Predict whether each of the following people will vote for Bush or Kerry:

Mary lives in Berkeley; I predict she will vote for ____________
Joe lives in Midland Texas; I predict he will vote for ____________
Kate lives in Detroit, Michigan; I predict she will vote for ____________
Hal lives in San Diego; I predict he will vote for ____________

How did you make your predictions? What information did you use? In this simple exercise, you were using limited knowledge about the individual, namely their location and gender, to predict how they would vote in the upcoming election. In a way, you were doing something like nearest neighbor classification. That is, you were using your knowledge of typical voter in Berkeley to predict that Mary would vote for Kerry, etc.

Armed with the following additional information, update your prediction as to whether each person will vote for Bush or Kerry:

Mary lives in Berkeley and earns $25,000 a year; I predict she will vote for ____________
Joe lives in Midland Texas and earns $150,000 a year; I predict he will vote for ____________
Kate lives in Detroit, Michigan and earns $80,000; I predict she will vote for ____________
Hal lives in San Diego and earns $75,000 a year; I predict he will vote for ____________

Did your predictions change with this new information? Are you any more or less confident of your predictions? Why? Now you are using additional information to locate say Hal in a smaller group of people, those who live in San Diego with incomes of about $75,000. You may not have changed your prediction that Hal would vote for Kerry, but you are probably less sure of it.

**Key Concept**  The main idea behind nearest neighbor classification is that given a set of classified records, to classify a new record you examine those records nearest to the new one. If the majority are classified as type A then you classify the new record as type A.

Two questions arise:
• How do we determine which records are nearest to the new one?

• How many records do we examine in order to classify the new one?

Simple Example  We examine each of these questions in turn using a simple set of data. the values of $x$ are given below along with the associated class (G or R).

\[
x \]
[1] 0.0  1.0  1.7  2.5  3.0  3.5  4.0  5.0  6.0  7.0

\[
col \]
[1] "G" "G" "G" "R" "R" "G" "R" "R" "R" "R"

A new observation with a value of 0.3 would be classified as a “G” in nearest neighbor classification. For 2 and 3 nearest-neighbor, it would also be classified as “G”. On the other hand, a new value of 3.6 would be classified as a “G” in nearest neighbor classification, but when $k = 2$ one neighbor is G and the other R. What should the classification be? NA? or toss a coin? When $k = 3$, the classification switches to “R” as two of its three nearest neighbors are red. The 10th nearest neighbor classification for these two points will be “R”. Why? We see in this case, we are including all of the previously classified data and so the classification will always be the same – the majority classification, which is “R” in this case.

Fill in the table below with the classifications for each of the 7 observations.

<table>
<thead>
<tr>
<th>Value</th>
<th>0.3</th>
<th>2.0</th>
<th>2.8</th>
<th>3.8</th>
<th>4.2</th>
<th>5.2</th>
<th>6.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>$k = 1$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$k = 2$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$k = 3$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$k = 10$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Notice that we used Euclidean distance between the points to determine the neighbors. This makes sense when the data values are continuous, but what if they are factors? Also what happens when we have two or more variables? Then we may want to scale each of the variables in order that one does not dominate the others. These issues are discussed later.

Finally, how do we evaluate the performance of the different nearest neighbor estimators? If the truth is eventually revealed, we can compute the error rate among the new “G” observations, and the error rate among the new “R” observations. Fill in the table below with 0s and 1s according to whether the prediction is correct (0) or not (1). Then the sum of these values divided by the number of predictions gives us our error rates. For \( k = 10 \) the prediction is always “R”, so the “G” error rate is \( 3/3 = 1 \) and the “R” error rate is 0. What are the corresponding “G” and “R” error rates for \( k = 3 \)? Which choice of \( k \) would you use? More on this later.

<table>
<thead>
<tr>
<th>Value</th>
<th>0.3</th>
<th>2.0</th>
<th>2.8</th>
<th>3.8</th>
<th>4.2</th>
<th>5.2</th>
<th>6.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRUTH</td>
<td>G</td>
<td>G</td>
<td>R</td>
<td>G</td>
<td>R</td>
<td>R</td>
<td>R</td>
</tr>
<tr>
<td>Correct? ( k = 3 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Correct? ( k = 10 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>