# Solution to Practice Problems: Linear regression/classification and kNN

1. **Regression vs. Classification**

   For each prediction task below, indicate whether it is a (R)egression or (C)lassification problem.

<table>
<thead>
<tr>
<th>Task</th>
<th>Regression or Classification?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Predict the path of a meteor, in terms of its (X,Y,Z) coordinates as a function of time, based on satellite readings and information about other astronomical objects in the meteor's vicinity.</td>
<td>Each X, Y, and Z is a real-valued position in 3-dimensional space, so this needs to be Regression.</td>
</tr>
<tr>
<td>Predict the price of a flight to Jamaica tomorrow, based on the price in the past few weeks.</td>
<td>Typically this would be done using regression, although you can argue that money is not infinitely divisible, so it is discrete, and therefore this should be classification.</td>
</tr>
<tr>
<td>Predict whether your item on eBay will sell for more than $100 or not.</td>
<td>Classification (two possible outputs, true or false)</td>
</tr>
<tr>
<td>Predict the final sales price of your item on eBay, based on the sales prices of similar items.</td>
<td>Regression (or classification, see above)</td>
</tr>
<tr>
<td>Predict whether Temple's men's basketball team will make it to the NCAA tournament this year.</td>
<td>Classification (true or false)</td>
</tr>
</tbody>
</table>
2. Linear predictions

a. Based on the above data, approximately how much would a house cost if it is 1000 square feet?

A best-fit line for this data is shown above. The point where Square Footage = 1000 occurs when House Price is approximately $90,000.

b. Based on the above data, if my budget for buying a house is $150,000, what is the biggest size house (approximately) that I should be looking at?

The point where House Price = $150,000 occurs when Square Footage is approximately 1750.
3. Parameter Estimation

For this problem, you will come up with a linear regression model for the House Price, given the Square Footage of the house.

Notice that the data does not all fit on a straight line, so we will need to use the method of "least-squares" regression.

a. What is the Hypothesis Class (the set of all possible models) for this problem?

The Hypothesis Class is all functions of the form \( HP = w_0 + w_1 \times SF \), where \( w_0 \) and \( w_1 \) are any real numbers.

b. Use the "closed-form solution" to estimate parameters for your regression model.

\[
  w_1 = \frac{\sum_i X_{1i} Y_i - \frac{1}{M} (\sum_i Y_i)(\sum_i X_{1i})}{\sum_i X_{1i}^2 - \frac{1}{M} (\sum_i X_{1i})^2}
\]

numerator of \( w_1 \) = \[800*74000 + 1150*110000 + 1450*134000 + 1600*138000 + 2300*192000 + 2450*220600 - 1/6 * (800+1150+1450+1600+2300+2450) * (74K + 110K + 134K + 138K + 192K + 220600)\]

= \[1,582,870,000 - 1/6 * 9750 * 868600\] = 171,395,000

denominator of \( w_1 \) = \[800^2 + 1150^2 + 1450^2 + 1600^2 + 2300^2 + 2450^2 -1/6*9750^2\]

= \[17,917,500 - 1/6 * 95,062,500\] = 2,073,750

So \( w_1 = 171,395,000 / 2,073,750 = 82.65 \) dollars per square foot
\[ w_0 = \frac{1}{M} \sum_i Y_i - \frac{w_1}{M} \sum_i X_{1i} \]

\[ w_0 = \frac{1}{6} \times 868,600 - \frac{82.65}{6} \times 9,750 \]

\[ = 10,460.42 \text{ dollars} \]

c. Using your estimated model, come up with an exact prediction for the price of a house that is 1000 sq. ft.

\[ \text{HP} = (10,460.42 \text{ dollars}) + (82.65 \text{ dollars per square foot}) \times (1000 \text{ sq ft}) \]

\[ = 93,110.42 \text{ dollars} \]

Close to our "eyeball" prediction of $90,000.
4. Gradient Descent

Check the boxes for all points that will reach the global minimum, and all points that will reach some local minimum, if these points are used as the starting point for a gradient descent algorithm that is used to minimize the LOSS function with respect to weight $w$.

