There are 11 questions on the following 6 pages. Do as many of them as you can in the available time. I will score each question out of 10 points AND TOTAL YOUR BEST 7 SCORES. (That is, this is a 70 point exam.)
1. This is a "no-intercept" one-variable linear regression linear model problem. That is, suppose

\[ y_i = \beta x_i + \epsilon_i \]

for \( i = 1, 2, \ldots, N \) where the \( \epsilon_i \) are iid \( N(0, \sigma^2) \). \( N = 5 \) training cases are in the following table.

<table>
<thead>
<tr>
<th>( x )</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>( y )</td>
<td>6</td>
<td>8.5</td>
<td>10</td>
<td>11.5</td>
<td>13</td>
</tr>
</tbody>
</table>

10 pts a) Find maximum likelihood estimates of the two (real-valued) parameters \( \beta \) and \( \sigma^2 \).

\[
\hat{\beta}_{MLE} = (X'X)^{-1}X'Y = \frac{1}{135} \begin{pmatrix} 18 & 39 & 50 & 69 & 91 \end{pmatrix} = 1.9407
\]

\[
\hat{\sigma}^2_{MLE} = \frac{1}{N} \sum (y_i - \hat{y})^2 = \frac{1}{5} \left[ (6 - 1.9407(3))^2 + (8.5 - 1.9407(4))^2 + (10 - 1.9407(5))^2 + (11.5 - 1.9407(6))^2 + (13 - 1.9407(7))^2 \right] \approx 0.2052
\]

b) Give 95% 2-sided prediction limits for an additional/future \( y \) based on a new value of the predictor, \( x = 5.5 \). (If you don't have an appropriate table of distribution percentage points with you, indicate exactly what % point of exactly what distribution (including d.f., etc.) you would put where in your calculation.)

\[
\text{Upper Limit} = 5.5(\hat{\beta}) + t_{\alpha/2, N-P} \sqrt{\hat{\sigma}^2_{MLE} \left( 1 + \frac{1}{N} \right)} = 10.67 + 2.565 \sqrt{\frac{0.2052}{5-1}} \approx 10.67 + 4.92
\]

\[
\text{Lower Limit} = 5.5(\hat{\beta}) - t_{\alpha/2, N-P} \sqrt{\hat{\sigma}^2_{MLE} \left( 1 + \frac{1}{N} \right)} = 5.5(1.9407) - 2.565 \sqrt{\frac{0.2052}{5-1}} \approx 5.5 - 4.92
\]

i.e. \( 10.67 \pm 4.92 \)
2. One needs to do classification based on two discrete variables \( x_1 \) and \( x_2 \). A large training set (with \( N = 100,000 \)) is summarized below in terms of counts of training cases with \((y_i, x_{1i}, x_{2i})\) of various types. (We implicitly assume these are based on an iid sample from the joint distribution of \((y, x_1, x_2)\).)

\[
\begin{array}{ccc}
\text{\( y = 0 \) cases} & x_2 = 1 & x_2 = 2 & x_2 = 3 \\
\hline
x_1 = 2 & 9,000 & 3,000 & 10,000 \\
x_1 = 1 & 6,000 & 4,000 & 8,000 \\
\end{array}
\]

\[
\begin{array}{ccc}
\text{\( y = 1 \) cases} & x_2 = 1 & x_2 = 2 & x_2 = 3 \\
\hline
x_1 = 2 & 4,000 & 12,000 & 15,000 \\
x_1 = 1 & 13,000 & 11,000 & 5,000 \\
\end{array}
\]

10 pts a) Of interest is an approximately optimal classifier based on the 100,000 training cases. Fill in the table below with either a 0 or 1 in each cell to indicate whether such a classifier classifies to \( y = 0 \) or to \( y = 1 \) for that \((x_1, x_2)\) pair.

Approximate \( \hat{y}^{\text{opt}}(x) \):

\[
\begin{array}{ccc}
x_2 = 1 & x_2 = 2 & x_2 = 3 \\
\hline
x_1 = 2 & 0 & 1 & 1 \\
x_1 = 1 & 1 & 0 & 0 \\
\end{array}
\]

10 pts b) Compute a (0-1 loss) training error rate, \( \overline{\text{err}} \), for your choice of classifier in a).

