Subject: Re: Confirming Sali values generated in Datawarrior / Machine learning and multiple descriptors in DW
Posted by thomas on Thu, 23 Sep 2021 11:41:14 GMT
View Forum Message <> Reply to Message

If you have multiple values in a cell, DataWarrior calculates and uses the mean (unless you have defined the cell to use the max, min, or median). If the columns is defined to be logarithmic (as useful for IC50 values), then the mean is a geometrical one, i.e. it is calculated on the logarithms. Evidently, your activity columns are defined to be logarithmical. Thus, activity1 used for the SALI is the mean of the logarithms. Original values: 116, 10000, 3162, 10000, 20598, ...; log values: 2.06, 4, 3.5, 4, 4.31, ...; mean of logs: 3.57 (probably different because of more values). log of activity 2: -0.537; delta activity: 4.11 (slighly different, because of values I cannot see in your screenshot). SALI = deltaActivity / 1-sim : 270.

The SALI value makes most sense with pIC50 values rather than activities, since the SALI considers differences of activities. If you have two compound pairs with activities 0 and 50% vs 50 & 100%. It is obvious that the first pair is rather different in the biological effect, which the second is almost the same (factor 2 of activity is not that dramatic). Therefore, activity differences are a bad foundation for SALIs. When using pIC50 values (or IC50 values with the column defined as logarithmic) a difference in 1 is always a factor of 10 and the delta activity value is independent from the numerical activity itself.

For the second question: consensus descriptors can only be useful, if multiple descriptors are equally well suited for a problem. If you dilute a good descriptors similarity with a weaker one, the result will be worse. In our case the SkeletonSpheres is superior to the other binary chemical descriptors (FragFp, PathFp, SphereFp) as far as chemical similarity from a medchem point of view is concerned. Thus, for SALI, you are save to just use the SkeletonSpheres. For machine learning purposes the situation is a little more complicated. Provided that you had really lots of diverse training data, then good methods will find the most predictive patterns from the input data. Then multiple descriptors would not be a problem and could provide more useful input. In reality, however, training data is noisy, biased and limited in size. And in my opinion chemical descriptors just don't have the information needed to predict ligand to protein interactions. Thus, what you do is basically learn your training data and be able to make rough estimates for compounds that are rather similar to the training data. Here you over-train anyway and adding more descriptors just makes it worse. But this is just my personal opinion.