Math 342W / 650.4 Spring 2022
Midterm Examination Two

Professor Adam Kapelner

May 16, 2022

Full Name ____

Code of Academic Integrity

Since the college is an academic community, its fundamental purpose is the pursuit of knowledge. Essential to the success of this educational mission is a commitment to the principles of academic integrity. Every member of the college community is responsible for upholding the highest standards of honesty at all times. Students, as members of the community, are also responsible for adhering to the principles and spirit of the following Code of Academic Integrity.

Activities that have the effect or intention of interfering with education, pursuit of knowledge, or fair evaluation of a student's performance are prohibited. Examples of such activities include but are not limited to the following definitions:

Cheating Using or attempting to use unauthorized assistance, material, or study aids in examinations or other academic work or preventing, or attempting to prevent, another from using authorized assistance, material, or study aids. Example: using an unauthorized cheat sheet in a quiz or exam, altering a graded exam and resubmitting it for a better grade, etc.

I acknowledge and agree to uphold this Code of Academic Integrity.

______________________________ ______________
signature date

Instructions

This exam is 110 minutes and closed-book. You are allowed **two** pages (front and back) of a "cheat sheet." You may use a graphing calculator of your choice. Please read the questions carefully. If the question reads "compute," this means the solution will be a number otherwise you can leave the answer in *any* widely accepted mathematical notation which could be resolved to an exact or approximate number with the use of a computer. I advise you to skip problems marked "[Extra Credit]" until you have finished the other questions on the exam, then loop back and plug in all the holes. I also advise you to use pencil. The exam score will be normed to be out of 100 points total plus extra credit if it exists. Partial credit will be granted for incomplete answers on most of the questions. $\boxed{\text{Box}}$ in your final answers. Good luck!

Problem 1 In class, we spoke about probability estimation for a binary phenomenon $\mathcal{Y} = \{0, 1\}$. We modeled each observation as an independent Bernoulli (θ_i) i.e. $Y_i \overset{ind}{\sim} \theta_i^{y_i}(1-\theta_i)^{1-y_i}$ where $\theta_i := \mathbb{E}[Y_i = 1 \mid \boldsymbol{x}_i]$ which for the Bernoulli is synonymous with $\mathbb{P}(Y_i = 1 \mid \boldsymbol{x}_i)$ and it varies with observation based on the features $\boldsymbol{x}_i$ which is a row vector of length $p+1$ since the first entry is set to be one.

To do so, we used a generalized linear model (GLM) which coerced the linear model $\boldsymbol{x} \cdot \boldsymbol{w}$ into the support of the parameter θ_i, a probability ranging from $[0, 1]$. To do this coercion, we used a link function $\phi(\boldsymbol{x} \cdot \boldsymbol{w})$ which mapped $\boldsymbol{x} \cdot \boldsymbol{w} \in \mathbb{R} \to \mathbb{S}\text{upp}[\theta_i] = [0, 1]$. Any monotonically increasing function with domain $\mathbb{R}$ and range $[0, 1]$ was legal. For example, any CDF of a random variable with support $\mathbb{R}$ fits this definition.

Let's use the link function $\phi(u)$ is the CDF of the standard normal denoted $\Phi(u)$. This algorithm is called "probit regression" and we'll denote it $\mathcal{A}_{\text{probit}}$.

- [5 pt / 5 pts] Write out the objective function to maximize which is the probability of the entire training set $\mathbb{D}$. Since this is a GLM, your answer must include the linear term for the ith observation, $\boldsymbol{x}_i \cdot \boldsymbol{w}$.

$$\mathbb{P}(Y_1, \ldots, Y_n \mid \boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = \prod_{i=1}^{n} \mathbb{P}(Y_i \mid \boldsymbol{x}_i) = \prod_{i=1}^{n} \Phi(\vec{w} \cdot \vec{x}_i)^{y_i} \left(1 - \Phi(\vec{w} \cdot \vec{x}_i)\right)^{1-y_i}$$

- [2 pt / 7 pts] Our algorithm $\mathcal{A}_{\text{probit}}$ involves running this optimization problem in the computer: $\boldsymbol{b} := \arg\max_{w} \{\text{your answer from the previous problem}\}$. What is the dimension of the vector $\boldsymbol{b}$?

