STA 5106 Computational Methods in Statistics I

Department of Statistics

Florida State University

Class 15

October 15, 2019

Chapter 2

Numerical Linear Algebra

Multiple Linear Regression

- The observations may belong to different sample times t_1, t_2, . . . , t_m such that

$$y(t_i) = \sum_{j=1}^{n} b_j x_j(t_i) + \varepsilon(t_i)$$

- In a matrix form these equations can be restated as

$$y = Xb + \varepsilon$$

where

$$y = \begin{pmatrix} y(t_1) \\ y(t_2) \\ \vdots \\ y(t_m) \end{pmatrix} \quad X = \begin{pmatrix} x_1(t_1) & x_2(t_1) & & x_n(t_1) \\ x_1(t_2) & x_2(t_2) & & x_n(t_2) \\ & & \ddots & \\ x_1(t_m) & x_2(t_m) & & x_n(t_m) \end{pmatrix} \quad b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} \quad \varepsilon = \begin{pmatrix} \varepsilon(t_1) \\ \varepsilon(t_2) \\ \vdots \\ \varepsilon(t_m) \end{pmatrix}$$

Orthogonal Transformations

- We are interested in solving the problem

$$\hat{b} = \arg\min_b \| y - Xb \|^2$$

- Let Q be an $m \times m$ orthogonal matrix, i.e. $QQ^T = Q^TQ = I_m$, and

$$y^* = Qy = QXb + Q\varepsilon = X^*b + \varepsilon^*$$

- Multiplication of an orthogonal matrix does not change the length (2-norm) of a vector.

- Therefore,

$$\hat{b} = \arg\min_b \| y - Xb \|^2 = \arg\min_b \| y^* - X^*b \|^2$$

Upper Triangular Matrix

- If we can select a Q in such a way that X^* is an upper triangular matrix, then

$$\begin{pmatrix} y_1^* \\ y_2^* \end{pmatrix} = \begin{pmatrix} X_1^* \\ 0 \end{pmatrix} b + \begin{pmatrix} \varepsilon_1^* \\ \varepsilon_2^* \end{pmatrix}$$

- Therefore,

$$\| y^* - X^* b \|^2 = \| y_1^* - X_1^* b \|^2 + \| y_2^* \|^2$$

and

$$\hat{b} = \arg\min_b \| y^* - X^* b \|^2 = \arg\min_b \| y_1^* - X_1^* b \|^2$$

- X_1^* being upper triangular the solution can be found by the backward substitution.

Householder Transformation

- **Definition 7** For a vector $v \in \mathbf{R}^m$, an $m \times m$ matrix H of the form

$$H = I_m - 2vv^T/(v^Tv)$$

is called a **Householder reflection matrix**.

- For a given $x \in \mathbf{R}^m$, we want to form an H, a householder matrix, in such a way that Hx has all but the first entry as zeros. In other words, $\boldsymbol{Hx = \lambda e_1}$ **for some constant** $\boldsymbol{\lambda}$**.**

- We found that v and Hx have the following forms:

$$v = x + \text{sign}(x_1)\, \|x\| e_1$$
$$Hx = -\,\text{sign}(x_1)\, \|x\| e_1$$

Householder Transformation

- Let H_j (j = 1, …, m) be the $m \times m$ matrix generated as:

$$H_j = \begin{pmatrix} I_{j-1} & 0 \\ 0 & \tilde{H}_j \end{pmatrix}$$

where $\tilde{H}_j = I - 2\frac{\tilde{v}\tilde{v}^T}{\tilde{v}^T\tilde{v}} \in \mathbf{R}^{(m-j+1)\times(m-j+1)}$

- Let X be an $m \times n$ matrix, $X^{(1)}$ be the result after the first transformation (j = 1), $X^{(2)}$ after the second transformation (j = 2) and so on. That is, $X^{(1)} = H_1X$, $X^{(2)} = H_1X^{(1)}$.

