MATH 342W / 642 / RM 742 Spring 2024 HW #5

Laasya Indrakanti

Monday 27th May, 2024

Problem 1

These are some questions related to probability estimation modeling and asymmetric cost modeling.

(a) [easy] Why is logistic regression an example of a "generalized linear model" (glm)?

(b) [easy] What is $\mathcal{H}_{pr}$ for the probability estimation algorithm that employs the linear model in the covariates with logistic link function?

(c) [easy] If logistic regression predicts 3.1415 for a new $\boldsymbol{x}_*$, what is the probability estimate that $y = 1$ for this $\boldsymbol{x}_*$?

(d) [harder] What is $\mathcal{H}_{pr}$ for the probability estimation algorithm that employs the linear model in the covariates with cloglog link function?

(e) [difficult] Generalize linear probability estimation to the case where $\mathcal{Y} = \{C_1, C_2, C_3\}$. Use the logistic link function like in logistic regression. Write down the objective function that you would numerically maximize. This objective function is one that is argmax'd over the parameters (you define what these parameters are — that is part of the question).

Once you get the answer you can see how this easily goes to $K > 3$ response categories. The algorithm for general K is known as "multinomial logistic regression", "polytomous LR", "multiclass LR", "softmax regression", "multinomial logit" (mlogit), the "maximum entropy" (MaxEnt) classifier, and the "conditional maximum entropy model". You can inflate your resume with lots of jazz by doing this one question!

(f) [easy] Graph a canonical ROC and label the axes. In your drawing estimate AUC. Explain very clearly what is measured by the x axis and the y axis.

(g) [easy] Pick one point on your ROC curve from the previous question. Explain a situation why you would employ this model.

(h) [harder] Graph a canonical DET curve and label the axes. Explain very clearly what is measured by the x axis and the y axis. Make sure the DET curve's intersections with the axes is correct.

(i) [easy] Pick one point on your DET curve from the previous question. Explain a situation why you would employ this model.

(j) [difficult] [MA] The line of random guessing on the ROC curve is the diagonal line with slope one extending from the origin. What is the corresponding line of random guessing in the DET curve? This is not easy...

Problem 2

These are some questions related to bias-variance decomposition. Assume the two assumptions from the notes about the random variable model that produces the δ values, the error due to ignorance.

(a) [easy] Write down (do not derive) the decomposition of MSE for a given $\boldsymbol{x}_*$ where $\mathbb{D}$ is assumed fixed but the response associated with $\boldsymbol{x}_*$ is assumed random.

(b) [easy] Write down (do not derive) the decomposition of MSE for a given $\boldsymbol{x}_*$ where the responses in $\mathbb{D}$ is random but the $\boldsymbol{X}$ matrix is assumed fixed and the response associated with $\boldsymbol{x}_*$ is assumed random like previously.

(c) [easy] Write down (do not derive) the decomposition of MSE for general predictions of a phenomenon where all quantities are considered random.

(d) [difficult] Why is it in (a) there is only a "bias" but no "variance" term? Why did the additional source of randomness in (b) spawn the variance term, a new source of error?

(e) [harder] A high bias / low variance algorithm is underfit or overfit?

(f) [harder] A low bias / high variance algorithm is underfit or overfit?

(g) [harder] Explain why bagging reduces MSE for "free" regardless of the algorithm employed.

(h) [harder] Explain why RF reduces MSE atop bagging M trees and specifically mention the target that it attacks in the MSE decomposition formula and why it's able to reduce that target.

(i) [difficult] When can RF lose to bagging M trees? Hint: think hyperparameter choice.

Problem 3

These are some questions related to missingness.

(a) [easy] [MA] What are the three missing data mechanisms? Provide an example when each occurs (i.e., a real world situation). We didn't really cover this in class so I'm making it a MA question only. This concept will NOT be on the exam.

Missing completely at random (MCAR)- a device measuring a subject's steps per day runs out of battery for some time.