This is just the number misclassified by the rule in a) divided by \( N = 100,000 \). This is

\[
\overline{\text{err}} = \frac{1}{100,000} \left[ 4,000 + 3,000 + 10,000 + 6,000 + 4,000 + 5,000 \right] = .32
\]
3. Here is a toy problem. Consider a 1-nn classification scheme based on a 1-dimensional input \( x \). Suppose that a size \( N = 5 \) training set of pairs \((y, x)\) is as below.

\[
T = \{(0,5),(1,6),(1,7),(0,7),(1,8)\}
\]

Then suppose that 3 bootstrap samples are

\[
T^*1 = \{(0,5),(1,6),(0,7),(1,8)\}
\]
\[
T^*2 = \{(0,5),(0,5),(1,6),(1,7),(0,7)\}
\]
\[
T^*3 = \{(1,7),(1,7),(0,7),(1,8),(1,8)\}
\]

10 pts a) Find the values of the 1-nn classifiers \( \hat{y}^*_{b}(x) \) based on each of the bootstrap samples for the training \( x \)'s and record them below. (Break "ties" any way you wish.) Then give values for the bootstrap classifier \( \hat{y}^{\text{boot}}(x) \).

<table>
<thead>
<tr>
<th>( x )</th>
<th>( x = 5 )</th>
<th>( x = 6 )</th>
<th>( x = 7 )</th>
<th>( x = 8 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \hat{y}^*_{1}(x) )</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>( \hat{y}^*_{2}(x) )</td>
<td>0</td>
<td>1</td>
<td>1 or 0</td>
<td>1 or 0</td>
</tr>
<tr>
<td>( \hat{y}^*_{3}(x) )</td>
<td>1 or 0</td>
<td>1 or 0</td>
<td>1 or 0</td>
<td>1</td>
</tr>
<tr>
<td>( \hat{y}^{\text{boot}}(x) )</td>
<td>0</td>
<td>1</td>
<td>0 (or 1)</td>
<td>1</td>
</tr>
</tbody>
</table>

10 pts b) Find the OOB (out-of-bag) (0-1 loss) error rate for 1-nn classification based on this small number of bootstrap samples.

\( \hat{y}^*_{1} \) is built on all but (1,7) and makes 1 error on it

\( \hat{y}^*_{2} \) is built on all but (1,8) and makes, say, \( \frac{1}{2} \) an error on that case

\( \hat{y}^*_{3} \) is built on all but (0,5) and (1,6) and makes, say, \( 2 \times \frac{1}{2} = 1 \) error on them

Then an OOB error rate is

\[
\frac{1}{4} \left[ 1 + \frac{1}{2} + \frac{1}{2} + \frac{1}{2} \right] = \frac{2.5}{4} = .625
\]

numbers of errors made on missing cases

\( \text{total of numbers of cases missing in the job} \)
4. For the same toy scenario as in the previous problem, (1-nn classification based on the training set $T = \{(0,5),(1,6),(1,7),(0,7),(1,8)\}$) find the 5-fold (0-1 loss) cross-validation error rate, $CV$.

Each individual data pair makes a “fold” and we thus need only find whether it’s classified correctly by assigning to its nearest neighbor. So

$$CV = \frac{1}{k} = \frac{1}{5} \left( \frac{1}{1} + \frac{0or1}{1} + \frac{1}{1} + \frac{1}{1} + \frac{0or1}{1} \right)$$

$$= \frac{1}{(015)} \phi \phi \phi \phi \phi \phi$$

which is, say, $\frac{4}{5}$

$$= \frac{13}{15}$$

5. Below is another training set involving a 1-d predictor $x$. Argue carefully that there is no support vector classifier based only on $x$ that has (0-1 loss) training error rate $err = 0$. Then identify (by making a graph, not doing calculus) a maximum margin support vector classifier (with $err = 0$) based on $x_1 = x$ and $x_2 = x^2$.

<table>
<thead>
<tr>
<th>$x$</th>
<th>$-4$</th>
<th>$-3$</th>
<th>$-2$</th>
<th>$-1$</th>
<th>$0$</th>
<th>$1$</th>
<th>$2$</th>
<th>$3$</th>
<th>$4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$y$</td>
<td>$1$</td>
<td>$1$</td>
<td>$0$</td>
<td>$0$</td>
<td>$0$</td>
<td>$0$</td>
<td>$1$</td>
<td>$1$</td>
<td>$1$</td>
</tr>
</tbody>
</table>

Rationale for no perfect SV classifier based on $x$ alone:

A SV classifier based on $x$ alone would split $x$ at some value, say $c$, classifying 0 to one side and 1 to the other. For no $c$ can we get all 1’s on one side and all 0’s on the other.