$p+1$

- [3 pt / 10 pts] Given $\boldsymbol{b}$, for a new observation $\boldsymbol{x}_*$, write the explicit functional form of $g(\boldsymbol{x}_*)$, an expression that computes $\hat{p}_*$, the estimate that $\mathbb{P}(Y_* = 1 \mid \boldsymbol{x}_*)$.

$$\hat{p}_* = \Phi(\vec{b} \cdot \vec{x}_*)$$

- [6 pt / 16 pts] Assume the dataset now had $p = 1$ and $\mathcal{A}_{\text{probit}}$ returned $b_0 = 1.77$ and $b_1 = 1.10$. Interpret the value $b_1 = 1.10$. This means you must write a few sentences in English below.

When comparing two mutually observed observations (A) and (B) sampled in the same way as observations in the training set where (A) has an x_1 value one unit larger than the x_1 value of (B) then (A) is predicted to have a probit-probability that differs by +1.10 units on average from the probit-probability of (B) assuming the linear-probit model is true.

Assume $p = 1$ for the rest of the problem. Displayed below is $\mathbb{D}_{\text{test}}^{\top}$ with $n_{\text{test}} = 10$ including the probability estimates from g denoted as the vector $\hat{\boldsymbol{p}}$ underneath $\mathbb{D}_{\text{test}}^{\top}$:

$\boldsymbol{x}_{\cdot 1}$	-2.51	0.73	-3.34	6.38	1.32	-3.28	1.95	2.95	2.30	-1.22
$\boldsymbol{y}$	1	1	0	1	1	0	1	1	1	0
$\hat{\boldsymbol{p}}$	0.08	0.68	0.03	0.99	0.79	0.04	0.88	0.95	0.91	0.23
$\mathbb{1}_{\hat{p} \geq 0.5} = \hat{y}$	0	1	0	1	1	0	1	1	1	0

- [5 pt / 21 pts] Circle the letters of all the following that are **true**.

  (a) You have enough information to compute the out-of-sample Brier scoring rule

  (b) You have enough information to compute the out-of-sample log scoring rule

  (c) You have enough information to compute the out-of-sample AUC metric

  (d) The oos AUC is definiteively greater than 0.5 for this model

  (e) You have enough information to compute an approximate out-of-sample DET

- [4 pt / 25 pts] We now use this probit regression model to do binary classification. Using the naive threshold classifier, compute the average oos misclassification error.

  $$\frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{y_i \neq \hat{y}_i} = \frac{1}{10}(1) = 0.1 = 10\%$$

- [3 pt / 28 pts] If the cost of false positives was \$2 and the cost of false negatives was \$1, compute an estimate of mean cost per prediction to the nearest cent.

  $$\frac{n_{FN} c_{FN} + n_{FP} c_{FP}}{n} = \frac{(1)(\$1) + 0(\$2)}{10} = \$0.10$$

- [4 pt / 32 pts] If the cost of false negatives was much much greater than the cost of false positives, what explicit thresholding rule would minimize mean cost per prediction? Hint: there are many correct answers.

  A very low p_{th} e.g. $\mathbb{1}_{\hat{p} \geq 0.07}$

Problem 2 In class, we never spoke about count modeling i.e. $\mathcal{Y} = \{0, 1, 2, \ldots\}$ but it is very similar to our discussion of probability estimation. We will now model each observation as an independent Poisson(θ_i) i.e. $Y_i \overset{ind}{\sim} \theta_i^{y_i} e^{-\theta_i}/y_i!$ where θ_i is the $\mathbb{E}[Y_i = 1 \mid \boldsymbol{x}_i]$ and it varies with observation based on the features $\boldsymbol{x}_i$ which is a row vector of length $p+1$ since the first entry is set to be one.

To do so, we will use a generalized linear model (GLM) which coerces the linear model $\boldsymbol{x} \cdot \boldsymbol{w}$ into the support of the parameter θ_i, a mean count ranging in $(0, \infty)$. To do this coercion, we can use a link function $\phi(\boldsymbol{x} \cdot \boldsymbol{w})$ which maps $\boldsymbol{x} \cdot \boldsymbol{w} \in \mathbb{R} \to \text{Supp}[\theta_i] = (0, \infty)$. Any monotonically increasing function with domain $\mathbb{R}$ and range $(0, \infty)$ is legal.

