Problem 1

In this problem we relate the performance of three learning procedures (Procedure W, Identification Trees and Nearest Neighbor) on a data set of Arch descriptions. The training data is attached on a separate page at the end of the quiz – feel free to tear it off for handy reference. It consists of eight samples: four arches and four near misses.

Procedure W operates on models represented as nets, while both Identification Trees and Nearest Neighbors operate on feature vectors. To enable comparisons between the methods, we have devised a simple way of encoding the Procedure W nets into binary (yes/no) feature vectors: Each of the features denotes the presence of a particular link in the corresponding Procedure W model.

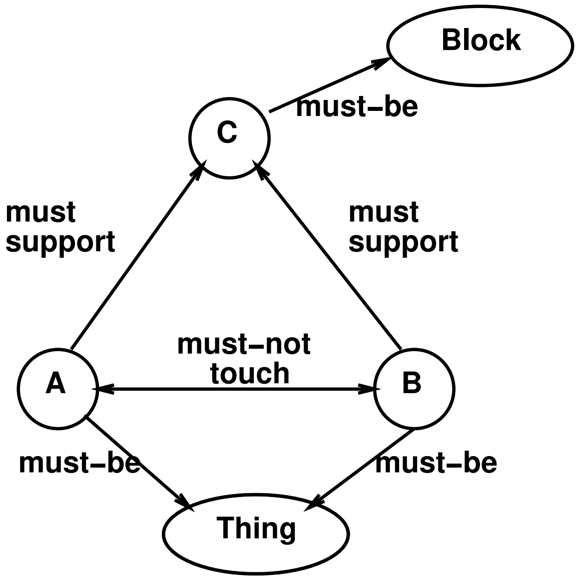
For example, the fact that A supports B in the sample is indicated by placing a yes in the `A-supports-B?` feature. Note that, in contrast to the text, we have assumed a symmetric `touch` relation and we don't have a `left-of` relation. We only show those features that are relevant to this particular data set; in general, many more features would be required.

Note that we have features indicating the classes (e.g., brick, cylinder) of the objects in the description. There should be a feature corresponding to each class in the AKO tree and a "yes" entry for the feature corresponding to **every** class that the object belongs to. We have not shown them all since they are not all relevant to this data set. Throughout this problem we assume the following AKO tree:

```
              Thing
             /     \
         Block    Curved
        /    \    /    \
    Brick  Wedge Cone Cylinder
```

Part A

Choose a sequence of samples from the data set given above so that Procedure W learns the following concept for an Arch. For each sample, indicate the induction heuristic used, drawing from the following list as given in Chapter 16: forbid-link, require-link, climb-tree, enlarge-set, drop-link, and close-interval. You may or may not have to use all the entries in the data set; the number of spaces below is not intended as a hint.



Sample number	Induction heuristic
1	initial model
5	require-link
6	forbid-link
2	climb-tree
3	climb-tree
4	climb-tree

The first 4 of these are straight out of the book. The last two serve to generalize the supports to "Thing."

Part B

In general, is there a unique order in which the samples must be presented for Procedure W to learn a particular concept? Briefly, justify your answer.

No, there are constraints on the order but it is not unique. For example, we could have shown the samples above in the order [1,6,5,4,3,2] or [1, 6, 5, 3, 2, 4], etc.

The most common mistakes were to say that order did not matter at all or that a unique order is required. Note that, in general, a different order may cause Procedure W to learn a different concept, e.g. showing Sample 8 after Sample 1. Also, note that the fact that only a single important difference is required between the model and the sample is not restrictive enough to determine a unique sequence.

