8. The example in the text regarding using the k-nearest neighbor algorithm for estimation has the closest record, overwhelming the other records in influencing the estimation. Suggest two creative ways that we could dilute this strong influence of the closest record.

The example in the text has the closest record overwhelming the other records in influencing the estimation. However, sometimes classification based on the closest record may not be the best approach. When geometry does not align well with domain expertise, we can mitigate the overpowering influence of the closest record(s) by having the algorithm concentrate on the most relevant predictors. There are actually two creative ways to do this.

The first of these ways is to limit the distance calculation to using the relevant predictors only or having the algorithm ignore less relevant predictors. This approach will prevent the learning algorithm from basing the distance calculations on predictors that have large yet less relevance.

The second of these ways is to retain all predictors in the learning algorithm, but determine appropriate multipliers to the more relevant predictors so they outweigh the less relevant predictors. This approach is referred to as ***stretching the axes*** and these multipliers may be determined directly by domain experts, or they can be data-driven using *cross-validation*. In cross-validation, a small subset is drawn at random and multipliers are chosen until the classification error on the training set is minimized.

9. Discuss the advantages and drawbacks of using a small value versus a large value for k.

Choosing the optimal value for k will most likely not be obvious and may require several attempts. If we choose a small value for k, then the model may be unduly influenced by outliers or noise that may lead to overfitting.

In contrast, if we were to choose a large value for k, then the model will be less likely to be influenced by noise, but it may cause the model to completely overlook the rare conditions generalizing the new observations to the larger mainstream classes.

11. What is locally weighted averaging, and how does it help in estimation?

Locally weighted averaging is an approach to estimating / predicting ***continuous*** values via a KNN model. It helps in estimation by leveraging the target values of an observation's nearest neighbors and calculating the target value of the new observation as a weighted average of the neighboring target values. A locally weighted average using KNN is formulated as follows:

Where $\hat{y}$ is the target to be estimated, y_i is the target value of each neighbor and w_i is the weight of the target value of each neighbor.

Solutions to Chapter 8
DECISION TREES

Prepared by James Cunningham, Graduate Assistant

1. **Describe the possible situations when no further splits can be made at a decision node.**

   When all branches of the decision tree terminate at pure leaf nodes, no additional splits are required. In this case, all records contained at each leaf node have the same target class value.

   One or more branches may terminate at diverse leaf nodes. The algorithm will attempt to split the records leading to a pure leaf node; however, if all records contain the same predictor values, no additional splits are possible.

2. **Suppose that our target variable is continuous numeric. Can we apply decision trees directly to classify it? How can we work around this?**

   To use a continuous target variable for classification, you must first discretize it. This process takes the range of values and coverts them into discrete categories.

3. **True or false: Decision trees seek to form leaf nodes to maximize heterogeneity in each node.**

   False. Decision trees algorithms attempt to create leaf nodes that are as pure as possible. This occurs when the leaf node contains the maximum number of records with the same target class value. This results in higher confidence, and increased classification accuracy.

4. Discuss the benefits and drawbacks of a binary tree versus a bushier tree.

A binary tree is limited in its ability to split on categorical attributes, and may result in sub-optimal classification accuracy; however, a decision tree containing binary splits is often more easily interpretable.