Missing at random (MAR)- data are missing on a question on mental health problems since men are less likely to disclose the information

Not missing at random (NMAR)- data are missing on a question on mental health problems since people with more extreme problems are less likely to disclose the information

(b) [easy] Why is listwise-deletion a *terrible* idea to employ in your $\mathbb{D}$ when doing supervised learning?

It makes the sample data less representative of the real population data, especially if the reason for having missing values itself is a feature. Doing this would introduce bias.

(c) [easy] Why is it good practice to augment $\mathbb{D}$ to include missingness dummies? In other words, why would this increase oos predictive accuracy?

Missingness might be a feature since the reason for having missing values might be affecting the response. It improves oos performance because it allows models to adapt to patterns of missingness and this results in a better fit.

(d) [easy] To impute missing values in $\mathbb{D}$, what is a good default strategy and why?

missForest is the best strategy because it can handle different types of data and it can handle complex interactions between variables. It is the most accurate method.

Problem 4

These are some questions related to gradient boosting. The final gradient boosted model after M iterations is denoted G_M which can be written in a number of equivalent ways (see below). The g_t's denote constituent models and the G_t's denote partial sums of the constituent models up to iteration number t. The constituent models are "steps in functional steps" which have a step size η and a direction component denoted $\tilde{g}_t$. The directional component is the base learner $\mathcal{A}$ fit to the negative gradient of the objective function L which measures how close the current predictions are to the real values of the responses:

$$\begin{aligned} G_M &= G_{M-1} + g_M \\ &= g_0 + g_1 + \ldots + g_M \end{aligned}$$

$$
\begin{aligned}
&= g_0 + \eta\tilde{g}_1 + \ldots + \eta\tilde{g}_M \\
&= g_0 + \eta\mathcal{A}\left(\langle \boldsymbol{X}, -\nabla L(\boldsymbol{y}, \hat{\boldsymbol{y}}_1)\rangle, \mathcal{H}\right) + \ldots + \eta\mathcal{A}\left(\langle \boldsymbol{X}, -\nabla L(\boldsymbol{y}, \hat{\boldsymbol{y}}_M)\rangle, \mathcal{H}\right) \\
&= g_0 + \eta\mathcal{A}\left(\langle \boldsymbol{X}, -\nabla L(\boldsymbol{y}, g_1(\boldsymbol{X}))\rangle, \mathcal{H}\right) + \ldots + \eta\mathcal{A}\left(\langle \boldsymbol{X}, -\nabla L(\boldsymbol{y}, g_M(\boldsymbol{X}))\rangle, \mathcal{H}\right)
\end{aligned}
$$

(a) [easy] From a perspective of only multivariable calculus, explain gradient descent and why it's a good idea to find the minimum inputs for an objective function L (in English).

It finds a function's minimum value by finding where the function is decreasing the fastest, which is a good idea to apply for a loss function since we want to minimize loss. It helps optimize parameters.

(b) [easy] Write the mathematical steps of gradient boosting for supervised learning below. Use L for the objective function to keep the procedure general. Use notation found in the problem header.

Gradient Boosting for Supervised Learning

fitting a response vector

$g_0 \leftarrow \vec{\hat{y}}_0$

$g_1 = g_0 + \eta\mathcal{A}(\langle X, -\vec{\nabla}L(\vec{y}, \vec{\hat{y}}_0)\rangle, \mathcal{H}) \leftarrow \vec{\hat{y}}_1 = \vec{\hat{y}}_0 - \eta\vec{\nabla}L(\vec{y}, \vec{\hat{y}}_0)$

base learner algorithm

instead of fitting a model from inputs to response, you fit a model from inputs to gradient of the objective function.

$g_0(x) + g_1(x)$

$g_2 = g_1 + \eta\mathcal{A}(\langle X, -\vec{\nabla}L(\vec{y}, \vec{\hat{y}}_1)\rangle, \mathcal{H})$

$g_0(x) + g_1(x) + \ldots + g_{m-1}(x)$

$g_M = g_{m-1} + \eta\mathcal{A}(\langle X, -\vec{\nabla}L(\vec{y}, \vec{\hat{y}}_{m-1})\rangle, \mathcal{H})$

(c) [easy] For regression, what is $g_0(\boldsymbol{x})$?

