Quiz 1

Started: Oct 25 at 8:34pm

Quiz Instructions

Please

Question 1

5 pts

Consider a regression problem in which the underlying data generating process is used to produce N i.i.d. examples, in which $\mathbf{x}$ is drawn uniformly from some domain, and y is drawn conditionally according to

$$y = F(\mathbf{x}) + \nu, \qquad \nu \sim \mathcal{N}(0, \sigma^2),$$

i.e., ν is additive zero-mean Gaussian noise with variance σ^2.

Unregularized least squares (no regularization!) is used to train a linear regression model on N examples sampled i.i.d. from this process.

Suppose now you have the power to change the data generating process in a variety of ways, to make the learner's life easier. Specifically, which of the following changes may lead the trained model to have lower **test** error?

Select all options (zero or more) that may help.

- [x] Increase N (number of training examples)
- [ ] Decrease N (number of training examples)
- [x] Make F linear, if it is not linear at the moment
- [x] Decrease the noise variance
- [ ] Increase the noise variance

Question 2

3 pts

If you chose at least one option above: for each option you chose, briefly explain why it would help.

If you chose none, explain why each of the options would not help.

- Increasing the N will let the model learn the underlying distribution better. If there is one data point, the model cannot know the distribution. But if 100 points are drawn, the model can infer the distribution better. The model will make more accurate predictions as the test data is also drawn from the distribution.
- F(x) is the "real distribution" that we use a linear model to fit. If F(x) is quadratic, our linear model will do poorly. But if F(x) is linear, our model can achieve a decent fit.
- Total error ^ 2 = noise + bias ^ 2 + variance ^ 2. Therefore, the lower the noise variance, the smaller the total error.

ul ▸ li 117 words </>

Question 3

6 pts

Suppose we fit an unregularized least squares linear regression model $\hat{y} = f(x)$ on a training set $\{(x_i, y_i)\}_{i=1}^{N}$ of (scalar) inputs x_i and the corresponding target values y_i. We can then compute the residuals (prediction errors)

