633 HW3

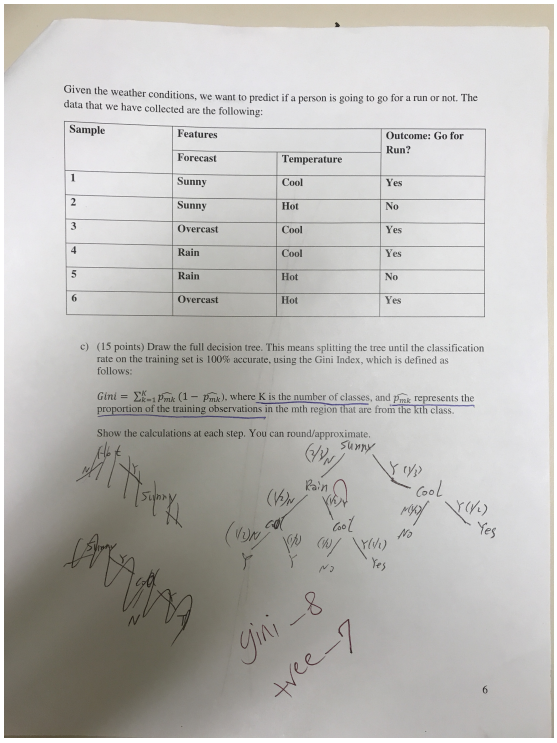
UIN: 128007734

November 2019

1 Question 1: Midterm Correction

Choose the question part (e.g. 1a or 2c) on the midterm you performed most poorly on. Please type up a solution guide that explains the solution and steps needed to arrive at this solution. Please show your work (or if it is a conceptual question, details on how you analyze the concept and evaluate the importance). Then include a section in which you detail your mistakes and explain your new understanding of the problem. Finally, attach an image of the question you are correcting to show the points taken off and the adjustments made. This question will count for both homework credit and give you the possibility of gaining up to 10 points back on the midterm question. If you do not have a question that you lost 10 points on, you will receive full credit for the question and the remaining points will be considered extra credit on top of your overall exam score.

The original problem on mid-term exam:



Given the weather conditions, we want to predict if a person is going to go for a run or not. The data that we have collected are the following:

Sample	Features		Outcome: Go for Run?
	Forecast	Temperature	
1	Sunny	Cool	Yes
2	Sunny	Hot	No
3	Overcast	Cool	Yes
4	Rain	Cool	Yes
5	Rain	Hot	No
6	Overcast	Hot	Yes

c) (15 points) Draw the full decision tree. This means splitting the tree until the classification rate on the training set is 100% accurate, using the Gini Index, which is defined as follows:

$Gini = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk})$, where K is the number of classes, and $\hat{p}_{mk}$ represents the proportion of the training observations in the mth region that are from the kth class.

Show the calculations at each step. You can round/approximate.

6

Answer:

Here we need to think about impurity. 0 means minimum impurity. 1/2 means highest impurity.

1) Over all speaking, we have four "yes" and two "no" (outcomes). The impurity is 1 - $(4/6)^2 - (2/6)^2 = 4/9$

2) Now let us check with "Forest", which has three weathers:

i: Sunny: 1 for "Yes", and 1 for "No"; Occupy (1/3) of the input data; the impurity is 1 - $(1/2)^2 - (1/2)^2 = 1/2$

ii: Rain: 1 for "Yes", and 1 for "No"; Occupy (1/3) of the input data; the impurity is 1 - $(1/2)^2 - (1/2)^2 = 1/2$

iii: Overcast: 2 for "Yes", and 0 for "No"; Occupy (1/3) of the input data; the impurity is 1 - $(2/2)^2 - (0/2)^2 = 0$

the overall impurity here is: (1/3)*(1/2)*2 = 1/3 = 3/9

3) Now let us check with "Temperature", which has two kinds: i: Cool: 3 for "Yes", and 0 for "No"; Occupy (1/2) of the input data; the impurity is 1 - $(3/3)^2 - (0/3)^2 = 0$

ii: Hot: 1 for "Yes", and 2 for "No"; Occupy (1/2) of the input data; the impurity is 1 - $(1/3)^2 - (2/3)^2 = 4/9$