Let's use the link function $\phi(u) = 10^u$. This algorithm is called "poisson regression" and we'll denote it $\mathcal{A}_{\text{poisson}}$.

- [6 pt / 38 pts] Write out the objective function to maximize which is the probability of the entire training set $\mathbb{D}$. Since this is a GLM, your answer must include the linear term for the ith observation, $\boldsymbol{x}_i \cdot \boldsymbol{w}$.

$$\mathbb{P}(Y_1, \ldots, Y_n \mid \boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = \prod_{i=1}^{n} \mathbb{P}(Y_i \mid \boldsymbol{x}_i) = \prod_{i=1}^{n} \frac{\left(10^{\vec{w} \cdot \vec{x}_i}\right)^{y_i} e^{-10^{\vec{w} \cdot \vec{x}_i}}}{y_i!}$$

- [1 pt / 39 pts] Our algorithm $\mathcal{A}_{\text{poisson}}$ involves running this optimization problem in the computer: $\boldsymbol{b} := \arg\max_{\boldsymbol{w}} \{\text{your answer from the previous problem}\}$. What is the dimension of the vector $\boldsymbol{b}$?

$p+1$

- [4 pt / 43 pts] For the n_{test} oos responses denoted by the vector $\boldsymbol{y}$ and oos predictions denoted by the vector $\hat{\boldsymbol{y}}$, propose a sensical error metric that gauges the oos performance of the model returned by $\mathcal{A}_{\text{poisson}}$. There are many acceptable answers.

- $SSE := (\vec{y} - \hat{\vec{y}})^T (\vec{y} - \hat{y})$
- $SAE := \sum_{i=1}^{n_{test}} |y_i - \hat{y}_i|$
- $MSE := \frac{SSE}{n-p-1}$
- $MAE = \frac{SAE}{n}$
- $RMSE := \sqrt{MSE}$
- $R^2 := 1 - SSE/SST$

- [4 pt / 47 pts] Assume the dataset now had $p = 1$ and ~~$\mathcal{A}_{\text{probit}}$~~ poisson returned $b_0 = 1.77$ and $b_1 = 1.10$. For $x_\star = 1$, compute $\hat{y}_\star$.

$$\hat{y}_\star = 10^{\vec{b} \cdot \vec{x}_\star} = 10^{\overset{b_0}{\overset{\shortparallel}{1.77}} + \overset{b_1 x_1}{\overset{\shortparallel}{1.10(1)}}} = 10^{2.87} = 741.31 \xrightarrow{\text{or}} 741$$

This page was intentionally left blank

Problem 3 In the lab we analyzed three tables: bills, bill payments, bill discounts which have 226,434 rows, 194,850 rows and 60 rows respectively. Here are the first six rows of the bills table followed by the first 6 rows of the bill payments table and the first 6 rows of the bill discounts table:

	id	due_date	invoice_date	tot_amount	customer_id	discount_id
1:	15163811	2017-02-12	2017-01-13	99490.77	14290629	7302585
2:	17244832	2016-03-22	2016-02-21	99475.73	14663516	7197225
3:	16072776	2016-08-31	2016-07-17	99477.03	14569622	7302585
4:	15446684	2017-05-29	2017-05-29	99478.60	14488427	7197225
5:	16257142	2017-06-09	2017-05-10	99678.17	14497172	7197225
6:	17244880	2017-01-24	2017-01-24	99475.04	14663516	7197225

2 unique values

	id	paid_amount	transaction_date	bill_id
1:	15272980	99165.60	2017-01-16	16571185
2:	15246935	99148.12	2017-01-03	16660000
3:	16596393	99158.06	2017-06-19	16985407
4:	16596651	99175.03	2017-06-19	17062491
5:	16687702	99148.20	2017-02-15	17184583
6:	16593510	99153.94	2017-06-11	16686215

	id	num_days	pct_off	days_until_discount
1:	5000000	20	NA	NA
2:	5693147	NA	2	NA
3:	6098612	20	NA	NA
4:	6386294	120	NA	NA
5:	6609438	NA	1	7
6:	6791759	31	1	NA

- [2 pt / 49 pts] If we were to do a left join where the left table was bill discounts and the right table was bills, what would be the maximum number of rows in the final joined table?