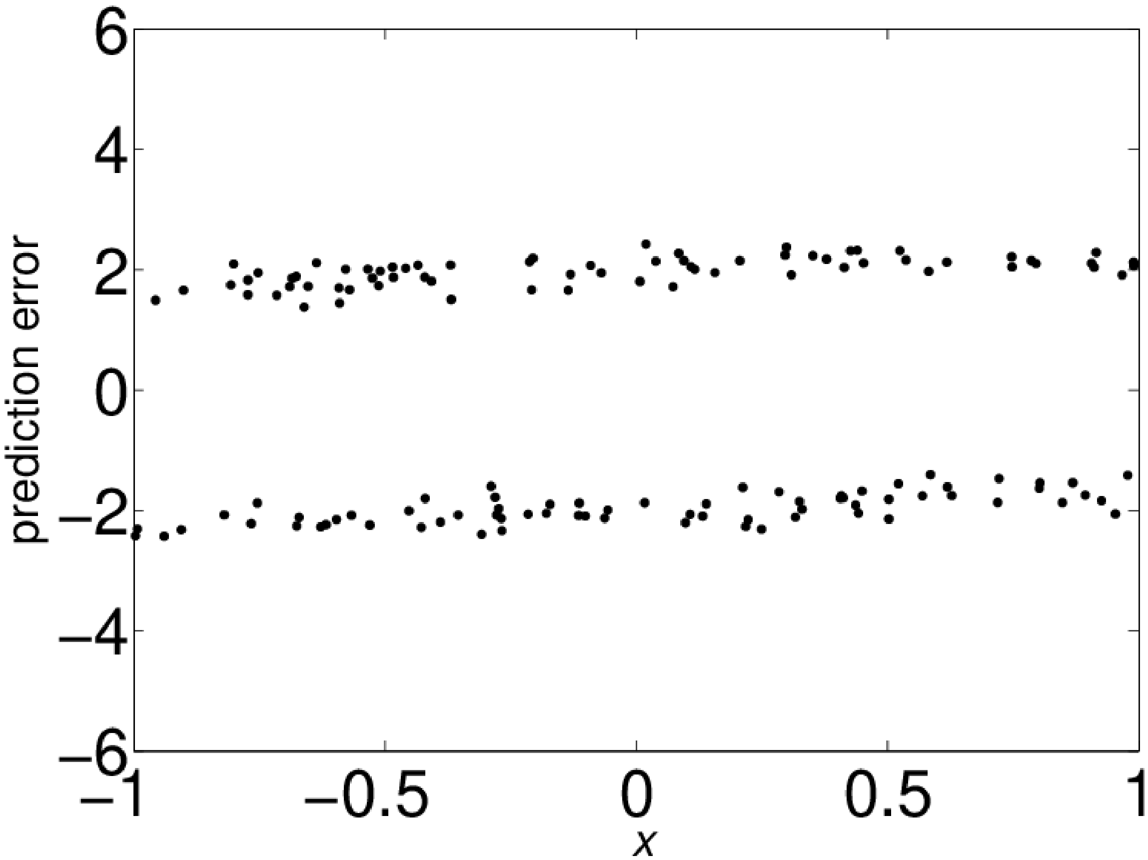
$$e_i = f(x_i) - y_i$$

for every training example, $i = 1, \ldots, N$ and plot the values of e_i vs. x_i. (Note that e_i is a signed quantity -- it can be positive or negative).

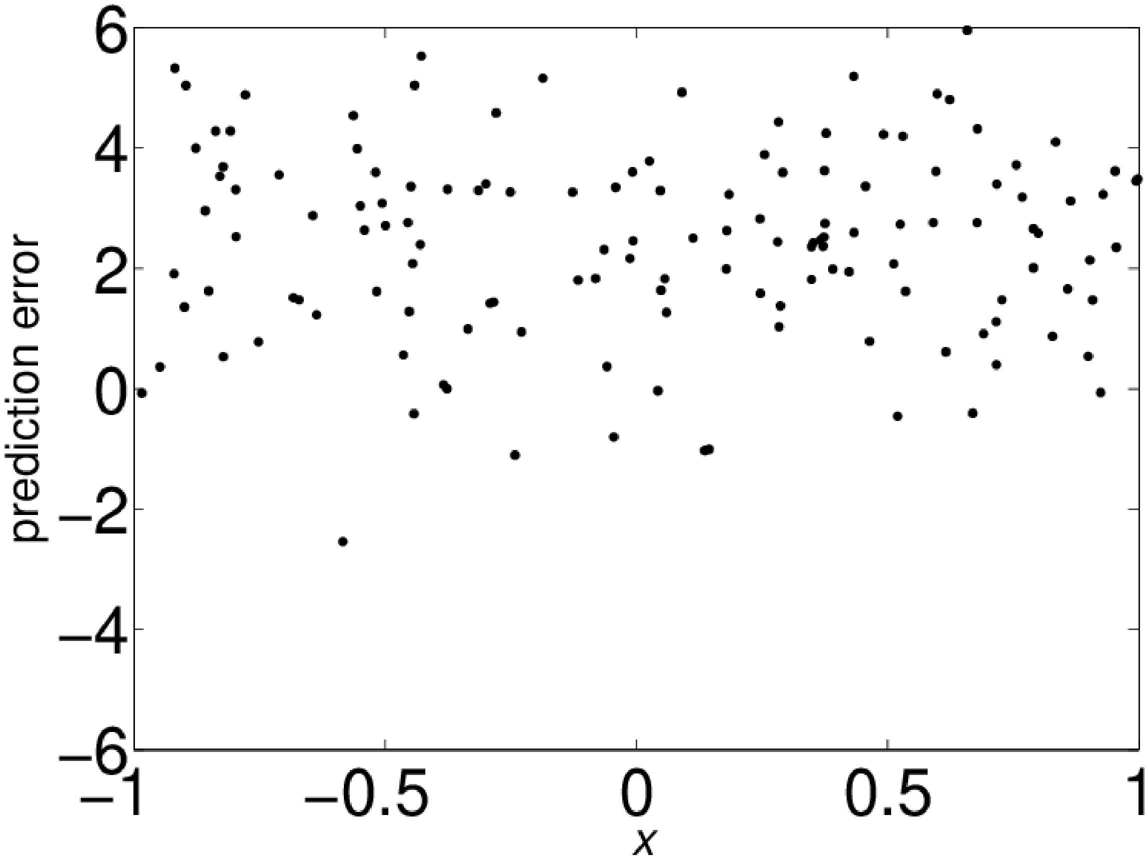
Consider the three plots below. Each plot may, or may not be showing the distribution of of e_i vs. x_i. on a training set of a linear least squares regression model $f(x)$.

