Takehome Midterm

STAT 409 - Data Science II

Due at 4/30/2023, 5:00pm

Submission Rules (Important!)

- You must sumbit your work (report & code) by e-mail (`sjshin4TA@gmail.com`, and cc to your email).
- **Due date: 4/30 (Sun) 5:00pm**
  - If I get your mail after 4/30 (Sun) 5:00pm, you will lose 30% of the credits you earned.
  - If I get your mail after 4/30 (Sun) 5:15pm, NO credit!
- Additional rules:
  - Subject line of the email: `STAT409_Midterm_StuduentID` (ex: `STAT409_Midterm_2020150001`)
  - File name of your report: `STAT409_Midterm_StuduentID.pdf`
  - File name of your code: `STAT409_Midterm_StuduentID.R`
  - All functions and codes must be included in a single file.
- If you do NOT strictly follow these rules above, you additionally lose 5% of your credits.

Problems

[Notice] You should be very precise and detailed to earn full credit.

1. (**50pts**, Completely Randomized Design) Consider comparing mean of I groups under the following model.

$$y_{ij} \overset{\text{indep}}{\sim} N(\mu_i, \sigma^2), \qquad i = 1, \cdots I; j = 1, 2, \cdots, J. \tag{1}$$

with $n = I \times J$. Our goal is to test the following hypotheses:

$$H_0 : \mu_1 = \mu_2 = \cdots = \mu_I \qquad \text{vs} \qquad \text{Not } H_0. \tag{2}$$

   (a) (7pts) Compute the maximum likelihood estimator (MLE) of $\mu_i, i = 1, \cdots, I$.

   (b) (8pts) Note that (1) can be equivalently rewritten as the following linear model:

$$y_{ij} = \mu_i + \epsilon_{ij}, \qquad \epsilon_{ij} \sim N(0, \sigma^2), \qquad i = 1, \cdots I; j = 1, 2, \cdots, J. \tag{3}$$

   or equivalently

$$\mathbf{y} = \mathbf{X}\boldsymbol{\mu} + \boldsymbol{\epsilon} \tag{4}$$

where

$$\mathbf{y} = (y_{11}, \cdots, y_{1J}, y_{21}, \cdots, y_{2J}, \cdots, y_{I1}, \cdots, y_{IJ})^T \in \mathbb{R}^n$$
$$\boldsymbol{\epsilon} = (\epsilon_{11}, \cdots, \epsilon_{1J}, \epsilon_{21}, \cdots, \epsilon_{2J}, \cdots, \epsilon_{I1}, \cdots, \epsilon_{IJ})^T \in \mathbb{R}^n$$
$$\boldsymbol{\mu} = (\mu_1, \mu_2, \cdots, \mu_I)^T \in \mathbb{R}^I$$

Provide a proper design matrix $\mathbf{X}$ and compute the ordinary least square (OLS) estimator of $\mu_i, i = 1, 2, \cdots, I$.

(c) (5pts) Compute an orthogonal projection matrix on col$\{\mathbf{X}\}$ denoted by $\mathbf{P_X}$ under (4).

(d) (5pts) Compute both fitted-value vector $\hat{\mathbf{y}}$ and residual vector $\hat{\mathbf{e}}$ under (4).

(e) (10 pts) It is well-known that the F-test statistic for testing (2) is

$$F = \frac{MS_{\text{trt}}}{MSE} = \frac{\sum_{i=1}^I \sum_{j=1}^J (\bar{y}_i - \bar{y})^2/(I-1)}{\sum_{i=1}^I \sum_{j=1}^J (y_{ij} - \bar{y}_i)^2/(I(J-1))} \sim F(I-1, I(J-1)) \quad \text{under } H_0 \quad (5)$$

where $\bar{y}_i = \sum_{j=1}^J y_{ij}/J$ and $\bar{y} = \sum_{i=1}^I \sum_{j=1}^J y_{ij}/n$. To justify (5), it is essential that MS_{trt} and MSE (or equivalently their numerators) are independent. Justify the independence using the geometry of regression based on $\mathbf{P_X}$ under (4).