$g_0(\boldsymbol{x}) = \vec{\hat{y}}_0 = \bar{y}$

(d) [easy] For probability estimation for binary response, what is $g_0(\boldsymbol{x})$?

log odds

$g_0(\boldsymbol{x}) = ln(\frac{\hat{p}_i}{1-\hat{p}_i})$

(e) [harder] What are all the hyperparameters of gradient boosting? There are more than just two.

(f) [easy] For regression, rederive the negative gradient of the objective function L.

For Regression consider $L(y, \hat{\vec{y}}) = SSE = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$

$\Rightarrow -\vec{\nabla} L(\vec{y}, \hat{\vec{y}}) = \begin{bmatrix} 2(y_1 - \hat{y}_1) \\ 2(y_2 - \hat{y}_2) \\ \vdots \\ 2(y_n - \hat{y}_n) \end{bmatrix} = 2\vec{e}$

$y_i^2 - 2y_i\hat{y}_i + \hat{y}_i^2$
$\Rightarrow -2y_i + 2\hat{y}_i$
$= -2(y_i - \hat{y}_i)$

(g) [easy] For probability estimation for binary response, rederive the negative gradient of the objective function L.

Likelihood function becomes:

$$\prod_{i=1}^{n}\left(\frac{e^{\hat{y}_i}}{1+e^{\hat{y}_i}}\right)^{y_i}\left(1-\frac{e^{\hat{y}_i}}{1+e^{\hat{y}_i}}\right)^{1-y_i} = \prod_{i=1}^{n}\frac{e^{y_i\hat{y}_i}}{1+e^{\hat{y}_i}}$$

Since maximizing the likelihood would yield the same answer as maximizing its log.

$$\ln(\text{likelihood}) = \sum_{i=1}^{n}\ln\left(\frac{e^{y_i\hat{y}_i}}{1+e^{\hat{y}_i}}\right) = \sum y_i\hat{y}_i - \ln(1+e^{\hat{y}_i})$$

we wish to minimize objective function in gradient descent so...

$$L(\vec{y},\vec{\hat{y}}) := \sum_{i=1}^{n} -y_i\hat{y}_i + \ln(1+e^{\hat{y}_i})$$

$$-\nabla L(\vec{y},\vec{\hat{y}}) = \begin{bmatrix} y_1 - \frac{e^{\hat{y}_1}}{1+e^{\hat{y}_1}} \\ y_2 - \frac{e^{\hat{y}_2}}{1+e^{\hat{y}_2}} \\ \vdots \\ y_n - \frac{e^{\hat{y}_n}}{1+e^{\hat{y}_n}} \end{bmatrix} = \vec{y} - \frac{e^{\vec{\hat{y}}}}{1+e^{\vec{\hat{y}}}} \overset{\text{conceptually}}{\overset{\downarrow}{=}} \vec{y} - \vec{\hat{p}}$$

(h) [difficult] For probability estimation for binary response scenarios, what is the unit of the output $G_M(\boldsymbol{x}_\star)$?

(i) [easy] For the base learner algorithm $\mathcal{A}$, why is it a good idea to use shallow CART (which is the recommended default)?

(j) [difficult] For the base learner algorithm $\mathcal{A}$, why is it a bad idea to use deep CART?

(k) [difficult] For the base learner algorithm $\mathcal{A}$, why is it a bad idea to use OLS for regression (or logistic regression for probability estimation for binary response)?

(l) [difficult] If M is very, very large, what is the risk in using gradient boosting even using shallow CART as the base learner (the recommended default)?

(m) [difficult] If η is very, very large but M reasonably correctly chosen, what is the risk in using gradient boosting even using shallow CART as the base learner (the recommended default)?