For each plot, either explain why it can not be showing residuals of least squares regression on the training set, or describe qualitatively (in words) the form of a data set that would produce such residuals (that is, describe the properties of the dependence of y on x that the residual plot at hand implies).

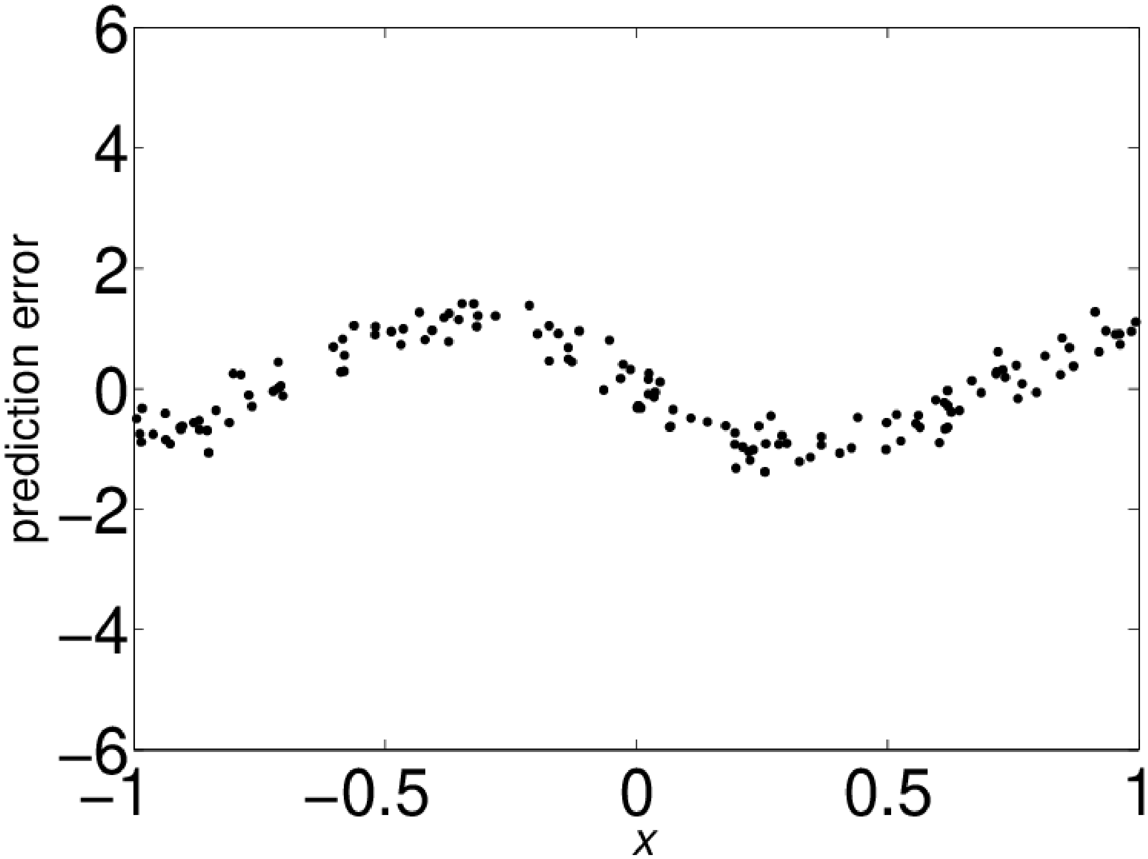
Plot A:



