Machine Learning for Business Analytics
Assignment 4
Due: December/ 26 23:59 (GMT+8)

Instruction

- There are two parts in this assignment. You only have to hand in Part 1 regarding the Julia Programming. **Part 2 will not be graded**; of notice is that similar questions may show up in midterm or final.
- Please hand in your Julia code with a **PDF file**. The wrong format will not be graded.
- **Copying the assignment will result in zero points.**
- Late submission will be graded according to the following rule:
  Your grade = original grade × (1 − 0.05*h*), if you submit *h* hours after the deadline.

Part 1 Programming Exercises (100%)

1. (House price, 25%) In this question, we use the house price dataset of house_sales_data() in the Julia package VMLS. Suppose the house price follows

$$\widehat{price}_i = \beta_0 + \beta_1 area_i + \beta_2 {area_i}^2 + \beta_3 beds_i + \beta_4 location1_i + \beta_5 location2_i + \beta_6 location3_i,$$

   where $locationj_i = 1$ if i at in location j and 0 otherwise for $j = 1,2,3,$ and β_k for $k = 0, 1, 2, 3, 4, 5,$ and 6 are the least square solution.
   a. Find β_k and what do they imply?
   b. What is the predicted house price at $area = 1.01$, $beds = 4$, and $location = 4$.
   c. Plot $\widehat{price}_i$ in the y-axis and $price$ in the x-axis and a 45-degree line.

2. (Cross-validation, 25%) Following question 1, use 5 randomly chosen folds and separate the data into training and testing datasets. (You may choose a reasonable number of data in each fold.)
   a. What are all the coefficients for the 5 folds?
   b. What are the training errors and testing errors?

3. (IRIS, 25%) In this question, we use the dataset iris_data() in the package VMLS. There are three classes and four features in the datasets. The three classes are Virginia, Setosa, and Versicolor. And the four features are:
   - x_1: the sepal length in cm
   - x_2: is the sepal width in cm
   - x_3: is the petal length in cm
   - x_4: is the petal width in cm.

   a. Build a multiclass (linear) classifier with intercept for the iris dataset using all four

features. What are the coefficients?

b. Construct the 3×3 confusion matrix for this example.

4. (Time series, 25%) In this question, we use the dataset temperature_data() from the package VMLS. Build an auto-regressive time series model with different memory values.
   a. Build an auto-regressive time series model with the memory of $1, 5$, and 10.
   b. Plot the temperature and the predictions from the three memory of $1, 5$, and 10 in one diagram. What do you observe?

Part 2 Mathematical Exercises (0%)

Chapter 12 Exercises

**In this chapter, (12.1) refers to "minimize $||Ax - b||^2$"

1. In least squares, the objective (to be minimized) is

$$||Ax - b||^2 = \sum_{i=1}^{m} (\tilde{a}_i^T x - b_i)^2,$$

where a $\tilde{a}_i^T$ are the rows of *A*, and the *n*-vector *x* is to chosen. In the weighted least squares problem, we minimize the objective

$$\sum_{i=1}^{m} w_i (\tilde{a}_i^T - b_i)^2$$

where w_i are given positive weights. The weights allow us to assign different weights to the different components of the residual vector. (The objective of the weighted least squares problem is the square of the weighted norm, $||Ax - b||_w^2$, as defined in exercise 3.28 on your textbook.)

1. (a) Show that the weighted least squares objective can be expressed as $\| D(Ax - b) \|^2$ for an appropriate diagonal matrix *D*. This allows us to solve the weighted least squares problem as a standard least squares problem, by minimizing $\| Bx - d \|^2$, where *B* = *DA* and *d* = *Db*.
2. (b) Show that when *A* has linearly independent columns, so does the matrix *B*.
3. (c) The least squares approximate solution is given by $\hat{x} = (A^T A)^{-1} A^T b$. Give a similar formula for the solution of the weighted least squares problem. You might want to use the matrix *W* = **diag**(*w*) in your formula.

2. Suppose *A* is an *m* × *n* matrix with linearly independent columns and QR factorization *A* = *QR*, and *b* is an *m*-vector. The vector $A\hat{x}$ is the linear combination of the columns of *A* that is closest to the vector *b*, i.e., it is the projection of *b* onto the set of linear combinations of the columns of *A*.