(f) (15 pts) Now, you have the following data from (1), where $I = 5$ and $J = 10$ (downloadable at https://www.dropbox.com/s/rol1p34tuq8cgll/crd.csv?dl=0):

	Group1	Group2	Group3	Group4	Group5
1	1.10	1.42	5.89	3.74	1.62
2	2.18	1.98	2.60	3.32	0.04
3	3.59	0.61	5.39	4.08	1.16
4	0.87	-0.04	5.75	2.72	3.90
5	1.92	2.78	3.80	2.22	2.62
6	2.13	-1.31	1.35	2.40	3.99
7	2.71	1.88	4.28	1.27	1.69
8	1.76	1.04	3.20	2.10	1.91
9	3.98	2.01	4.59	2.44	1.82
10	1.86	1.43	4.09	2.75	0.80

Write your own `R` (or other language) code to compute MS_{trt}, MSE, and F-statistic based on $\mathbf{P_X}$ under (4). You must use regression computation we have learned in the class, and do not directly compute sample averages such as $\bar{y}_i$ or $\bar{y}$ nor any built-in function for the linear regression such as `lm()`. Using your code, please compute MS_{trt}, MSE, and F-statistic.

2. (**50pts**, Nonlinear Regression) You are given two independent sets of training and test data sets. downloadable from the following link, respectively:

- Train: https://www.dropbox.com/s/k9ob2ygtp8g0wih/train.csv?dl=0
- Test: https://www.dropbox.com/s/330llkf731xpwqc/test.csv?dl=0

To fit the model, we consider a nonlinear regression model:

$$y_i = f(x_i) + \epsilon_i, \qquad i = 1, 2, \cdots, n \quad (6)$$

where ϵ_i is a random error.

Scatter plots for training and test sets are given in the following:

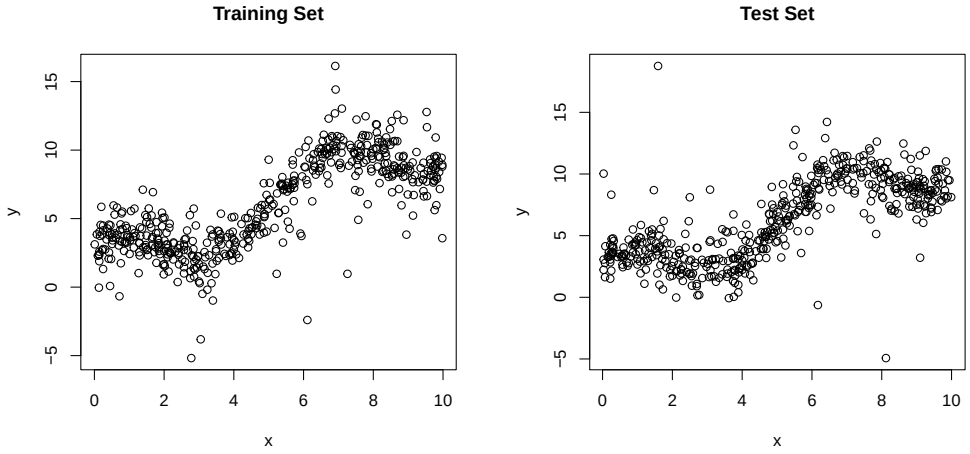

Figure 1: Scatter plots of training (left) and test (right) sets.

(a) (5pts) A ploynomial regression assumes

$$f(x) = \beta_0 + \beta_1 x + \cdots \beta_p x^p$$

where p is a unknown positive integer. Fit the polynomial regression for different values of $p \in \{1, 2, \cdots, 10\}$, and draw fitted regression lines on the scatter plot.

(b) (5pts) Using the result from (a), draw a plot of R^2 as a function of $p \in \{1, 2, \cdots, 10\}$ (i.e., scatter plot of p and R^2).

(c) (7pts) From the previous results, explain why $R^2 = 1 - SSE/SST$ is not a proper criterion for model selection in terms of training-error.

(d) (5pts) Find the best model for the polynomial regression (i.e. p) that optimizes the performance in terms of Bayesian Information Criteria (BIC) defined as $BIC(p) = SSE + \log n \cdot p$. $$

(e) (7pts) Fix $p = 10$ for the polynomial regression, and then apply both LASSO and Ridge regression (including tuning) to the data, then plot the fitted curves on the scatter plot. You may use functions in `glmnet` package.

(f) (7pts) Under (6), Apply B-spline mean regression to find the best fit for the data and then plot the fitted curve on the scatter plot. You must include the tuning step for the `df` in `bs()` function in `splines` package.

(g) (7pts) Apply B-spline median regression (i.e., quantile regression with $\tau = 0.5$), and then plot the fittect curve on the scatter plot. You may use `rq()` function in `quantreg` package. Again, you must include the tuning step for the `df` in `bs()` function in `splines` package.

(h) (7pts) Among those that you trained above, which one do you think the best model is? Justify your answer.