- Let

$$X = \begin{pmatrix} x_{11} & x_{12} & & x_{1n} \\ x_{21} & x_{22} & & x_{2n} \\ & & \ddots & \\ x_{m1} & x_{m2} & & x_{mn} \end{pmatrix}$$

Householder Transformation

- Then

$$X^{(1)} = H_1 X = \begin{pmatrix} x_{11}^1 & x_{12}^1 & & x_{1n}^1 \\ 0 & x_{22}^1 & & x_{2n}^1 \\ & & \ddots & \\ 0 & x_{m2}^1 & & x_{mn}^1 \end{pmatrix}, \quad X^{(2)} = H_2 X^{(1)} = \begin{pmatrix} x_{11}^1 & x_{12}^1 & & x_{1n}^1 \\ 0 & x_{22}^2 & & x_{2n}^2 \\ 0 & 0 & \ddots & \\ 0 & 0 & & x_{mn}^2 \end{pmatrix},$$

- Finally,

$$X^{(n)} = H_n H_{n-1} \cdots H_1 X = \begin{pmatrix} x_{11}^1 & x_{12}^1 & & x_{1n}^1 \\ 0 & x_{22}^2 & & x_{2n}^2 \\ 0 & 0 & \ddots & \\ 0 & 0 & & x_{nn}^n \\ 0 & 0 & & 0 \\ \cdots & & & \\ 0 & 0 & & 0 \end{pmatrix}$$

Singular Value Decomposition

- **Theorem 2** For any $X \in \mathbf{R}^{m \times n}$ (assuming $m \geq n$) there exist orthogonal matrices $U \in \mathbf{R}^{m \times m}$ and $V \in \mathbf{R}^{n \times n}$ such that

  $$U^T XV = \Sigma, \text{ where } \Sigma = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n) \in \mathbf{R}^{m \times n},$$

  and where $\sigma_1 \geq \sigma_2 \geq \ldots, \sigma_n \geq 0$.

  σ_i' s are called the singular values of X and the columns of U and V are called the singular vectors of X.

- For systems with large number of components, it is common to **reduce dimensions** before any statistical analysis.

- If the original observation space is $\mathbf{R}^D$, then the problem reduces to finding an appropriate projection that takes elements of $\mathbf{R}^D$ to elements of $\mathbf{R}^M$ ($M < D$) in a linear fashion.

PCA Algorithm

- **Algorithm 20 (PCA of Given Data)** Let X be the $D \times n$ matrix where **each column** denotes an independent observation vector for the random vector x.

  1. Find the sample covariance matrix $C \in \mathbf{R}^{D \times D}$ of the elements of X,

  2. Compute the singular value decomposition (SVD) of C to obtain the orthogonal matrix $U \in \mathbf{R}^{D \times D}$,

  3. Set U_1 to be the first M columns of U, and,

  4. define $Z = U_1^T X \in \mathbf{R}^{M \times n}$.

Chapter 3

Non-Linear Statistical Methods

Non-linear Optimization

- An instance of nonlinear optimization problems in statistics occurs in cases of **maximum likelihood estimation**.

- In general, **the solution which maximizes the likelihood function** can be found out by **seeking the roots of the first derivative of the likelihood function.**

- **Simple Iteration** we solve an equivalent problem of finding the fixed point of another function $g(x) = x + f(x)$.

$$f(x^*) = 0 \Leftrightarrow g(x^*) = x^*.$$

- The iteration is given by the formula:

$$x_{i+1} = g(x_i) = x_i + f(x_i) \ .$$

Newton-Raphson’s Method

- Newton-Raphson’s Method is one of the most popular techniques used in numerical root finding or fixed point estimation.

- Iteration formula:

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}, \quad (f'(x_i) \neq 0)$$

- Newton-Raphson is one of the fastest known algorithms for root finding in general situations.

Likelihood Maximization

- We look at maximizing likelihood for parameter θ given an observation x (= $\{x_1, \ldots, x_n\}$) and a likelihood function $f(x|\theta)$.

- Let the data vector x be made up of two components $x = \{x_o, x_m\}$ where x_o : observed part, x_m: missing part. Our goal is to solve for the maximizer:

$$\hat{\theta} = \arg\max_{\theta} f(x_o \mid \theta)$$

- **Expectation Maximization (EM) algorithm** is beneficial mostly where $f(x|\theta)$ is easier to maximize compare to maximizing $f(x_o|\theta)$.

Algorithm

- **Algorithm 30 (EM Algorithm)**

  Choose an initial value for θ_0 and set $k = 0$.