(a) Show that $A\hat{x} = QQ^T b$. (The matrix QQ^T is called the projection matrix.)
(b) Show that $\| A\hat{x} - b \|^2 = \| b \|^2 - \| Q^T b \|^2$. (This is the square of the distance between b and the closest linear combination of the columns of *A*.)

3. In the special case *n* = 1, the general least squares problem (12.1) reduces to finding a scalar *x* that minimizes $\| ax - b \|^2$, where *a* and *b* are *m*-vectors. (We write the matrix *A* here in lower case, since it is an m-vector.) Assuming a and b are nonzero, show that $\| a\hat{x} - b \|^2 = \| b \|^2 (\sin\theta)^2$, where θ is angle between *a* and *b*. This shows that the

optimal relative error in approximating one vector by a multiple of another one depends on their angle.

4. A generalization of the least-squares problem (12.1) adds an affine function to the least-squares objective,

$$\text{minimize } \| Ax - b \|^2 + c^T x + d,$$

where the *n*-vector *x* is the variable to be chosen, and the (given) data are the *m*×*n* matrix *A*, the m-vector *b*, the *n*-vector *c*, and the number *d*. We will use the same assumption we use in least squares: The columns of *A* are linearly independent. This generalized problem can be solved by reducing it to a standard least squares problem, using a trick called completing the square.

Show that the objective of the problem above can be expressed in the form

$$\| Ax - b \|^2 + c^T x + d = \| Ax - b + f \|^2 + g,$$

for some *m*-vector *f* and some constant *g*. It follows that we can solve the generalized least squares problem by minimizing $\| Ax - (b - f) \|$, an ordinary least squares problem with solution $\hat{x} = A^{\dagger}(b - f)$.

Chapter 13 Exercises

1. Consider the straight-line fit ($\hat{f}(x) = \theta_1 + \theta_2 x$), with data given by the *N*-vectors x^d and y^d. Let $r^d = y^d - \hat{y}^d$ denote the residual or prediction error using the straight-line model. Show that $\mathbf{rms}(r^d) = \mathbf{std}(y^d)\sqrt{1-\rho^2}$, where ρ is the correlation coefficient of x^d and y^d(assumed non-constant). This shows that the RMS error with the straight-line fit is a factor $\sqrt{1-\rho^2}$ smaller than the RMS error with a constant fit, which is $\mathbf{std}(y^d)$. It follows that when x^d and y^d are highly correlated $(\rho \approx 1)$ or anti-correlated $(\rho \approx -1)$, the straight-line fit is much better than the constant fit.

2. Suppose that the *N*-vector x gives the value of a (scalar) Boolean feature across a set of *N* examples. (Boolean means that each x_i has the value 0 or 1. This might represent the presence or absence of a symptom, or whether or not a day is a holiday.) How do we standardize such a feature? Express your answer in terms of p, the fraction of xi that have the value 1. (You can assume that $p > 0$ and $p < 1$; otherwise the feature is constant.)

3. Suppose that the n-vector x and the m-vector y are thought to be approximately related

by a linear function, i.e., $y \approx Ax$, where A is an $m \times n$ matrix. We do not know the matrix A, but we do have observed data,

$$x^{(1)}, \ldots, x^{(N)}, \quad y^{(1)}, \ldots, y^{(N)}.$$

We can estimate or guess the matrix A by choosing it to minimize

$$\sum_{i=1}^{N} ||A\,x^{(i)} - y^{(i)}||^2 = ||AX - Y||^2,$$

where $X = [x^{(1)} \ldots x^{(N)}]$ and $Y = [y^{(1)} \ldots y^{(N)}]$. We denote this least squares estimate as $\hat{A}$. (The notation here can be confusing, since X and Y are known, and A is to be found; it is more conventional to have symbols near the beginning of the alphabet, like A, denote known quantities, and symbols near the end, like X and Y, denote variables or unknowns.)

(a) Show that $\hat{A} = YX^+$, assuming the rows of X are linearly independent.

(b) Suggest a good way to compute $\hat{A}$, and give the complexity in terms of $n, m,$ and N.

