There are four steps to complete **the perception of tic-tac-toe game**.

The fist step is reading the data from the file and store them as Point type, every point contains nine char features, x or o or blank and one string label, positive or negative.

Then I processed these points to make the data easy to percept. I created a new class ProcessingData, and store the sixteen features in an int array. The first eight features are depended on the number of x character. As shown in the picture, I set feature1 as the number of x in the fist row, it’s one. And the second, third row. And the feature4 is the number of x in the fist column. feature5 and feature6 applies the same. And feature7 and feature8 are respectively the number of x in the two diagonals. The last eight features are depended on o, which is the same as in x.

And then I set the positive as 1 and negative as -1 in the status.

The third step is to implement an algorithms which is fit for this perception. I adopted the perceptron algorithm, I get the weight of each features first. It is a type of online linear classifier, i.e. a classification algorithm that makes its predictions based on a linear predictor function combining a set of weights with the feature vector. in that it processes elements in the training set one at a time.

Then calculate the hypothesis and the error rate to get the final precision.

The final step is make some modification to improve the precision. At first, I have alfa about 0.01, the precision was about 50 percentage. And then I decreased it to about one over one hundred times. And also, I add a loop of repeating to calculate the hypothesis for 1 throunds time, it worked like off-line linear classifier. Finally I get the optimal precision up to 98 percentage.