Now we will build an Identification Tree for this data. In this problem, we will use the following simple measure of disorder for a set (assuming only two classes, arch and non-arch): Identify the most common class in the set (if it's balanced, pick randomly); then, the disorder is the following ratio:

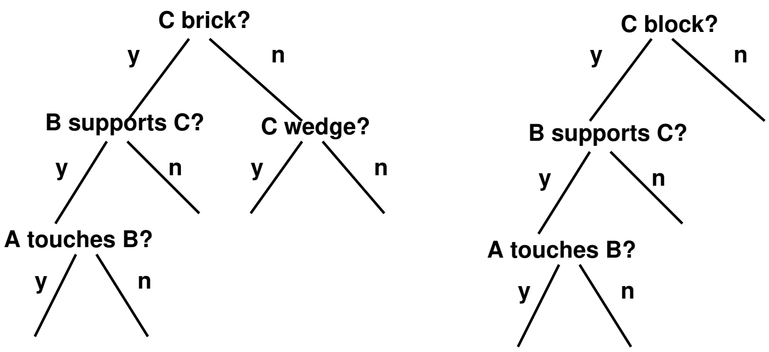
$$\frac{\text{number in less common class}}{\text{number in most common class}}$$

.

Note that this measure is zero for a homogeneous set and 1 for an evenly split set, which is precisely what we want. We will further assume that the disorder for a test will be the simple average of the disorder for the branches. In this question, you should be able to make most of the disorder comparisons by inspection.

Part C

Which of the following identification trees is consistent with this disorder measure and the procedure for growing trees described in the text? Explain briefly how you arrived at the answer. Hint: You should not need to do the disorder computations in detail.



The correct answer is the one on the right, with "C block?" as the top node. Note that "C block?" picks out a homogenous 2 negative sample set (7,8) on its "n" branch and so it has a disorder score of 0 on that branch. On the "y" branch it has the rest – 4 positives and 2 negatives – and a disorder of 1/2. So, its average disorder is 1/4. In contrast, "C brick?" has 1 positive and 2 negatives on it's "n" branch (disorder=1/2) and 3 positives and 2 negatives on it's y-branch (disorder=2/3) for an average disorder of 7/12.

Note that simply saying that the one on the right was better simply because the first test picks out a homogeneous set and the other didn't is not a guarantee that it is the correct choice. In general, one has to compare what happens on the other branches.

Part D

Can we always build an Identification Tree from one of Procedure W's models? If so, explain the relationship between the models learned by Procedure W and an Identification Tree. In particular, indicate what corresponds to a required (must) link and a forbidden (must not) link in the Identification Tree. If not, explain why not.

Yes. Required links (must) indicate that all the positive samples are down the "y" branch of a test for the presence of the corresponding feature and forbidden links (must-not) indicate that all the positive samples are down the "n" branch of the corresponding feature. This is illustrated by the right-hand tree in Part C.

There is one slightly sticky point here because the enlarge-set and close-interval heuristics create sets that are not represented as AKO tree classes and so, strictly speaking, we need to define features for every subset of classes and every interval. But, in principle, this is doable.

Part E

Can we always build a single Procedure W model from an Identification Tree built from the type of feature data we have been considering? If yes, explain why. If not, explain why not.

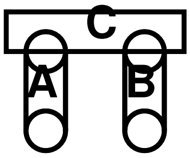
Not always. If all the positive samples end up at the same leaf of the Identification Tree, then we can build a Procedure W model. If the positive samples are split among the leaves (see the left hand tree in Part C) then no single procedure W model can represent them. Procedure W models represent an "AND" of a set of conditions. Each such model could be represented by a single IF-THEN rule. An ID tree, in general, can represent an "OR" of models – requiring multiple IF-THEN rules.

Many people answered No to this question because they were concerned that the ID tree would not have all the same information that the "ideal" Procedure W model would have. Several points need to be made: One is that the question did not ask about a particular "ideal" Procedure W model but simply "a" model. The other is that generally information that is important for separating positives from negatives in the training set will appear in the ID tree. The two procedures have different "biases" and will therefore learn different representations from the data but, in this question, we were focusing not on the learning procedures but on the representations that they learn.

Suppose that we wanted to use Nearest Neighbor classification on the same Arch feature data. A natural metric to use is the, so called, Hamming distance — a count of the number of "bits" (yes/no features) that are different between two complete feature vectors.

