2 hours and 30 mins

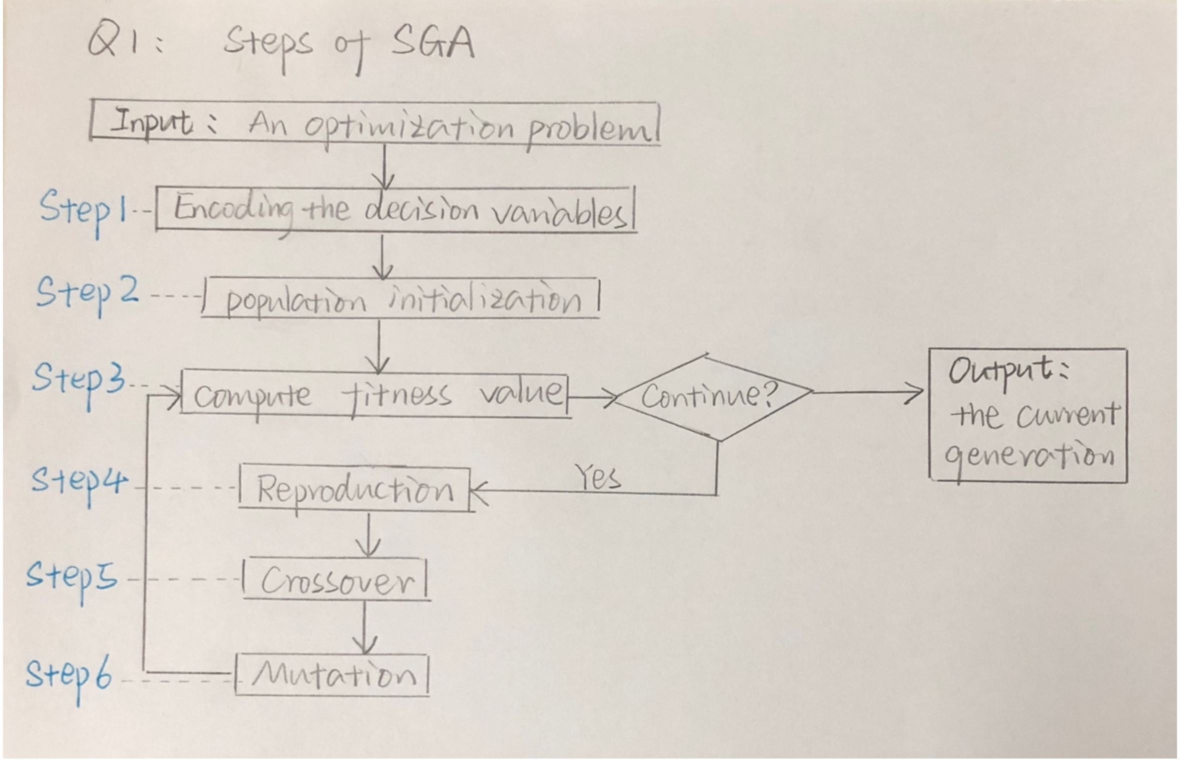
Course Code/Title: ______________________________

1. **Draw a flowchart showing the steps of the "simple genetic algorithm" (SGA) and describe each step. Where appropriate, you may make use of examples to illustrate your answers.**

(20 Marks)

Answer:

The flowchart is shown in Figure. 1.


Q1: Steps of SGA
Input: An optimization problem
Step 1
Encoding the decision variables
Step 2
population initialization
Step3
Compute fitness value
Continue?
Output: the current generation
Yes
Step4
Reproduction
Step5
Crossover
Step6
Mutation

Assume that we want to solve an optimization problem:

$$\text{min. } f(x) = x + 1$$
$$\text{s.t. } 0 \leq x \leq 31$$

Step 1: Decision variable encoding. For this problem we use 5 bits to represent the 32 real numbers.