Maximum margin classifier based on $(x_1, x_2) = (x, x^2)$: $\hat{y}(x_1, x_2) = I[\beta_0 + \beta_1 x_1 + \beta_2 x_2 > 0]$

$$\beta_0 = -6.5 \quad \beta_1 = 0 \quad \beta_2 = 1$$
6. In a logistic regression problem with two standardized predictors \( x_1 \) and \( x_2 \), the model

\[
\ln \left( \frac{P[y = 0 \mid x]}{1 - P[y = 0 \mid x]} \right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2
\]

is fit twice, once by maximum likelihood and once by Lasso-penalized maximum likelihood. The two sets of fitted coefficients produced were

\[
\hat{\beta}_0 = .44, \hat{\beta}_1 = .98, \hat{\beta}_2 = .66 \quad \text{(run #1)}
\]

\[
\hat{\beta}_0 = .52, \hat{\beta}_1 = 1.14, \hat{\beta}_2 = .82 \quad \text{(run #2)}
\]

Which of the two sets of coefficients do you think is the set of Lasso coefficients and why?

The first set has smaller \( |\beta_1| \) and \( |\beta_2| \) and thus has a "less complex" fitted form for \( \ln \left( \frac{P}{1-P} \right) \). It is the Lasso version.

The fits were made using a data set with about the same numbers of \( y = 0 \) and \( y = 1 \) data points.

Based on run #1, what classifier would be appropriate for a case where it's known that \( P[y = 0] = .1 \)?

We go from a case-control data set to the rightmost logistic regression form via \( \beta = \hat{\beta}^a \leq \hat{\beta} \), and \( \hat{\beta}_2 \) don't change. We change \( \hat{\beta}_0 \) via

\[
\hat{\beta}_0 = \hat{\beta}_0^a - \ln \left( \frac{N_0}{N_1} \right) + \ln \left( \frac{g(0)}{1-g(0)} \right) = .94 - .20 = -2.20
\]

Then \( \hat{y}(x) = 1 \left[ \hat{y}_0^a \hat{y}_1 + .98x_1 + .66x_2 < 0 \right] \)

7. Suppose that 5 different classifiers are fit to a data set. If one wishes to make an ensemble classifier using weights \( w_1 = .3, w_2 = .1, w_3 = .2, w_4 = .25, \) and \( w_5 = .15 \) what "ensemble" classification is produced by each of the following sets of individual classifier results? (Write 0 or 1 in the empty cell for each set of \( \hat{y} \)’s. Write "0/1" for any "ties.")

<table>
<thead>
<tr>
<th>( \hat{y}_1 )</th>
<th>( \hat{y}_2 )</th>
<th>( \hat{y}_3 )</th>
<th>( .25 \hat{y}_4 )</th>
<th>( .15 \hat{y}_5 )</th>
<th>( \hat{y}_{\text{Ensemble}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>( 0 )</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>( 1 )</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>( 0/1 )</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>( 0 )</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>( 0 )</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>( 1 )</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>( 0 )</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>( 1 )</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>( 0 )</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>( 0/1 )</td>
</tr>
</tbody>
</table>
8. Below is a $p = 2$ classification training set for 2 classes.

Using empirical misclassification rate as your splitting criterion and forward selection, find a reasonably simple binary tree classifier that has empirical error rate 0. Carefully describe it below, using as many nodes as you need. Then draw in the final set of rectangles corresponding to your binary tree on the graph above.

At the root node: split on $x_1 / x_2$ (circle the correct one of these) at the value $\phantom{14}$
Classify to Class 0 if $\phantom{<}$ (creating Node #1)
Classify to Class 1 otherwise (creating Node #2)

At Node # $\phantom{1}$: split on $x_1 / x_2$ at the value $\phantom{11}$
Classify to Class 0 if $\phantom{<}$ (creating Node #3)
Classify to Class 1 otherwise (creating Node #4)

At Node # $\phantom{4}$: split on $x_1 / x_2$ at the value $\phantom{5}$
Classify to Class 0 if $\phantom{<}$ (creating Node #5)
Classify to Class 1 otherwise (creating Node #6)

At Node # $\phantom{7}$: split on $x_1 / x_2$ at the value $\phantom{8}$
Classify to Class 0 if $\phantom{<}$ (creating Node #7)
Classify to Class 1 otherwise (creating Node #8)

At Node # $\phantom{9}$: split on $x_1 / x_2$ at the value $\phantom{10}$
Classify to Class 0 if $\phantom{<}$ (creating Node #7)
Classify to Class 1 otherwise (creating Node #8)

At Node # $\phantom{10}$: split on $x_1 / x_2$ at the value $\phantom{11}$
Classify to Class 0 if $\phantom{<}$ (creating Node #9)
Classify to Class 1 otherwise (creating Node #10)