  60 + 226,434 − 2 = 226,492

- [2 pt / 51 pts] If we were to do a full join where the left table was bill discounts and the right table was bills, what would be the maximum number of rows in the final joined table?

  60 + 226,434 = 226,494

- [6 pt / 57 pts] Draw below a long version of the first six rows of the bill discounts table where the metric variables are the columns num_days, pct_off, days_until_discount and the id column is still the id column. Make sure the long table you display does not have any missingness. Use the listwise deletion procedure to address any missingness if it exists.

id	value	metric
5000000	20	num_days
5693147	2	pct_off
6098612	20	num_days
6386294	120	num_days
6609438	1	pct_off
6609438	7	days_until_discount
6791759	31	num_days
6791759	1	pct_off

After merging the three tables appropriately, we generated a feature paid_in_full $\in \mathcal{Y} = \{0, 1\}$ which will be our prediction target where 1 = the customer indeed paid on time. We also generate reasonable features and drop other columns that have no relevance to our prediction problem. Below is the first 6 rows of the final data frame. The first column is y followed by $p = 8$ features.

```
   paid_in_full tot_amount num_days_to_pay disc_days discount_pct_off
1:            0   99505.86               1        13                2
2:            1   99576.09              30         4               NA
3:            0   99475.42              30         2               NA
4:            0   99479.24               1        13                2
5:            0   99475.05              30        13                2
6:            0   99475.05              30         4               NA
   disc_delay num_previous_bills num_prev_bills_yes owed_per_day
1:         NA                107                  0    99505.857
2:         NA               4859                922     3319.203
3:         60               1046                  0     3315.847
4:         NA               1023                  0    99479.237
5:         NA                800                  0     3315.835
6:         NA               1595                860      3315.83
```

We then assume the missingness is this data frame is imputed using the missForest algorithm. Assume the final data frame does not have any missingness whatsover.

We then sample 2,000 observations from that final imputed data frame to fit two models of all features on `paid_in_full`:

($\mathcal{A}$ = RF) A random forest classification model with 500 trees, 4 variables tried at each split and nodesize = 400. Here are the OOB results:

	predicted 0	predicted 1	model errors
actual 0	1568	170	0.098
actual 1	60	202	0.229
use errors	0.037	0.457	0.115

Accuracy: 88.5%

($\mathcal{A}$ = CART) A classification tree model with nodesize = 1. Here are the OOB results:

	predicted 0	predicted 1	model errors
actual 0	614	19	0.030
actual 1	16	83	0.162
use errors	0.025	0.186	0.048

Accuracy: 95.219%

- [13 pt / 70 pts] Circle the letters of all the following that are **true**.

(a) The nodesize is a hyperparameter of $\mathcal{A}$ = RF

(b) The nodesize is a hyperparameter of $\mathcal{A}$ = CART

(c) The number of trees is a hyperparameter of $\mathcal{A}$ = RF

(d) The number of trees is a hyperparameter of $\mathcal{A}$ = CART

(e) The number of variables tried at each split is a hyperparameter of $\mathcal{A}$ = RF

(f) The number of variables tried at each split is a hyperparameter of $\mathcal{A}$ = CART

(g) $\mathcal{A}$ = CART cannot overfit since $p = 8$ while $n = 2,000$

(h) In this example, the CART model is estimated to do better than the RF model when predicting in the future if $c_{FP} = c_{FN}$

For the remainder of this true/false set of questions, assume the terms "MSE", "bias" and "variance" are the terms employed in the bias-variance tradeoff theorem we discussed in class. We will assume that this theorem extends to situations where $\mathcal{Y} = \{0, 1\}$ even though it was proven for $\mathcal{Y} \subseteq \mathbb{R}$.

(i) If the RF model was fit on more than 2,000 observations, it would have had less bias.