Step 2: Population initialization. We randomly generate 4 solutions:

$$x_1 = 10011(19), x_w = 01000(8), x_3 = 11000(24), x_4 = 01101(13)$$

Step 3: We now calculate the fitness value and the proportion of each generated solutions, i.e.,

$$f(x_1) = 20, f(x_2) = 9, f(x_3) = 25, f(x_4) = 14$$

Step 4: Reproduction. Use roulette wheel selection to obtain 4 candidates and put them into mating pool. For example,

Step 5: Crossover. Select n/2 pairs of parent strings from mating pool and use one-point crossover with crossover site k. k is randomly generated between (1,L-1), where L is the length of string. Choose crossover probability P_c. For each pair, randomly generated a number, if this number is smaller than P_c, then we apply one-point crossover and compute 2 offsprings; otherwise we use the parent strings as offsprings.

Step 6: Mutation. Chose mutation probability P_m. For each offspring obtained from step 5, randomly generated a number, if this number is smaller than P_m, then we apply mutation, i.e., change the bit 1 to 0, and 0 to 1, respectively.

After the 6 steps, we calculate the fitness value of each string again. If the value is satisfying the stopping criterion, then we output these strings as the final generation. The stopping criterion can be:
1) reached a pre-defined generation number;
2) reached a suitable fitness value;
3) no noticeable change in the last few generations.

2. (a) The following expressions are extracted from the differential evolution algorithm:

Expression 1: $\boldsymbol{V_{i,G} = X_{r1,G} + F(X_{r2,G} - X_{r3,G})}$

Expression 2: $\boldsymbol{u_{j,i,G} = \begin{cases} v_{j,i,G}, & if\ rand_{i,j}[0,1) \leq Cr\ or\ j = j_{rand}, \\ x_{j,i,G}, & otherwise. \end{cases}}$

Expression 3: $$\boldsymbol{X_{i,G+1}} = \begin{cases} \boldsymbol{U_{i,G}}, & \textbf{if } \boldsymbol{f(X_{i,G}) \leq f(U_{i,G})}, \\ \boldsymbol{X_{i,G}}, & \textbf{if } \boldsymbol{f(X_{i,G}) < f(U_{i,G})}. \end{cases}$$

Describe clearly the role of each expression and all the symbols in the expressions.

(14 Marks)

(a) Answer:

Expression 1 is the formula of calculating the donor vector $V_{i,G}$ in differential evolution. For each population member initialized in a differential evolution problem, we randomly select other 3 members (e.g., $X_{r1,G}$, $X_{r2,G}$, and $X_{r3,G}$ here). The donor vector $V_{i,G}$ is defined as the first selected member ($X_{r1,G}$) plus the difference of the last two members ($X_{r2,G}$ and $X_{r3,G}$) scaled by a scaling factor F. F is a pre-defined constant with the value in range (0, 1.5).

Expression 2 is the uniform crossover operation in differential evolution. After we constructed the donor vector, we now are going to implement the crossover operation. First, we need to choose the crossover probability C_r. Then for each member, we randomly generate a number between 0 and 1, if its value is smaller than crossover probability C_r, assign $v_{j,i,G}$ to the trial vector $u_{j,i,G}$; otherwise, assign $x_{j,i,G}$ to the trial vector $u_{j,i,G}$.

Expression 3 is the selection operation in differential evolution. Here we follow the "Darwin's theory of natural selection ", i.e., the fitter value will survival. After forming all the trial vectors, we will need to calculate their fitness values and compare them with the fitness values of the original ones. For a maximization problem, if $f(X_{i,G}) \leq f(U_{i,G})$, then we select $U_{i,G}$ to the next generation, i.e., $X_{i,G+1} = U_{i,G}$; otherwise, we keep $X_{i,G}$ to the next generation, i.e., $X_{i,G+1} = X_{i,G}$.

(b) Estimate the lower bound for the survival probability of the schema 1 * * 0 0 * * * under uniform crossover. Generalise your result to schemas with defining length $\boldsymbol{\delta}$ and order $\boldsymbol{o}$. (Do not attempt to give an exact answer.)

(6 Marks)

(b) Answer:

The defining length is $O(\delta) = 5 - 1 = 4$.
The order is $O(H) = 3$.

