This homework is individual work. All assignments are due at the beginning of class on the due date. Assignments turned in late will drop 10 points for each period of 24 hours for which the assignment is late. In addition, no assignments will be accepted after the solutions have been made available. Please include your cornell net id on your homework.

A. Statistical Learning Theory

1. Error Bounds (20 POINTS)

Consider the instance space X of points $\vec{x} = (x_1, x_2)$ with integer coordinates $([0..10], [0..10]) \subset N^2$. The task is to learn a concept $c : X \rightarrow \{0, 1\}$ which can be described as a rectangle $((a, b), (c, d)) \in N^2 \times N^2$. (a, b) is the left top corner of the rectangle, (c, d) is the lower right corner. An example is labeled positive, if it lies inside the rectangle or on it's boundary. It is negative, if it lies outside. The hypothesis space H are all rectangles over the instance space X.

You have a training sample of size n and your learning algorithm found a hypothesis that has zero training error. Give a bound for the prediction error of this hypothesis!

2. Infinite Hypothesis Spaces (20 POINTS)

In class we only derived generalization error bounds for finite hypothesis spaces. For infinite hypothesis spaces, these bounds cannot be applied directly, since they contain the number of hypothesis $|H|$ as a factor. The key trick to extend them to infinite hypothesis spaces is to consider the "effective" number of hypotheses, which is finite. To illustrate this, consider the following example.

Our hypothesis space are all intervals $[a, b]$ on the real line (i.e. the instance space $X = \Re$). A point x is classified positive, if $x \in [a, b]$, negative otherwise. Clearly, this hypothesis space is infinite. However, many of these hypotheses are "redundant" for any given sample.

Consider a sample $S = (x_1, ..., x_n)$ of n (distinct) points. Derive a formula for the number of hypotheses that classify the examples *in different ways*. The bound should hold for any such sample.

B. Clustering

1. Hierarchical Agglomerative Clustering (20 POINTS)

Create by hand the clustering tree for the following sample of ten points in one dimension

$$S = (-2.2, -2.0, -0.3, 0.1, 0.2, 0.4, 1.6, 1.7, 1.9, 2.0). \quad (1)$$

Use single link clustering (i.e. $d(C_i, C_j) = min_{x \in C_i, x' \in C_j} ||x - x'||$). Based on the clustering tree, argue that three is the natural number of clusters.

2. K-Means as Greedy Search (20 POINTS)

The k-means algorithms (see Algorithm 1 in the handout) can be viewed as searching for the clustering $C_1, ..., C_k$ with mean vectors $\vec{\mu}_1, ..., \vec{\mu}_k$ that minimizes the following objective function, typically called the sum-of-squared-

error criterion J_e:

$$J_e = \sum_{i=1}^{k} \sum_{\vec{x} \in C_i} ||\vec{x} - \vec{\mu}_i||^2 \tag{2}$$

Prove that in every iteration of the algorithm (before convergence), the objective function J_e never increases. In particular, prove that in each individual step

- "classify n samples according to the nearest $\vec{\mu}_i$" and
- "recompute $\vec{\mu}_i = \frac{1}{|C_i|} \sum_{\vec{x} \in C_i} \vec{x}$"

the objective function J_e never increases.

3. Finding the Optimal k in k-Means (20 POINTS)

One shortcoming in k-means is that you have to specify the value of k. How about the following strategy for picking k automatically: try all possible values $k \in \{1, 2, ..., n\}$ and pick the k that minimizes J_e. Argue why this is a good / bad idea!