Plot B:



Plot C:



Advice: Pay attention to both the shape and position of point clouds and the axis labels/values. Don't worry about potential tiny numerical effects not noticeable by eye -- this is not a trick question.

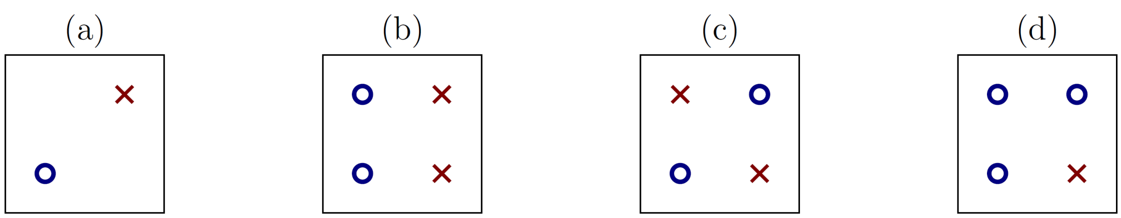
- Picture 1 is possible. Consider a training set where y's lies evenly on y = sqrt(2) stripe and y = - sqrt(2) stripe. Then, the best regression line would be y = 0. And the residues would be positive or negative 2.
- Picture 2 is not possible. The least-square regression will minimize the sum of squared distances. If some residuals are positive, some must be negative.
- Picture 3 is possible. A training set with a similar "sine shape" would produce such a residue plot -- some y_i's are above the prediction line and some below.

ul ▸ li 95 words </>

Question 4

2 pts

Each of the following four plots depicts a classification data set in 2D, with two classes denoted by circles and crosses. Which plot(s) show a data set that is not linearly separable (i.e., can not be classified with a linear classifier $\hat{y}(\mathbf{x}) = \text{sign}\left(\mathbf{w} \cdot \mathbf{x} + b\right)$?



- [ ] (a)
- [ ] (b)
- [x] (c)
- [ ] (d)

Question 5

2 pts

Suppose we trained a binary linear classifier on a training set, with class labels $y_i \in \{\pm 1\}$, and obtained a linear classifier $\hat{y}(\mathbf{x}) = \text{sign}\left(\mathbf{w} \cdot \mathbf{x} + b\right)$

(Assume, as usual, that if the score function inside the sign is zero, will produce a randomly picked label).

Now we are told that there was a bug in the labeling procedure, and the classes were switched; that is, \emph{all} examples labeled $y_i = +1$ should have been labeled $y_i = -1$ and vice versa. How will you amend the classifier above in light of this change? (choose a single answer)

(A) No need to adapt; the classifier doesn't care that the training labels have been reserved, and will still produce good predictions.

(B)The new classifier is obtained by changing the sign of both $\mathbf{w}$ and b,

$$\hat{y}(\mathbf{x}) = \text{sign}\left(-\mathbf{w} \cdot \mathbf{x} - b\right)$$

(C) The new classifier is obtained by changing the sign of b (no need to change $\mathbf{w}$),

$$\hat{y}(\mathbf{x}) = \text{sign}\left(\mathbf{w} \cdot \mathbf{x} - b\right)$$

(D) No simple modification exists; we need to go back and re-train the classifier on the re-labeled data.

- ( ) (A) no need to fix anything, the classifier is fine as is
- (•) (B) need to flip the sign of both w and b
- ( ) (C) need to only flip the sign of b
- ( ) (D) no simple change, need to retrain on relabeled data

Quiz saved at 9:46pm

Submit Quiz