Chapter 14 Exercises

1. Let $\tilde{f}(x)$ denote the continuous prediction of the Boolean outcome y, and $\hat{f}(x) = \mathbf{sign}(\tilde{f}(x))$ the actual classifier. Let σ denote the RMS error in the continuous prediction over some set of data, i.e.,

$$\sigma^2 = \frac{(\tilde{f}(x^{(1)}) - y^{(1)})^2 + \cdots + (\tilde{f}(x^{(N)}) - y^{(N)})^2}{N}$$

Use the Chebyshev bound to argue that the error rate over this data set, i.e., the fraction of data points for which $\hat{f}(x^{(i)}) \neq y^{(i)}$, is no more than σ^2, assuming $\sigma < 1$.

2. We consider the least squares *K*-class classifier of 14.3.1 in your textbook. We associate with each data point the *n*-vector x, and the label or class, which is one of $1, \ldots, K$. If the class of the data point the n-vector x, and the label or class, which is one of $1, \ldots, K$. If the class of the data point is k, we associate it with a K-vector y, whose entries are $y_k = +1$ and $y_j = -1$ for $j \neq k$. (We can write this vector as $y = 2e_k - 1$.) Define $\tilde{y} = (\widetilde{f_1}(x), \ldots, \widetilde{f_K}(x))$, which is our (real-valued or continuous) prediction of the label y. Our multi-class prediction is given by $\hat{f}(x) = argmin_{k=1,\ldots,K} \widetilde{f_k}(x)$. Show that $\hat{f}(x)$ is also the index of the nearest neighbor of $\tilde{y}$ among the vectors $2e_k - 1$, for $k = 1, \ldots, K$. In other words, our guess $\hat{y}$ for the class is the nearest neighbor of our continuous prediction $\tilde{y}$, among the vectors that encode the class labels.

Chapter 15 Exercises

1. We consider the special case of the multi-objective least squares problem in which the variable x is a scalar, and the k matrices A_i are all 1×1 matrices with value $A_i = 1$, so $J_i = (x - b_i)^2$. In this case our goal is to choose a number x that is

simultaneously close to all the numbers $b_1, \ldots, b_k$. Let $\lambda_1, \ldots, \lambda_k$ be positive weights, and $\hat{x}$ the minimizer of the weighted objective

$$J = \lambda_1 J_1 + \cdots + \lambda_k J_k = \lambda_1 \left||A_1 x - b_1|\right|^2 + \cdots + \lambda_k \left||A_k x - b_k|\right|^2$$

Show that $\hat{x}$ is a weighted average (or convex combination) of the numbers $b_1, \ldots, b_k$, i.e., it has the form

$$x = w_1 b_1 + \cdots + w_k b_k,$$

where w_i are nonnegative and sum to one. Give an explicit formula for the combination weights w_i in terms of the multi-objective least squares weights λ_i.

2. Consider the regularized data fitting problem $||\mathrm{y} - \mathrm{A}\theta||^2 + \lambda \left||\theta_{2:p}\right||^2$. Recall that the elements in the first column of A are one. Let $\hat{\theta}$ be the solution of above equation, i.e., the minimizer of

$$||\mathrm{A}\theta - \mathrm{y}||^2 + \lambda \left(\theta_2^2 + \cdots + \theta_p^2\right),$$

and let $\tilde{\theta}$ be the minimizer of

$$||\mathrm{A}\theta - \mathrm{y}||^2 + \lambda ||\theta||^2 = ||\mathrm{A}\theta - \mathrm{y}||^2 + \lambda \left(\theta_1^2 + \theta_2^2 + \cdots + \theta_p^2\right),$$

In which we also penalize θ_1. Suppose columns 2 through p of A have mean zero (for example, because features 2,…,*p* have been standardized on the data set). Show that $\hat{\theta}_k = \tilde{\theta}_k$ for $k = 2, \ldots, p$.

3. Consider a linear dynamical system given by $x_{t+1} = Ax_t + Bu_t$, where the n-vector x_t is the state at time *t*, and the m-vector u_t is the input at time *t*. The goal in regulation is to choose the input so as to make the state small. (In applications, the state x_t = 0 corresponds to the desired operating point, so small x_t means the state is close to the desired operating point.) One way to achieve this goal is to choose u_t so as to minimize

$$\left||x_{t+1}|\right|^2 + \rho \left||u_t|\right|^2,$$

where ρ is a (given) positive parameter that trades off using a small input versus making the (next) state small. Show that choosing u_t this way leads to a state feedback policy $u_t = Kx_t$, where K is an $m \times n$ matrix. Given a formula for K (in terms of A, B, and ρ). If an inverse appears in your formula, state the conditions under which the inverse exists.