STAT 4710: Practice Midterm Exam 2

Name

Contents

Instructions

Materials

The allowed materials are as stated on the Syllabus:

> "Students may consult all course materials, including course textbooks, for all assignments and assessments. For programming-based assignments (homeworks and exams), students may also consult the internet (e.g. Stack Overflow) for help with general programming tasks (e.g. how to add a dashed line to a plot). Students may not search the internet for help with specific questions or specific datasets on any homework or exam. In particular, students may not use solutions to problems that may be available online and/or from past iterations of the course."

Collaboration

The collaboration policy is as stated on the Syllabus:

> "Students are prohibited from collaborating on the quizzes or exams."

Writeup

Use this document as a starting point for your writeup, adding your solutions after "**Solution**". Add your R code using code chunks and add your text answers using **bold text**. Consult the preparing reports guide for guidance on compilation, creation of figures and tables, and presentation quality. In particular, if the instructions ask you to "print a table", you should use `kable`. If the instructions ask you to "print a tibble", you should not use `kable` and instead print the tibble directly.

Programming

The `tidyverse` paradigm for data visualization, manipulation, and wrangling is required. No points will be awarded for code written in base R.

We'll need to use the following `R` packages:

```
library(kableExtra)    # for printing tables
library(cowplot)       # for side by side plots
library(glmnetUtils)   # to run ridge and lasso
library(stat471)       # for class-specific functions
library(tidyverse)     # for everything else
```

Grading

The point value for each problem sub-part is indicated. Additionally, the presentation quality of the solution for each problem (as exemplified by the guidelines in Section 4 of the preparing reports guide will be evaluated on a per-problem basis (e.g. in this homework, there are three problems). There are 100 points possible on this midterm, 85 of which are for correctness and 15 of which are for presentation.

Submission

This is a practice exam, so please do not submit it. For your actual midterm exam, you will compile your writeup to PDF and submit to Gradescope.

Medicare costs at inpatient rehabilitation facilities

According to medicare.gov, an inpatient rehabilitation facility (IRF) is "a hospital, or part of a hospital, that provides intensive rehabilitation to inpatients. Many patients with conditions like stroke or brain injury are transferred or admitted to an inpatient rehabilitation facility." IRFs can incur significant medical costs, as quantified by the Medicare spending per beneficiary (MSPB) measure. In this exam, we will be building predictive models for the MSPB based on several attributes of IRFs. We will use a dataset (`irf-data.csv`) from The Centers for Medicare & Medicaid Services. The variables of interest are coded in these data using "measure codes". For the purposes of this exam, we are interested in 15 of the measure codes, presented in Table 1 along with shortened variable names and descriptions. The first variable in Table 1 (MSPB) is the response and the remaining variables are features.

Table 1: Features and response variable of interest in the inpatient rehabilitation facility data.

Measure code	Variable name	Variable description
I_020_01_MSPB_SCORE	mspb	Medicare Spending Per Beneficiary (MSPB)
I_006_01_SIR	uti_rate	Catheter-associated urinary tract infection rate
I_008_02_OBS_RATE	abilities_goals	Percentage of patients whose functional abilities were assessed and functional goals were included in their treatment plan
I_009_03_ADJ_CHG_SFCR_SCORE	self_care_ability_improved	Patients' ability to care for themselves changed between facility admission and discharge
I_010_03_ADJ_CHG_MOBL_SCORE	mobility_improved	Patients' ability to move around changed between facility admission and discharge
I_011_03_OBS_RATE	self_care_expectation_met	Percentage of patients who achieve or exceed a self-care ability expected for their condition at discharge
I_012_03_OBS_RATE	mobility_expectation_met	Percentage of patients who achieve or exceed the level of movement expected for their condition at discharge
I_013_01_OBS_RATE	fall_rate	Percentage of IRF patients who experience one or more falls with major injury during their IRF stay
I_015_01_SIR	cdi_rate	Clostridium difficile infection rate
I_016_01_OBS_RATE	flu_vax_rate	Influenza Vaccination Coverage Among Healthcare Personnel
I_017_01_PPR_PD_RSRR	readmission_rate_after	Rate of potentially preventable hospital readmissions 30 days after discharge from an IRF
I_018_01_PPR_WI_RSRR	readmission_rate_during	Rate of potentially preventable hospital readmissions during the IRF stay
I_019_02_DTC_RS_RATE	rate_return	Rate of successful return to home and community from an IRF
I_021_01_OBS_RATE	medications_reviewed	Percentage of patients whose medications were reviewed and who received follow-up care when medication issues were identified
I_022_01_ADJ_RATE	pressure_ulcers	Percentage of patients with pressure ulcers/injuries that are new or worsened

1 Wrangling (40 points for correctness; 0 points for presentation)

1.1 Import (8 points)

1. (4 points) Import the IRF data from `irf-data.csv` into a tibble called `irf_data_raw`. Print this tibble (no need to make a fancy table out of it).
2. (4 points) How many IRFs are represented in these data? Note that each IRF is coded using the `CMS Certification Number (CCN)`. How many measure codes are represented in these data?