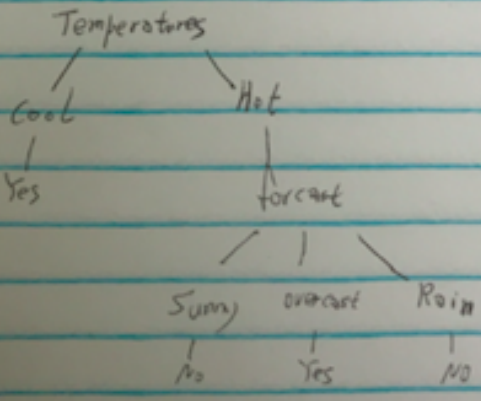
the overall impurity here is: (4/9)*(1/2) = 2/9

Smaller impurity means better for classification. Thus, here we plan to use "Temperature" as the 1st classifier, and use "Forcast" as the 2nd one

To calculate the impurity:

	forcast			Temperatures	
	Sunny	Rain	Overcast	Cool	Hot
Yes	1	1	2	Yes: 3	1
No	1	1	0	No: 0	2
impurity:	$\frac{1}{2}$	$\frac{1}{2}$	0	impurity: 0	$\frac{2}{3}$
No. of class	2	2	2	No. of class 3	3
Rate	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	Rate $\frac{1}{2}$	$\frac{1}{2}$
overall impurity:	$\frac{1}{3}\times\frac{1}{2}+\frac{1}{3}\times\frac{1}{2}+\frac{1}{3}\times 0=\frac{3}{9}$			overall impurity:	$0+\frac{1}{2}\times\frac{2}{3}=\frac{2}{6}$

Thus, we use temperatures to be the 1st classifier: ← smaller impurity, better classifier



2 Question 2: Gradient Descent Boosting: Hastie et al. algorithm 10.4

(a) Please implement algorithm 10.4 for multi-class gradient boosting. You must manually im plement this. If you use other sources for coding help you must cite them. (Please note - if you look up implementations for this and do not appropriately cite them - this is considered cheating). (b) Comparison of your implementation with a popular package (XGBoost, GBM, or Light GBM - your choice): Using the Heart dataset - develop a multi-class classification for Heart Disease (AHD) and Thal crossed together (this requires you to create labels based upon the combination of Yes/NO and Fixed, Normal, Reversable for the final two columns). Investigate the number of iterations needed to optimize the testing error, then make the testing error start to rise for your implementation and the package you choose to compare against. For multiclass classification - please determine what metric you will be optimizing for comparison (note that AUROC only works in binary classification). (c) Please provide your feature importance, and compare to the package choice's feature impor tance. Describe similarity vs. differences and explain (in particular with relation to hyperpa rameters).

Algorithm 10.1 *AdaBoost.M1.*

1. Initialize the observation weights $w_i = 1/N,\ i = 1, 2, \ldots, N$.
2. For $m = 1$ to M:
   (a) Fit a classifier $G_m(x)$ to the training data using weights w_i.
   (b) Compute
   $$\text{err}_m = \frac{\sum_{i=1}^{N} w_i I(y_i \neq G_m(x_i))}{\sum_{i=1}^{N} w_i}.$$
   (c) Compute $\alpha_m = \log((1 - \text{err}_m)/\text{err}_m)$.
   (d) Set $w_i \leftarrow w_i \cdot \exp[\alpha_m \cdot I(y_i \neq G_m(x_i))],\ i = 1, 2, \ldots, N$.
3. Output $G(x) = \text{sign}\left[\sum_{m=1}^{M} \alpha_m G_m(x)\right]$.

Algorithm 10.4 *Gradient Boosting for K-class Classification.*

1. Initialize $f_{k0}(x) = 0,\ k = 1, 2, \ldots, K$.
2. For m=1 to M:
   (a) Set
   $$p_k(x) = \frac{e^{f_k(x)}}{\sum_{\ell=1}^{K} e^{f_\ell(x)}},\ k = 1, 2, \ldots, K.$$
   (b) For $k = 1$ to K:
      i. Compute $r_{ikm} = y_{ik} - p_k(x_i),\ i = 1, 2, \ldots, N$.
      ii. Fit a regression tree to the targets $r_{ikm},\ i = 1, 2, \ldots, N$, giving terminal regions $R_{jkm},\ j = 1, 2, \ldots, J_m$.
      iii. Compute
      $$\gamma_{jkm} = \frac{K-1}{K} \frac{\sum_{x_i \in R_{jkm}} r_{ikm}}{\sum_{x_i \in R_{jkm}} |r_{ikm}|(1 - |r_{ikm}|)},\ j = 1, 2, \ldots, J_m.$$
      iv. Update $f_{km}(x) = f_{k,m-1}(x) + \sum_{j=1}^{J_m} \gamma_{jkm} I(x \in R_{jkm})$.
3. Output $\hat{f}_k(x) = f_{kM}(x),\ k = 1, 2, \ldots, K$.