3. (a) In the context of multimodal optimization, describe the following niche formation strategies:

(i) Crowding

(ii) Restricted Tournament Selection

(10 Marks)

(a) Answer:

(i) Crowding: The crowding strategy is proposed by De Jong. For a multimodal optimization problem, after generating each individual in the population, we need to calculate the similarity of this individual to other individuals based on a distance metric. The distance metric could be Euclidean distance, Manhattan distance, and so on. Then the newly generated individuals can be replaced by similar individuals in the population. For example, two parent strings are randomly selected and further processed by the reproduction and mutation operations and generate two offsprings. These two offsprings can replace their near (have small value of distance metric) parents if they have more suitable fitness values.

(ii) Restricted Tournament Selection: The difference between the crowding strategy and the restricted tournament selection is that the population of choosing similar individuals of the restricted tournament selection is limited by a window with size w. With these w individuals, we pick the nearest ones to the offsprings and restricts the competition with the similar individuals. The complexity of using window is $O(NPw)$, while the complexity of the crowding strategy is $O(NP)$.

(b) Describe the following with respect to a general multi-objective optimization scenario:

(i) Constrained Domination

(ii) Pareto Optimality

(10 Marks)

(b) Answer:

(i) Constrained Domination: For a multi-objective optimization problem with constraints, we state that constraint variable $x^{(1)}$ dominates another constraint variable $x^{(2)}$ if the following conditions are met:

1) For all the objectives in our multi-objective optimization problem, $x^{(1)}$ is no worse than $x^{(2)}$.

2) In at least one objective in our multi-objective optimization problem, $x^{(1)}$ is strictly better than $x^{(2)}$.

(ii) Pareto Optimality: First, we need to define the non-dominated solutions, which are a set of solutions P' that are not dominated by any member of a set of solutions P. Assume that S is the set of all feasible solutions of our multi-objective optimization problem, the Pareto optimality is that when $P = S$, the resulting set of solutions P' contains all the Pareto optimal solutions.

4. (a) **Describe the affinity propagation clustering algorithm with details.**

(13 Marks)

(b) **Identify three tunable parameters in the orthogonal random forest and briefly explain their roles.**

(7 Marks)

(a) Answer:

The affinity propagation clustering algorithm is a centroid-based clustering algorithm which is with no need of defining specification of the number of clusters as a priori. Five parameters are defined in affinity propagation clustering, i.e.,:

1) Exemplar: defining the center of cluster.

2) Similarity $s(i,k)$: defining how well the data sample with index k is suitable of being an exemplar.

3) Preference p: defining the number of exemplar (i.e., the cluster).

4) Responsibility $r(i,k)$: sent from data sample i to exemplar k, indicate how well-suited exemplar k is to serve data sample i.

5) Availability $a(i,k)$: sent from exemplar k to data sample i, indicate how appropriate for data sample i is to choose exemplar k as its exemplar.

The affinity propagation clustering algorithm follows four steps:

Step1: Compute the similarity matrix and preference p: similarity $s(i,k)$ can be set to a negative squared error (Euclidean distance), i.e.,

$$s(i,k) = -||x_i - x_k||^2$$

and p can be set as the median or the minimum of the input similarities.

Step2: Calculate and update responsibility matrix: availability $a(i,k)$ is initialized to 0. Then the responsibility $r(i,k)$ can be computed as follows:

$$r(i,k) = s(i,k) - \max_{k' \text{s.t. } k' \neq k} \{a(i,k') + s(i,k')\}$$

Step3: Calculate and update availability matrix: availability $a(i,k)$ can be computed as follows:

$$a(i,k) = \min\{0, r(k,k) + \sum_{i' \text{s.t. } i' \notin \{i,k\}} \max\{0, r(i',k)\}\}$$

Step4: Choose the Exemplar: for data sample i, the value of exemplar k that maximizes $a(i,k) + r(i,k)$ either identifies data sample i as an exemplar if k=i, or identifies the data sample that is the exemplar for i. When updating, $r(i,k)$ and $a(i,k)$ are damped to avoid numerical oscillations, i.e.,:

$$r(i,k) = (1-\lambda)r(i,k) + \lambda r(i,k)', \; a(i,k) = (1-\lambda)a(i,k) + \lambda a(i,k)'$$

where $r(i,k)'$ and $a(i,k)'$ are previous values, $r(i,k)$ and $a(i,k)$ are current. The final exemplars are those who satisfy $a(i,i) + r(i,i) > 0$.