  1. **Expectation Step:** Compute

  $$Q(\theta \mid \theta_k, x_o) = E[\log f(x_o, x_m \mid \theta) \mid \theta_k, x_o].$$

  2. **Maximization Step:** Set

  $$\theta_{k+1} = \arg\max_{\theta} Q(\theta \mid \theta_k, x_o).$$

  3. Check convergence. If not converged, set $k = k + 1$ and go to Step 1.

Chapter 9

Statistical Pattern Recognition

Classification

- Consider the problem of classifying an observation $X \in \mathbf{R}^{\mathrm{n}}$ into one of the following classes: $C_1, C_2, \ldots, C_k$.
- Partitioning the observation space $\mathbf{R}^{\mathrm{n}}$ into k regions: if an observation falls into region i, we declare it to be of class C_i.
- In most cases, one uses a mapping from $\mathbf{R}^{\mathrm{n}}$ to $\mathbf{R}^{\mathrm{d}}$ for a $d << n$ to map observations into a more manageable space.
- Let g: $\mathbf{R}^{\mathrm{n}} \rightarrow \mathbf{R}^{\mathrm{d}}$ be such a map. For an observation $X \in \mathbf{R}^{\mathrm{n}}$, the vector $x = g(X)$ is called a **feature vector** or a representation of X.
- For example, obtaining U through (PCA).

Feature Space

- The space of all feature vectors is called the **feature space**. This is the space in which a statistical analysis is performed to obtain pattern classification.

- Metric methods are often used to perform classification. In case the feature space is Euclidean, $\mathbf{R}^d$, one can use the *p*-norm:

$$\| x_1 - x_2 \| = \left(\sum_{l=1}^{d} | x_1(l) - x_2(l) |^p \right)^{1/p}$$

- $p = 2$ provides the usual Euclidean norm and is used most often in practice.

Nearest Neighbor

- **Training Data:** This data is mostly labeled, i.e. with each observation we are provided the class to which it belongs.

- **Test Data:** The test data serves as a set of future observations to which a classifier is applied and tested.

- **Nearest Neighbor (NN) Classifier** In this classifier, one computes the distance between $y = g(Y)$ and all possible training vectors $g(X_i^j)$, and assigns that class to Y that has the nearest element to y. That is,

$$\hat{i} = \arg\min_{1\le i\le k}(\min_j d(g(Y), g(X_i^j))$$

K-Means Clustering Algorithm

- The **K-means algorithm** is an algorithm for putting *N* data points in an *I*-dimensional space into *K* clusters. Each cluster is parameterized by a vector $m^{(k)}$ called its mean.

- **Assignment step.** Compute indicator variables:

$$r_k^{(n)} = \begin{cases} 1 & \text{if } \hat{k}^{(n)} = k \\ 0 & \text{if } \hat{k}^{(n)} \neq k \end{cases}$$

- **Update step.** Update the means

$$m^{(k)} = \frac{\sum_n r_k^{(n)} x^{(n)}}{\sum_n r_k^{(n)}}$$

Midterm Report

Criteria for a good report:

- Clear description of the goal (problem statement)
- Clear description of the approach
  - Data (training set, testing set)
  - Method (dimension reduction, classification)
- Clear description of the results
  - Overall performance
  - Illustrative examples
  - Figures: legible with legends, captions, and labels
  - Tables: legible with captions.
- Summary

Additional Work

- Result when $k > 40$, particular $k = 644$.
- Give examples to show success and failure using low dimensional representation.
- Time cost as a function of k
- K-Nearest Neighbor (KNN)
- Statistical analysis

Some Tips on Presentation

- Large font (make the content easy to read)

- More figures, less text (make the content more intuitive). This is different from the notes in class where all details are needed.

- Good logic in the presentation: Motivation (General problem) → Specific Problem → current methods → your method → result → summary.

Peer Review on Presentation

- **Factors to consider:**
  - Structure: Clear introduction? Clear method description? Clear result? Clear conclusion?
  - Presentation: Loud enough? Clear enunciation? Speaking to the audience, not the screen?
  - Slides: Clearly make the important points? Not cluttered? Graphs and text large enough to see? Too many slides and/or too little on each?