In their paper “Learning Internal Representations by Error Propagation”, Rumelhart et. al aim to find a “learning rule” for networks which have hidden units. A learning rule for networks that do not contain hidden units, called the “delta rule” had already been developed, so Rumelhart et. al aim to generalize the delta rule to networks that contain hidden layers. Then, Rumelhart et. al demonstrate how this generalized rule can be applied to networks in order to solve more complex problems such as the XOR problem.

One of the things I really enjoyed about this paper was how Rumelhart et. al rooted their derivations in already established theory by showing that their findings were simply generalizations of established theorems such as the delta rule. This is a good “sanity check” for both the authors and the readers, as it shows a certain level of robustness in the new theory. I also appreciated the periodic summary of results of the derivations because they allowed me to better understand the meaning behind the arithmetic. Another great feature of this paper is that it demonstrates the utility of the theory it proposes by outlining the results of some simulations run using the theory the paper develops. This helps the reader understand how they might use the theory the authors developed to solve their own problems.

One criticism I have for Rumelhart et. al is that I don’t feel that they did a good job of explaining all of their notation. Specifically, the geometrical meaning of a “pattern” was not clear to me. This could certainly be my lack of familiarity with the nomenclature, but I believe that providing some diagrams to support the notation would be beneficial, specifically for Equation 1 so that the reader can paint a picture of what the physical meanings of each symbol is without needing to be well versed in the nomenclature. Another criticism I have of the paper is that the authors do not express the calculations in vector form. In my opinion, the mathematics involved in neural networks lends itself perfectly to using vectors and vector notation. Expressing the theory using vectors gives a more complete picture of what is happening to the network as a whole. It also makes things clearer, as there aren’t as many indices floating around to confuse the reader. Using the vector forms of the equations in the theory would also give more insight into why the delta rule “implements a gradient descent”.

One inspiration I got from this paper is about the utility of vector calculus. I am a dual major in Physics and Computer Science, and while I use lots of vector calculus in my physics studies, I don’t use it very often in computer science. This paper helped me realize that I should be using my familiarity with vector calculus to look for problems in computer science that can be solved with vector calculus. Another thing this paper inspired me to do is to look more deeply at the back propagation equations. For time reasons, the equations for back propagation of errors were simply stated in lecture. This paper demonstrates how crucial back propagation is to the functionality of neural networks, and as such I would like to more closely examine the back propagation equations to see how they are derived.