Part F

Consider the following sample:



How would this sample be classified by a Single Nearest Neighbor procedure that used the Hamming distance? Would it be an Arch, Not-Arch, or is there a tie? Which sample or samples in the data set are nearest?

There is a tie between samples 3, 4 and 8. All involve changing one component from a brick to a cylinder. The Hamming distance is 4 for each of these. Note that because both Brick and Block are always true together and that Cylinder and Curved are always true together, the actual distances are higher than one would expect.

Part G

Is the answer from Nearest Neighbor using this metric consistent with what Procedure W or the Identification Tree in Part C would give? Explain the difference, if any.

This sample is unambiguously an Arch both for the Procedure W model and the Identification Tree. But, the Nearest Neighbor procedure has no idea that the difference concerning whether C is a Block is more important than the difference concerning whether B is a Block.

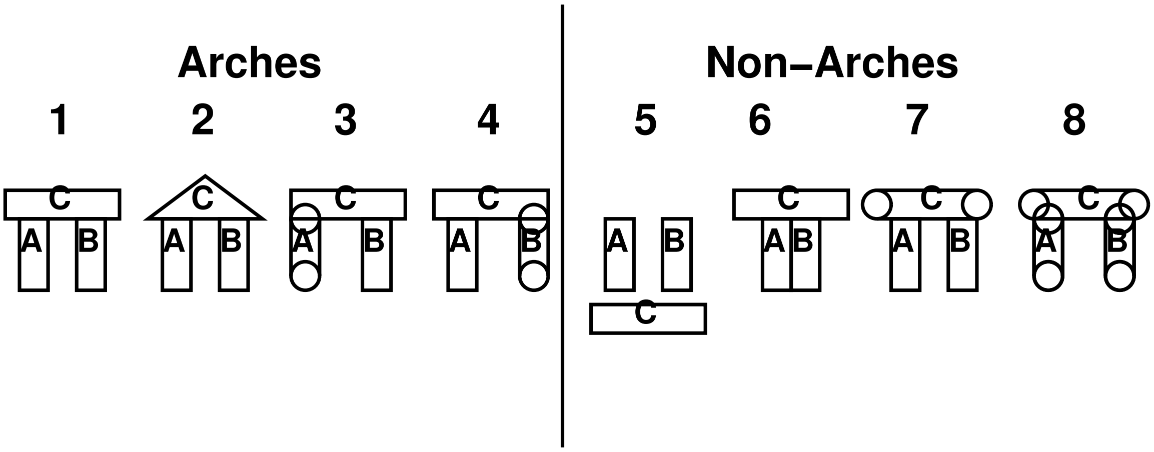
Part H

Based on the Identification Trees of Part C, suggest a variation to the Nearest Neighbor metric used above, that might improve the performance of Nearest Neighbors in this domain.

We can use the Identification tree to choose which features are important and then use Nearest Neighbor. So, "C block?" would be checked but not "C brick?" or "B brick?"or "B block?". Then samples 3 and 4 would be unambiguously closer (Hamming distance of 0) than sample 8 (Hamming distance of 2) – as it should be.

Several people suggested using a voting scheme (i.e. k-Nearest Neighbor for k¿1). This is not a change in the metric; it is a change in the algorithm and wouldn't be as reliable.

Training Data for Problem 4



	1	2	3	4	5	6	7	8
A supports C?	y	y	y	y		y	y	y
B supports C?	y	y	y	y		y	y	y
A touches B?						y		
A brick?	y	y		y	y	y	y	
A block?	y	y		y	y	y	y	
A cylinder?			y					y
A curved?			y					y
B brick?	y	y	y		y	y	y	
B block?	y	y	y		y	y	y	
B cylinder?				y				y
B curved?				y				y
C brick?	y		y	y	y	y		
C wedge?		y						
C block?	y	y	y	y	y	y		
C cylinder?							y	y
C curved?							y	y