1.2 Clean (24 points)

1. (4 points) Create a new tibble called `irf_data_1` from `irf_data_raw` that contains only the columns `CMS Certification Number (CCN)`, `Facility Name`, `City`, `State`, `Measure Code`, `Score`, renamed to `facility_id`, `facility_name`, `city`, `state`, `measure_code`, `score`, respectively. Print this tibble (no need to create a fancy table).
2. (4 points) The column `measure_code` contains the measure codes. Create a new tibble called `irf_data_2` from `irf_data_1` that contains only the rows representing measure codes in Table 1. Print this tibble (no need to create a fancy table). [Hint: Use the tibble `variables` underlying Table 1.]
3. (4 points) Table 8 of the data dictionary contains all of the measure codes in the original data (listed as "Provider Variables"). Why might we want to use only the subset of variables in Table 1 for our predictive models?
4. (4 points) The column `score` contains the value of each measure, with missing values indicated using the string `Not Available`. What fraction of IRFs have any missing scores? [Hint: Use the `any` function.]
5. (4 points) Create a new tibble called `irf_data_3` from `irf_data_2` that contains only those IRFs for which none of the scores are missing. Print this tibble (no need to create a fancy table). How many IRFs remain? [Hint: First create a list of IRFs for which none of the scores are missing using the code in the previous bullet point.]
6. (4 points) Convert the `score` column from `character` to `numeric`. Replace the `measure_code` column with a column named `feature`, which is a factor whose levels are named according to the second column of Table 1. Store the resulting tibble in `irf_data_4`. Print this tibble (no need to create a fancy table).

1.3 Tidy (8 points)

1. (4 points) Tidy the tibble `irf_data_4`, storing the result in a new tibble called `irf_data_tidy`. Print this tibble (no need to create a fancy table).
2. (4 points) What does each row in `irf_data_tidy` represent? How many rows are there?

2 Exploration (25 points for correctness; 5 points for presentation)

2.1 Response distribution (15 points)

1. (5 points) Print the mean, standard deviation, minimum, and maximum of `mspb` in a nice table.
2. (5 points) Produce a histogram of `mspb`, with a vertical dashed line at the mean. Comment on the shape of this distribution.
3. (5 points) Produce a (nice) table of the IRFs with the top 5 `mspb` values; for each IRF print its `facility_name`, `city`, `state`, and `mspb`. What cities (or city) do the two most costly IRFs come from?

2.2 Relationships among features (10 points)

1. (5 points) Some of the features appear to be related to each other, e.g. `self_care_ability_improved` and `self_care_expectation_met`; `mobility_improved` and `mobility_expectation_met`; `readmission_rate_during` and `readmission_rate_after`. Create scatter plots of these three pairs of features, displaying all three side by side, and comment on the degree to which each pair appears to be correlated.
2. (5 points) Discuss how the findings in the above bullet point might impact the fitted coefficients of ridge and lasso regressions.

3 Elastic net regression (25 points for correctness; 5 points for presentation)

Next, let's train penalized regression models to predict the case-fatality ratio based on the available features. We use the following train-test split (uncomment the code below once `irf_data_tidy` has been created):

```r
# num_train_samples <- 300
# set.seed(1)
# train_samples <- sample(1:nrow(irf_data_tidy), num_train_samples)
# irf_train <- irf_data_tidy %>% filter(row_number() %in% train_samples)
# irf_test <- irf_data_tidy %>% filter(!(row_number() %in% train_samples))
```

3.1 Training and tuning (15 points)

1. (3 points) Using `irf_train`, fit a 10-fold cross-validated elastic net regression of `mspb` on the 14 other features in Table 1, where the cross-validation is over `alpha` as well as `lambda`.
2. (3 points) Produce a plot of the CV error (minimized across `lambda` for each `alpha`) versus `alpha`. Which value of alpha performs best? What does this suggest about the number of features actually impacting the response?
3. (2 points) Produce a regular CV plot (showing CV error as a function of lambda) for the alpha selected in the previous bullet point. Based on this plot, how many features have nonzero coefficients in the model selected by the one-standard error rule?
4. (3 points) Produce the trace plot for the `glmnet` model from the previous bullet point. [Hint: To make the plot easier to read, change the y axis scale by appending `+ coord_cartestian(ylim = c(-0.02, 0.02))` after your `plot_glmnet` call.] What is the first feature that comes in with a positive coefficient? What is the first feature that comes in with a negative coefficient? Do the signs of these coefficients make sense to you? Why or why not?
5. (2 points) Produce a nice table with the features selected by the above `glmnet` model (excluding intercept) and their fitted coefficients. According to these coefficients, if an IRF increases its `rate_return` by 10 while keeping other things equal, how would this change the expected `mspb`?
6. (2 points) Compare the relative magnitudes of the coefficients based on the one-standard-error rule for `rate_return` and `uti_rate` using the table from part 5 and the trace plot from part 4. Do the table and plot contradict each other? Why or why not?

3.2 Performance evaluation (10 points)

Let's compare the performance of elastic net models for different `alpha`. While the fit objects for each `alpha` can be extracted from the original fit object from Section 3.1, we haven't learned how to do this. We will instead separately re-train models for different `alpha`.

1. (5 points) Train three cross-validated elastic net models—one for `alpha` = 0, one for `alpha` = 0.5, and one for `alpha` = 1—this time cross-validating over lambda only.

2. (5 points) Compute the RMSE prediction error for each of these three models, with `lambda` chosen based on the one-standard-error rule. Print these in a nice table, together with the corresponding `alpha` values (use `digits = 4` in your call to `kable`). Comment on the ordering among the prediction errors, and the extent to which it is consistent with the CV error versus `alpha` plot from Section 3.1.