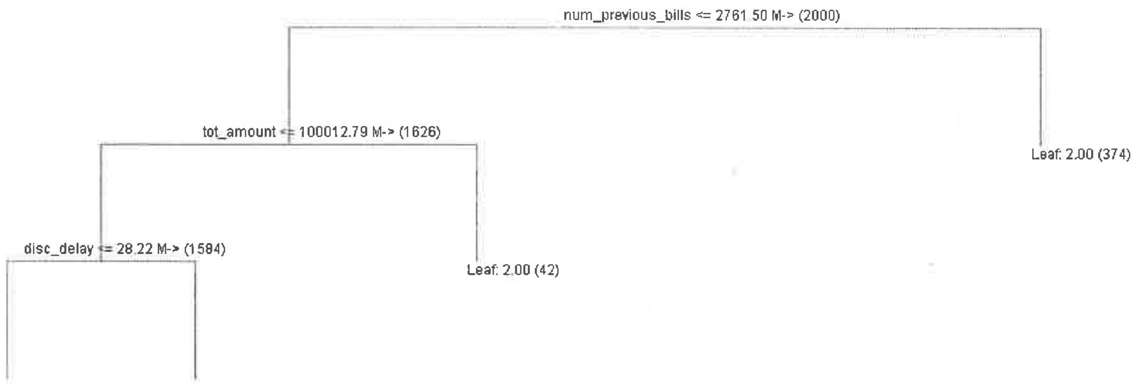
(j) If the CART model was fit on more than 2,000 observations, it would have had less bias.

(k) The RF model has less variance than the CART model

(l) If the RF model was fit with nodesize = 1, it would have had less bias than the CART model

(m) If the RF model was fit with nodesize = 1, it would have had less variance than the CART model

Below is an illustration of tree #1 in the RF model (fit with 500 trees, 4 variables tried at each split and nodesize = 400). The leaf value of "2.00" means that $\hat{y} = 1$ for that leaf. The left direction means the inequality in the split rule was true.



- [9 pt / 79 pts] Circle the letters of all the following that are **true**.

  (a) If RF model was fit with nodesize < 400, the tree would likely have more nodes and be deeper

  (b) This tree was fit seeing approximately 2/3 of the training data's observations supplied to the algorithm

  (c) This tree was fit seeing approximately $2/3 \times 2000 = 1333$ observations

  (d) This tree was fit with seeing half of the columns of the training data supplied to the algorithm

  (e) The RF model would predict this bill to be paid back if `num_previous_bills` > 2761.5

  (f) In this displayed tree, if `num_previous_bills` = 1000 and `tot_amount` = 50,000, then we are unsure what $\hat{y}$ for this tree would be.

  (g) In this displayed tree, if `tot_amount` = 150,000, then we are unsure what $\hat{y}$ for this tree would be.

  (h) The `num_previous_bills` feature is definitely the most important feature in the RF model.

  (i) The `num_previous_bills` feature is definitely the most important feature in the CART model.

Problem 4 Your training data $\mathbb{D}$ consists of a survey among births of mice in a laboratory where many features are recorded: weight, body length, hair length, gender. Mice are known to be born with equal chance of male and female. Among a sample of 50 mice, 25 were recorded male, 16 were recorded female and 9 gender values are missing.

- [2 pt / 81 pts] Regardless of any previous knowledge of biology, what would be the naive imputed values for the 9 missing gender values?

  all male

- [3 pt / 84 pts] Of the three missing data mechanisms we studied which one is *least* likely to be the mechanism that creates the missingness in the mice gender values?

  MCAR

- [4 pt / 88 pts] Consider the missingness mechanism to be one of the remaining two mechanisms. In order for missForest to be able to impute the missing mice's gender well, what would this dataset need to exhibit? Write a few sentences below.

  The mice's weight, body lenth, hair length should contain information about the mice's gender.

