Scientific Computing Semester Review

1. MATLAB Basic Command

- 2D plot, 3D plot
- m-file function
- array,matrix manipulation(vector calculation in MATLAB)

2. Financial Computing

- Compounding: basic formula:$f = p(1+r)^n$/rule of 70/rule of 200/how to compute mortgage
- Internal Rate of Return: The rate at an investment plan breaks even

$$NPV = \sum_{i=0}^{n} \frac{c_i}{(1+r)^i} = 0 \qquad NFV = \sum_{i=0}^{n} c_i(1+r)^{n-i} = 0$$

3. K-means clustering

- Objective: Deduction for unlabeled data, difference between input data set and cluster center should be as small as possible.
- Objective function:

$$J(X_{d\times n}; C_{d\times m, m<<n}, A_{n\times m}) = \sum_{j=1}^{m}\sum_{i=1}^{n} a_{ij} \parallel x_i - c_j \parallel^2 \qquad A : assignment\ matrix$$

- Step: fixed C to find best A, fixed A to find best C, repeat above til converge

$$a_{ij} \begin{cases} 1\ if j = arg\ min \parallel x_i - c_q \parallel^2 \\ 0, otherwise \end{cases} \qquad c_i = \frac{\sum_{i=1}^{n} a_{ij}x_i}{\sum_{i=1}^{n} a_{ij}}$$

  * guarantee to converge!(J is reduced repeatedly)

4. Image Compression

- True color: Each pixel is represented by [R,G,B]vector
  Index-color: Each pixel is represented by an index into a color map of 2^b colors
- Compression Ratio

$$\rho = \frac{before}{after} = \frac{m*n*3*8}{m*n*\log_2 c + c*3*8} \approx \frac{24}{\log_2 c}$$

- Blocked Base Compression Ratio(application of k-means clustering):

$$\rho = \frac{before}{after} = \frac{m*n*3*8}{\frac{m}{q}*\frac{n}{q}*\log_2 c + c*3*8} \approx \frac{24q^2}{\log_2 c + \frac{24cq^4}{m*n}}$$

5. MLE for PDF

- Maximum Likelihood Estimation: A way to find statistical model with optimum data
  ⇒ Formulate objective function and derive parameter
  ⇒ Inequality of Arithmetic and Geometric Means

$$\frac{\sum_{i=1}^{n} x_i}{n} \geq \left(\prod_{i=1}^{n} x_i\right)^{1/n}$$

  equality holds only when $x_1 = x_2 = \cdots = x_n$

- Probability Density Function:
  - $J(p,q,r) = p^{n1}q^{n2}r^{n3}$,with $p+q+r=1$,$p,q,r \geq 0$
  - Common- Guassian Distribution:$g(x,\mu,\sigma^2) = \frac{1}{\sigma\sqrt{2\pi}}\, exp[-0.5(\frac{x-\mu}{\sigma})^2]$

    MLE: $\mu = \frac{1}{n}\sum x_i \quad \sigma^2 = \frac{1}{n}\sum(x_i-\mu)^2$

6. ML for Classification

- K-nearest Neighbour Classifier: Find first k nearest neighbour of a given point, determine the class by voting among those points.
  - Voronoi Diagram
  - PROS:intuitive/no computation for model construction
    CONS:massive computation when data is big/no straightforward way
- Quadratic Classifiers: Select a class of parameterized PDF then identify parameters via MLE
  ⇒ test stage: $C = arg\ maxPr(C) * pdf_c(x)$
  - Characteristic: decision boundary is quadratic function
  - PROS: easy compute when data is small/easy for LOOCV
    CONS: covariance matrix is hard to compute or not exist/cannot handle bi-model data
- Naive Bayes Classifiers:
  Identify class pdf $pdf_c = pdf_{1,c} * pdf_{2,c} * \cdots * pdf_{d,c}$
  ⇒ test stage: $C = arg\ maxPr(C) * pdf_c(x)$

- Characteristic: statistical independent/each feature governed by a PDF
- PROS:Fast computation/Robust than QC
  CONS:not able to deal bi-model/class boundary not complex as QC