(b) Answer:

The three tunable parameters in the orthogonal random forest are:

1) The ensemble size L, which indicates the number of trees in the forest.

2) The scalar m, which indicates the number of features randomly selected to split in at each non-leaf node.

3) The parameter $minleaf$, which indicates the maximum number of samples in an impure terminal node.

5. (a) Sketch the ensemble deep random vector functional link neural network showing the data flow. Define the important components and explain the operations in the context of supervised classification.

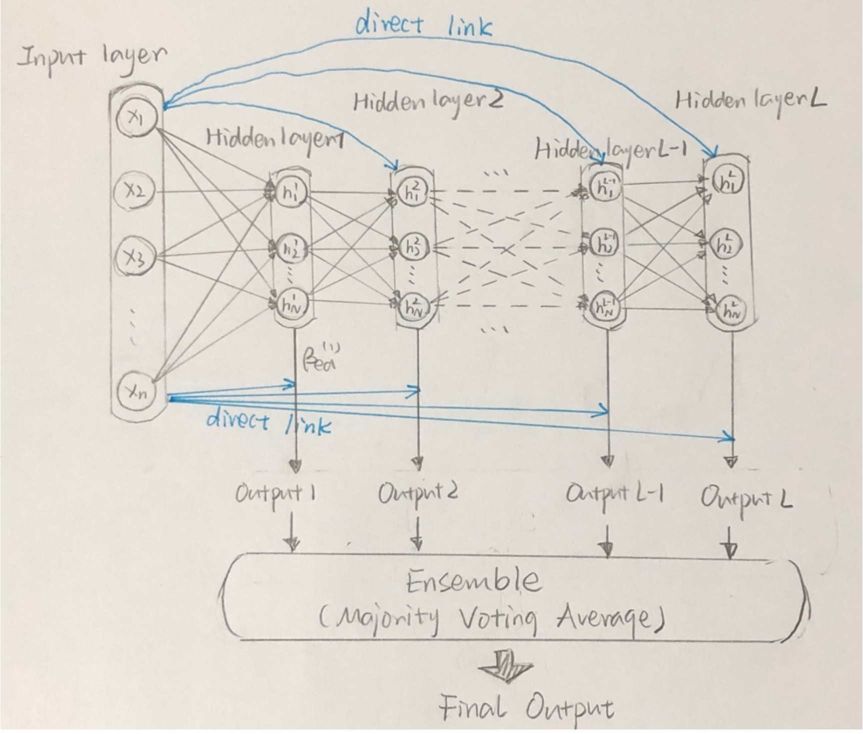
(13 Marks)

(b) Starting from linear ridge regression, derive the kernel ridge regression solution showing the steps clearly.

(7 Marks)

(a) Answer:

The architecture of the ensemble deep random vector functional link neural network is shown in Figure. 2.



The learning objective in a shallow RVFL is to minimize $||D\beta - Y||^2 + \lambda||\beta||^2$, where Y is the ground-truth label, β is the learning parameter, λ is the regularization parameter. $D = \left[H^{(1)}, H^{(2)}, \dots, H^{(L)}, X\right]$ contain all the fixed weight parameters.
In ensemble deep RVFL, several hidden layers are stacked, hence the outputs of the hidden layers are changed into the following forms:
1) 1st hidden layer: $H^{(1)} = g(X * W^{(1)})$.
2) 2nd to the L-th hidden layers: $H^{(L)} = g\left(\left[H^{(L-1)}, X\right] * W^{(L)}\right)$.