Problem 5 You seek to create a better model to predict the $y := \ln(\text{wind speed})$ of storms using ten continuous non-dummy linearly independent features of each storm $x_1, x_2, \ldots, x_{10}$. Consider the OLS algorithm on the following hypothesis sets consisting of linear models where the terms are described below:

$$\begin{aligned}
\mathcal{H}_0 &:= \{w_0 : w_0 \in \mathbb{R}\} \\
\mathcal{H}_1 &:= \mathcal{H}_0 \cup \{\text{all linear terms } w_j \text{ for all } x_j : w_j \in \mathbb{R} \text{ for all } j\} \\
\mathcal{H}_{2a} &:= \mathcal{H}_1 \cup \{\text{all linear terms } w_j \text{ for all } x_j \times x_k \text{ where } j \neq k : w_j \in \mathbb{R} \text{ for all } j\} \\
\mathcal{H}_{2b} &:= \mathcal{H}_{2a} \cup \left\{\text{all linear terms } w_j \text{ for all } x_j^2 : w_j \in \mathbb{R} \text{ for all } j\right\} \\
\mathcal{H}_{3a} &:= \mathcal{H}_{2a} \cup \left\{\text{all linear terms } w_j \text{ for all } x_j \times x_k \times x_\ell \text{ where } j \neq k, k \neq \ell, j \neq \ell : w_j \in \mathbb{R} \text{ for all } j\right\} \\
\mathcal{H}_{3b} &:= \mathcal{H}_{2b} \cup \mathcal{H}_{2a} \cup \left\{\text{all linear terms } w_j \text{ for all } x_j^2 \times x_k \text{ where } j \neq k \text{ and all } x_j^3 : w_j \in \mathbb{R} \text{ for all } j\right\}
\end{aligned}$$

Let g_m denote the model that is produced by OLS when $\mathcal{H}_m$ is employed e.g. $g_1 = b_0 + b_1 x_1 + \ldots + b_{10} x_{10}$ is the standard OLS model since it uses the x_j terms from $\mathcal{H}_1$ and the intercept from $\mathcal{H}_0$.

- [3 pt / 91 pts] What is the most likely reason the response was defined as the log of the measured metric wind speed?

the values of wind ~~vand~~ speed have a long right tail (skewed right)

- [3 pt / 94 pts] What is the number of terms in the mathematical model g_{2a}?

$1 + 10 + \binom{10}{2} = 11 + 45 = 56$

- [0 pt / 96 pts] If you were to employ $\mathcal{H}_{3b}$ instead of $\mathcal{H}_{3a}$, which of the three types of modeling error can potentially decrease?

unable to be determined

- [0 pt / 98 pts] If you were to employ $\mathcal{H}_{3b}$ instead of $\mathcal{H}_{3a}$, which of the three types of modeling error can potentially increase?

unable to be determined

- [4 pt / 102 pts] If you knew some of your future predictions would be extrapolations, which would you be more comfortable employing: $\mathcal{H}_{3b}$ or $\mathcal{H}_{3a}$ and why? Write a couple of sentences below.

$\mathcal{H}_{3a}$ since $\mathcal{H}_{3b}$ has 3rd-order polynomial terms. Such models can exhibit Runge's phenomenon which is

- [9 pt / 111 pts] Let SSE_m denote the SSE for g_m, let SSR_m denote the SSR for g_m, let MSE_m denote the MSE for g_m, let RMSE_m denote the RMSE for g_m, let R^2_m denote the R^2 for g_m. Circle the letters of all the following that are **true**.

  (a) $\text{RMSE}_0 < \text{RMSE}_1 < \text{RMSE}_{2a} < \text{RMSE}_{2b} < \text{RMSE}_{3a} < \text{RMSE}_{3b}$

  (b) $\text{RMSE}_0 < \text{RMSE}_1 < \text{RMSE}_{2b} < \text{RMSE}_{2a} < \text{RMSE}_{3b} < \text{RMSE}_{3a}$

  (c) $R^2_0 < R^2_1 < R^2_{2a} < R^2_{2b} < R^2_{3a} < R^2_{3b}$

  (d) $R^2_0 < R^2_1 < R^2_{2b} < R^2_{2a} < R^2_{3b} < R^2_{3a}$

  (e) $\text{SST} = \text{SSE}_0 + \text{SSE}_1 + \text{SSE}_{2b} + \text{SSE}_{2a} + \text{SSE}_{3b} + \text{SSE}_{3a} + \text{SSR}_0 + \text{SSR}_1 + \text{SSR}_{2b} + \text{SSR}_{2a} + \text{SSR}_{3b} + \text{SSR}_{3a}$

  (f) $g_0(\boldsymbol{x}_\star) < g_1(\boldsymbol{x}_\star) < g_{2b}(\boldsymbol{x}_\star) < g_{2a}(\boldsymbol{x}_\star) < g_{3a}(\boldsymbol{x}_\star) < g_{3b}(\boldsymbol{x}_\star)$ for all $\boldsymbol{x}_\star \in \mathcal{X}$

- [3 pt / 114 pts] Circle the letters of all the following that are **true**.