<table>
<thead>
<tr>
<th>Starting point for Gradient Descent</th>
<th>Will reach the global minimum?</th>
<th>Will reach some local minimum?</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>b</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>c</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>d</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>e</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>f</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>g</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>h</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>i</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>j</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>k</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>l</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>m</td>
<td></td>
<td>x</td>
</tr>
</tbody>
</table>

Gradient descent will always find a local minimum (provided the learning rate is small enough), but only the points in the "well" around point i will reach the global minimum. The other points will reach the bottom of their local "well".
5. Perceptron

a. Is the data above linearly separable?
   Yes, as shown with black line

b. If the perceptron algorithm is initialized with the line shown above, will it eventually find a linear separator? Yes, if the data is linearly separable, perceptron always converges.

c. Is the data above linearly separable? No, there's no line that can separate the data.

d. If the perceptron algorithm is initialized with the line shown, will it eventually find a linear separator? No, the data isn't linearly separable, so there is no linear separator for perceptron to find.
6. k-Nearest Neighbor

Using K=5, predict the color of points a, b, and c.
### 7. Types of Machine Learning

<table>
<thead>
<tr>
<th>Model</th>
<th>Classification or Regression?</th>
<th>Generative or Discriminative?</th>
<th>Parametric or Nonparametric?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bayes Net</td>
<td>Classification</td>
<td>Generative</td>
<td>Parametric</td>
</tr>
<tr>
<td></td>
<td>(from what you’ve seen,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>although it’s possible to</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>do regression as well)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Naïve Bayes</td>
<td>Classification</td>
<td>Generative</td>
<td>Parametric</td>
</tr>
<tr>
<td>Linear Regression</td>
<td><strong>Regression</strong></td>
<td>Discriminative</td>
<td>Parametric</td>
</tr>
<tr>
<td>Linear Classifier</td>
<td>Classification</td>
<td>Discriminative</td>
<td>Parametric</td>
</tr>
<tr>
<td>K-Nearest Neighbor</td>
<td>Classification (or regression)</td>
<td>Discriminative</td>
<td><strong>Nonparametric</strong></td>
</tr>
<tr>
<td>Algorithm</td>
<td>Supervised or Unsupervised?</td>
<td>Online or batch?</td>
<td>Closed-form or iterative?</td>
</tr>
<tr>
<td>------------------------------------</td>
<td>-----------------------------</td>
<td>-----------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>MLE</td>
<td>Supervised</td>
<td>Batch</td>
<td>Closed-form</td>
</tr>
<tr>
<td>Laplace Smoothing</td>
<td>Supervised</td>
<td>Batch</td>
<td>Closed-form</td>
</tr>
<tr>
<td>Minimize Squared Error (for linear regression)</td>
<td>Supervised</td>
<td>Batch</td>
<td>Closed-form</td>
</tr>
<tr>
<td>Gradient Descent</td>
<td>Supervised</td>
<td>Batch</td>
<td><strong>Iterative</strong></td>
</tr>
<tr>
<td>Perceptron</td>
<td>Supervised</td>
<td>Online</td>
<td><strong>Iterative</strong></td>
</tr>
<tr>
<td>k-NN training (memorization)</td>
<td>Supervised</td>
<td>Online</td>
<td>Closed-form</td>
</tr>
</tbody>
</table>
## Complexity Control

<table>
<thead>
<tr>
<th>Model</th>
<th>Name an appropriate method to prevent overfitting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bayes Net/Naïve Bayes</td>
<td>Laplace smoothing</td>
</tr>
<tr>
<td>Linear Regression</td>
<td>L1 or L2 regularization + gradient descent</td>
</tr>
<tr>
<td>Linear Classification</td>
<td>Maximum margin learning</td>
</tr>
<tr>
<td>k-NN</td>
<td>Choose higher values of k</td>
</tr>
</tbody>
</table>