strand number doesn't increase because of small insertions and point mutations on the loops. Where you gonna keep that hydrophobic thing as it gets larger, until it's large enough to thread through the membrane?  
But then... why aren't there insertions of whole β-hairpins every once in a while?

HHOMP clusters are measures of estimated evolutionary relatedness based upon a model of evolution that assigns too low a probability of the insertion of a β-hairpin. It's not just that related proteins are structurally similar. Also, related proteins that are significantly structurally different are hard for us to recognize as related. In penalizing insertions based on size, in a way that does not, as far as I know, have anything to do with the distribution of sizes of insertions; and in seeing any insertion as the same probability, whether or not it looks like a functional piece of another protein; the CLANS clustering done in HHOMP also selects for structural similarity, in addition to evolutionary relatedness. This is good.

So. What happens in the PAM model of evolution that the sequence alignment is based on?

There are two proteins, each a set of named dots. Every once in a while, a dot will change name. Also every once in a while, a group of dots will be added or removed. Over this time, the coordinates drift very slowly, except when dots are added or removed, in which case they change suddenly.

This mental image defines something important: *evolutionary correspondence*. And because the structures shift slowly, evolutionarily corresponding residues will be close together in a structural alignment, barring insertions or deletions. INsertions or deletions in the strand region will screw this all up.

In this model of evolution, the sequence alignment is a maximum likelihood estimate of the evolutionary correspondences, which themselves are good estimates of the intuitive structural correspondences that you use when you do the dot renaming manually from a structural alignment.

So. That's why this works. How can I improve the estimation of the evolutionary correspondences?

Imagine if there were two groups of proteins in a cluster: those that had a two-residue insertion in the middle of strand 1 since their common ancestor, and those that didn't. If the template didn't, then I want to get rid of the ones that did. In general I want to remove any structures that have gaps in the alignment when aligned with the template.

If I want to calculate fraction of residues correctly aligned, then I don't actually need to work with the structures directly. I can just check the alignments against the structural alignment produced by TopMatch. This is a metric by which, sometimes, a worse alignment method will get a better score, since if there are insertions or deletions in the strands, then the structure-based sequence alignment will actually be different than the correct evolutionary alignment.