1. Suppose that you are given a hypothesis for a particular domain, you test it on 100 independent examples and you observe 91% accuracy. Calculate a $N = 0.95$ confidence interval for the accuracy of the hypothesis. Use the two solutions given in class to the problem of unknown variance to compare the effect on the size of the intervals. Solution (1) uses the fact that $p(1 - p) \leq \frac{1}{4}$ and solution (2) uses $\hat{p}$ instead of $p$ in the term for the variance.

2. You run your favorite algorithm on a new dataset using a 10 fold cross validation scheme and get the following accuracies in the folds: 0.91 0.74 0.79 0.82 0.92 0.83 0.78 0.85 0.81 0.87. Use the formulation of the $T$ confidence interval to calculate two intervals for the average accuracy, using confidence of 0.99 and 0.95 respectively.

3. Consider classification with linear threshold units, where we predict Yes ($y = +1$) when $\sum_{k} w_k x_k \geq 0$ and No ($y = -1$) otherwise. Now consider a 2-dimensional space where the example (2,1) is positive and where (1, -3) is negative.

What is the range of possible $w$ vectors that agree with these labels? follow the example from class and draw the 2-dimensional space in terms of $w_1$ and $w_2$ identifying that space. Please make sure to explain your answer.

4. You train the Perceptron algorithm using an initial value $\vec{w} = (0, 0)$ and $\eta = 1$ on the following sequence of examples $[(-2,1), -], [(1,1), +], [(5,2), -]$ where each square parenthesis shows an example and its label.

To be concrete about the algorithm: We test and possibly update just one time for each example, and when $\vec{w} \cdot \vec{x} = 0$ the algorithm predicts + (i.e. it will update on a negative example and not update on a positive example in such a case).

(1) Calculate the updates of the weight vector on this sequence. Please make sure to show your computation.

(2) At the end of training you want to predict the label of the examples (2,3) and (3,1). Show the calculation of the prediction of the last weight vector, of the Voted Perceptron predictor, and of the Average Perceptron predictor on each of the two examples. Note that since we have 3 training examples we have 4 weight vectors participating in the votes of the voted perceptron and average predictor. Please make sure to show your computations.

5. Recall that in class we showed that the perceptron with margin (PwM) algorithm can be obtained as a gradient descent algorithm for the margin-error function (also known as the hinge loss). Explain how this is done and show the details of how the update rule of PwM is derived.