  (a) If ridge regression with a cross-validated λ was employed for linear models ~~with~~ under $\mathcal{H}_{3b}$ it is likely many of the b_j values would be set to exactly zero.

  (b) If the lasso with a cross-validated λ was employed for linear models ~~with~~ under $\mathcal{H}_{3b}$ it is likely many of the b_j values would be set to exactly zero.

  (c) The out-of-sample (oos) RMSE_{3b} under OLS is likely larger than the out-of-sample RMSE_{3b} if ridge regression was employed for linear models under $\mathcal{H}_{3b}$

For the remainder of the problem, we employ $\mathcal{H}_{3b}$ and $\mathcal{A} = \text{OLS}$. Let $p+1$ refer to the total number of columns in the design matrix under $\mathcal{H}_{3b}$ and $\mathcal{A} = \text{OLS}$.

Let $\mathbb{D} = \mathbb{D}_{\text{train}} \cup \mathbb{D}_{\text{select}}$ and run stepwise regression by training each iterated model on $\mathbb{D}_{\text{train}}$ and gauging oos performance on $\mathbb{D}_{\text{select}}$ where $K = 5$. Let g_{step} denote the model produced by this procedure and let p_{step} denote the number of linear terms in g_{step}.

- [5 pt / 119 pts] Circle the letters of all the following that are **true**.

  (a) There are no observations that are both $\in \mathbb{D}_{\text{train}}$ and $\in \mathbb{D}_{\text{select}}$

  (b) There are exactly 20% of the n observations in $\mathbb{D}_{\text{select}}$ if n is divisible by 5

  (c) It is likely that $p_{\text{step}} < p + 1$

  (d) If you randomize the order of $\mathbb{D}$, split it into a different $\mathbb{D}_{\text{train}} \cup \mathbb{D}_{\text{select}}$, then run the stepwise algorithm, it will definitely return the same model as when you ran it the first time

  (e) Using the residuals from g_{step}'s predictions on $\mathbb{D}_{\text{select}}$ will give an honest estimate of g_{step}'s future performance

We now use the nested cross-validation resampling procedure from class. Let $\mathbb{D} = \mathbb{D}_{\text{train}} \cup \mathbb{D}_{\text{select}} \cup \mathbb{D}_{\text{test}}$ and run stepwise regression by training each iterated model on $\mathbb{D}_{\text{train}}$ and gauging oos performance on $\mathbb{D}_{\text{select}}$ where $K_{\text{inner}} = 5$. This represents the number of folds among $\mathbb{D}_{\text{train}} \cup \mathbb{D}_{\text{select}}$. We then cross validate this cross validation with $K_{\text{outer}} = 4$. Let g_{final} denote the final model from this procedure.

- [7 pt / 126 pts] Circle the letters of all the following that are **true**.

  (a) There are exactly 20% of the n observations in $\mathbb{D}_{\text{select}}$ if n is divisible by 5

  (b) There are exactly 25% of the n observations in $\mathbb{D}_{\text{test}}$ if n is divisible by 4

  (c) If you randomize the order of $\mathbb{D}$, split it into a different $\mathbb{D}_{\text{train}} \cup \mathbb{D}_{\text{select}} \cup \mathbb{D}_{\text{test}}$, then run the stepwise algorithm, it will definitely return the same model g_{final} as when you ran it the first time

  (d) This method results in 4 potentially different g_{step} models

  (e) This method results in 5 potentially different g_{step} models

  (f) This method results in 20 potentially different g_{step} models

  (g) g_{final} is the best of the many g_{step} models produced in this procedure