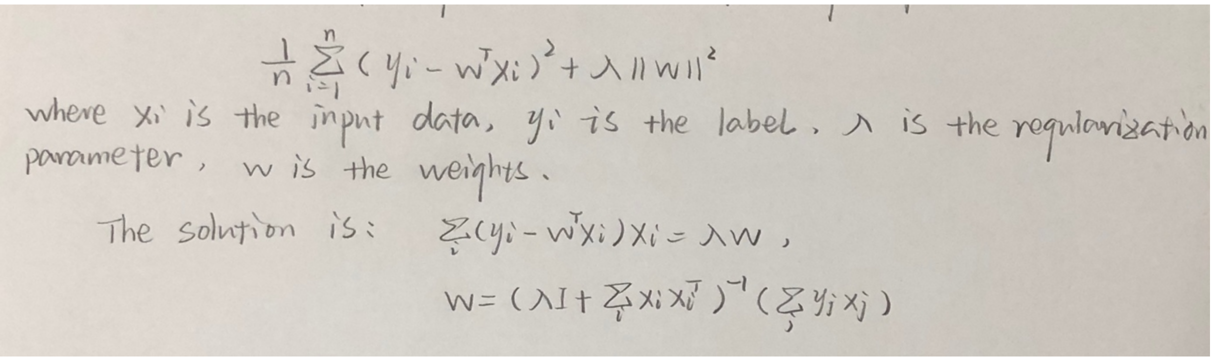
With the ensemble strategy, the learning parameter β is separated into the following:

$$\beta \rightarrow \beta_{d1}, \beta_{d2}, \dots, \beta_{dL}$$

which can be learnt indecently by the primal or dual solutions as follows:
1) Prime space: $\beta = (\lambda I + D^{\mathrm{T}}D)^{-1}D^{\mathrm{T}}Y$ (the number of training samples > the number of total features).
2) Dual space: $\beta = D^{\mathrm{T}}(\lambda I + D^{\mathrm{T}}D)^{-1}Y$ (the number of training samples < the number of total features).

For the ensemble layer, the final output is obtained by using the majority voting average. Each high-level sub-model is fed with original input samples and output the final result.

(b) Answer:
The linear ridge regression has the following expression:



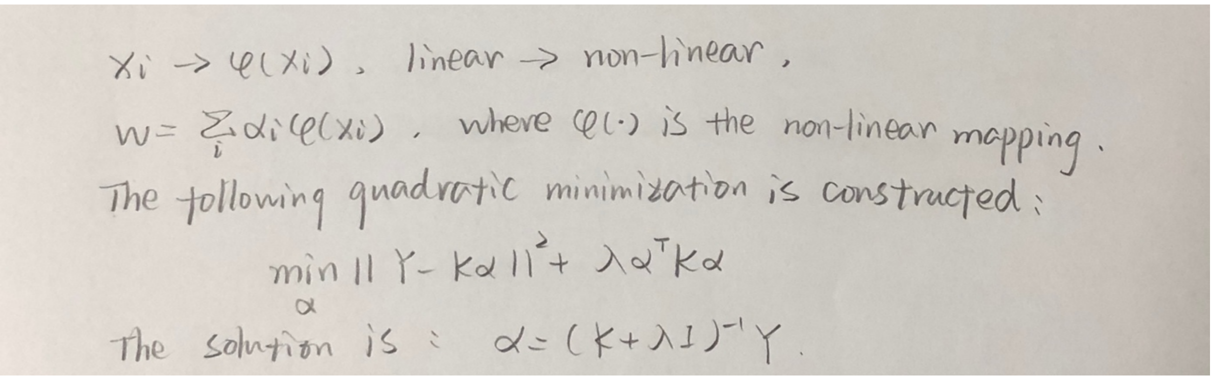
$$\frac{1}{n}\sum_{i=1}^{n}(y_i - w^T x_i)^2 + \lambda ||w||^2$$

where x_i is the input data, y_i is the label. λ is the regularization parameter, w is the weights.

The solution is: $\sum_i (y_i - w^T x_i) x_i = \lambda w$,

$$w = (\lambda I + \sum_i x_i x_i^T)^{-1} (\sum_j y_j x_j)$$

By using the kernel trick, the weight is changed into:



$x_i \to \varphi(x_i)$, linear $\to$ non-linear,

$w = \sum_i \alpha_i \varphi(x_i)$, where $\varphi(\cdot)$ is the non-linear mapping.

The following quadratic minimization is constructed:

$$\min_{\alpha} \| Y - K\alpha \|^2 + \lambda \alpha^T K \alpha$$

The solution is: $\alpha = (K + \lambda I)^{-1} Y$.

END OF PAPER