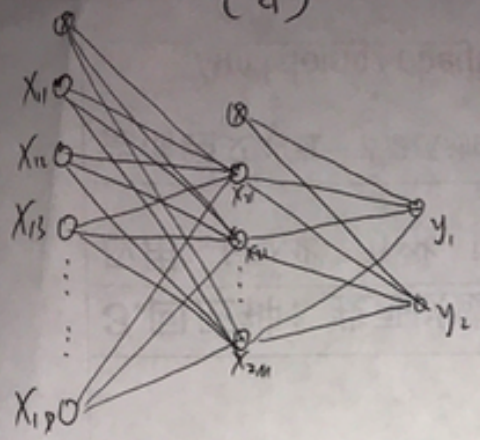
Algorithm 10.1 *AdaBoost.M1.*

1. Initialize the observation weights $w_i = 1/N,\ i = 1, 2, \ldots, N$.
2. For $m = 1$ to M:
   (a) Fit a classifier $G_m(x)$ to the training data using weights w_i.
   (b) Compute
   $$\text{err}_m = \frac{\sum_{i=1}^{N} w_i I(y_i \neq G_m(x_i))}{\sum_{i=1}^{N} w_i}.$$
   (c) Compute $\alpha_m = \log((1 - \text{err}_m)/\text{err}_m)$.
   (d) Set $w_i \leftarrow w_i \cdot \exp[\alpha_m \cdot I(y_i \neq G_m(x_i))],\ i = 1, 2, \ldots, N$.
3. Output $G(x) = \text{sign}\left[\sum_{m=1}^{M} \alpha_m G_m(x)\right]$.

3 Question 3: Activation Function in Neural Networks

(a) Provide a schematic representation of the network. Define the variables for the input, the output, and the weights. (b) Express the output as a function of the input.

(a)



$X_1 \quad W_1 \quad X_2 \quad W_2 \quad Y$

$X_1 = \{X_{11}, X_{12}, \cdots X_{1D}\}$

$X_2 = \{X_{21}, X_{22}, \cdots X_{2D}\}$

$$W_1 = \begin{bmatrix} w_{01} & w_{02} & \cdots & w_{0M} \\ w_{11} & w_{12} & \cdots & w_{1M} \\ w_{21} & w_{22} & \cdots & w_{2M} \\ w_{31} & w_{32} & \cdots & w_{3M} \\ \vdots & \vdots & & \\ w_{D1} & w_{D2} & \cdots & w_{DM} \\ 1 & 1 & \cdots & 1 \end{bmatrix} D$$

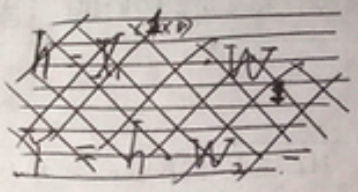
M

$$W_2 = \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \\ \vdots & \vdots \\ w_{M1} & w_{M2} \\ 1 & 1 \end{bmatrix} M$$

2

(b)

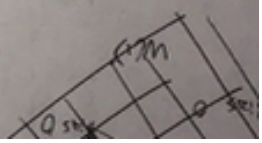
$\Rightarrow$



$h^{(1 \times M)} = \text{sigmoid}(X^{(1 \times D)} \cdot W_1^{(D \times M)})$

$Y^{(1 \times 2)} = \text{sigmoid}(h^{(1 \times M)} \cdot W_2^{(M \times 2)})$

$= S(S(X \cdot W_1) \cdot W_2)$



(c) Calculate the number of parameters that need to be inferred.
Number of Parameters = (D+1)*M + (M+1)*2

(d) Show that there exists an equivalent network with hidden unit activation functions given by the hyperbolic tangent