CSE 6363 - *Machine Learning*

Project 1- Spring 2022

Due Date: Mar. 21 2022, 3:30 pm

Data Set Generation

This assignment consists a number of implementation and result analysis questions. Some of them (including 1 b), 1 c), 2 and 3) require data. To obtain the data for these problems you need to do the following:

- Go to https://ranger.uta.edu/~huber/cse6363/Proj1/generate_data_proj1.php
- Enter your student ID number (the 1000... number on your student ID) and hit submit
- Save the generated web page and submit it with your assignment
- Copy the generated data to files and parameters on your computer and use them with the corresponding questions

Make sure that you enter your own student ID. Results on data for other student ID numbers will not be considered correct solutions.

Linear Regression

1. Consider a simplified fitting problem in the frequency domain where we are looking to find the best fit of data with a set of periodic (trigonometric) basis functions of the form $1,\ x,\ sin(x),\ cos(x),\ sin(k*x),\ cos(k*x),\ sin(2*k*x),\ cos(2*k*x),\ ...$, where k is effectively the frequency increment. The resulting function for a given ”frequency increment”, k, and ”function depth”, d, and parameter vector Θ is then:

$$\boxed{y = \Theta_0 * 1 + \Theta_1 * x + \sum_{i=1}^{d} (\Theta_{2*i} * \sin(i*k*x) + \Theta_{2*i+1} * \cos(i*k*x))}$$

For example, if $k = 1$ and $d = 2$, your basis (feature) functions are $1,\ x,\ sin(x),\ cos(x),\ sin(2x),\ cos(2x)$, and we are looking for the best matching parameters Θ for the function $\Theta_0 + \Theta_1 * x + \Theta_2 * sin(x) + \Theta_3 * cos(x) + \Theta_4 * sin(2x) + \Theta_5 * cos(2x)$. This means that this problem can be solved using linear regression as the function is linear in terms of the parameters Θ.

You obtain your value for the ”frequency increment” k and thus your basis functions as part of the data generation process described above.

a) Implement a linear regression learner to solve this best fit problem for 1 dimensional data. Make sure your implementation can handle fits for different ”function depths” (at least to ”depth” 6).

How will the input d be provided? User input?

b) Apply your regression learner to the data set that was generated for Question 1b) and plot the resulting function for ”function depth” 0, 1, 2, 3, 4, 5, and 6. Plot the resulting function together with the data points (using your favorite plotting program, e.g. Matlab, Octave, ...)

c) Evaluate your regression functions by computing the error on the test data points that were generated for Question 1c). Compare the error results and try to determine for what "function depths" overfitting might be a problem. Which "function depth" would you consider the best prediction function and why.

d) Repeat the experiment and evaluation of part b) and c) using only the first 20 elements of the training data set part b) and the Test set of part c). What differences do you see and why might they occur ? Why?

Locally Weighted Linear Regression

2. Another way to address nonlinear functions with a lower likelihood of overfitting is the use of locally weighted linear regression where the neighborhood function addresses non-linearity and the feature vector stays simple. In this case we assume that we will use only the raw feature, x, as well as the bias (i.e. a constant feature 1). Thus the locally applied regression function is $\boxed{y = \Theta_0 + \Theta_1 * x.}$

   As discussed in class, locally weighted linear regression solves a linear regression problem for each query point, deriving a local approximation for the shape of the function at that point (as well as for its value). To achieve this, it uses a modified error function that applies a weight to each data point's error that is related to its distance from the query point. Here we will assume that the weight function for the i^{th} data point and query point x is:

   $$\boxed{w^{(i)}(x) = e^{-\frac{(x^{(i)}-x)^2}{2\gamma^2}}}$$

   where γ is a measure of the "locality" of the weight function, indicating how fast the influence of a data point changes with its distance from the query point.

   Your value for γ is provided during data generation.

   a) Implement a locally weighted linear regression learner to solve the best fit problem for 1 dimensional data.

   b) Apply your locally weighted linear regression learner to the data set that was generated for Question 1b) and plot the resulting function together with the data points (using your favorite plotting program, e.g. Matlab, Octave, ...)

   c) Evaluate the locally weighted linear regression on the Test data from Question 1 c). How does the performance compare to the one for the results from Question 1 c) ?

   d) Repeat the experiment and evaluation of part b) and c) using only the first 20 elements of the training data set. How does the performance compare to the one for the results from Question 1 d) ? Why might this be the case ?

   e) Given the results form parts c) and d), do you believe the data set you used was actually derived from a function that is consistent with the function format in Question 1 ? Justify your answer.

Logistic Regression

3. Consider again the problem from Questions 2 and 3 in the first assignment where we want to predict the gender of a person from a set of input parameters, namely height, weight, and age. Assume the same datasets you generated for the first assignment.

a) Implement logistic regression to classify this data (use the individual data elements, i.e. height, weight, and age, as features). Your implementation should take different data sets as input for learning.

b) Plot the resulting separating surface together with the data points (using your favorite plotting program, e.g. Matlab, Octave, ...). To do this plotting you need to project the data and function into into one or more 2D space. The best visual results will be if projection is done along the separating hyperplane (i.e. into a space described by the normal of the hyperplane and one of the dimension within the hyperplane).

c) Evaluate the performance of your logistic regression classifier in the same way as for Homework 1 using leve-one-out validation and compare the results with the ones for KNN and Naïve Bayes (either from your first assignment or, if you did not implement these, using an existing implementation). Discuss what differences exist and why one method might outperform the others for this problem.

Left

d) Repeat the evaluation and comparison from part c) with the age feature removed. Again, discuss what differences exist and why one method might outperform the others in this case.

1. my data
2. readme
3. code