- Deep Neural Network: given desired i/o pairs and construct a model with it(structure identification & parameter identification)
  $\Rightarrow$ Multilayer perceptron
  - >3 layers(deep neural network) for arbitrary regions
  - Training: gradient descent(check:sec 11.)/Gauss-Newton method
  - Chain rule & partial derivative to implement a basic math model

- Performance Evaluation: recognition rate$\uparrow$,error rate$\downarrow$,computation load$\downarrow$
  - Stage:training set/validation set/test set
  - Method:inside test/one-side holdout test/two-side holdout test/M-fold cross validation/leave one-out cross validation

7. Data Fitting and Regression Analysis

- Data fitting :
  1. Using \ to solve basic regression problem $A\theta = b \;\Rightarrow\; \theta = A\backslash b$
  2. Error sum function is the objective function we want to minimize, can be shown as sum of squared error$\sum[y_i - f(x_i)]^2$
  3. MATLAB provide `polyfit` and `polyval` to simplify our codes.

- Least-squares Estimate
  1. $E(\theta) = (A\theta - b)^T(A\theta - b)\ \nabla_\theta(\theta) = 0 \Rightarrow \theta = (A^TA)^{-1}A^Tb$
  2. Application: distance to a line/plane
     $$d^2 = \left(\frac{a}{\sqrt{a^2+b^2}}\,x_0 + \frac{b}{\sqrt{a^2+b^2}}\,y_0 - \frac{c}{\sqrt{a^2+b^2}}\right)^2$$

- Non-linear Regression Problem $\Rightarrow$ Transformationėxample: $y = ae^{bx} \Rightarrow lny = lna + bx$

8. Principal Component Analysis

- Objective: Reduction of unlabelled data with dimensionality reduction, data variance $\uparrow$
  * not design for clasification problem

- i/o: zero justified dataset $\Rightarrow$ unit vector $\mathbf{u}$ that square sum of dataset's projection onto $\mathbf{u}$ is maximized

- Application: line/plane fitting, machine learning, face recognition
- PCA for Total Least Squares: minimum projection on $\mathbf{y}$ means maximum projection on $\mathbf{x}$, so $\mathbf{y}$ is the normal vector

9. Page Rank

- Connectivity Matrix(row:in/column:out)$\Rightarrow$ Transition Probability Matrix
  $A = pA_1 + (1-p)A_2$ A_1:follow with the link A_2:jump to random page
  PageRank: z for initial state, $Az, A^2z...A^kz$,until $A^{k+1}z = A^kz$
- Compute Pagerank: eigenvector method/powermethod(for large n)
- Eigenvalue and Characteristic
  1. Row,Column sum with eigenvalue and eigenvector
  2. Eigenvalue Decomposition $A^k = VD^kV^{-1}$
- Application: Team ranking, university rank,recommendation

10. Audio Basic

- Parameter of recording audio file: sampling rate/Bit solution(8/16bits)/no. of channel
- Time domain vs Frequency domain

	Time	Frequency
features	Period,Intensity,Timbre(Waveform)	Energy,Pitch,Timbre

11. Optimization

- Gradient Descent: $x_{next} = x_{now} - \eta\nabla f(x_{now})$ η: learning rate
  * not guarantee for global optimum
- DSS

12. Dynamic Programming

An effective method for finding optimum solution to a multi-stage decision problem.

1. Characteristic: Decomposition/Subproblem optimality
2. Steps: Optimum-value function $\rightarrow$ Recurrent formula $\rightarrow$ Answer
3. Applications: LCS,Edit distance, matrix chain products,hidden Markov model...

4. Example :QBSH

   Find the alignment between the pitch vector of human's singing and the note vector of a given melody, both in the unit of semitone or MIDI number.

$$dist(p,q) = \min_{u_1,u_2,\ldots u_m} \sum_{i=1}^{m} |p(i) - q(u_i)|,$$

   subject to $u_1 = 1$ and $u_1 \leq u_2 \leq u_3 \cdots \leq u_m$.