Programming test answer sheet

Name: Tianyu Xia Application No.: C2108498 Date of Birth: 28-AUG-1999

1. Operations for the right sum

I cannot work out this problem.

2. Equivalent networks

2.1 Background on multi-layer perceptron

2.2 Question: Formulating equivalent networks

Network1:

$\vec{a}^{(1)}=W^{(1)}\vec{a}^{(0)}+\vec{b}^{(1)}$

$\vec{a}^{(2)}=W^{(2)}\vec{a}^{(1)}+\vec{b}^{(2)}$

$\vec{a}^{(3)}=W^{(3)}\vec{a}^{(2)}+\vec{b}^{(3)}$

Input=$\vec{a}^{(0)}$, output=$\vec{a}^{(3)}$

Network2:

$\vec{a}^{(3)}=\widetilde{W}\vec{a}^{(0)}+\tilde{b}$

Input=$\vec{a}^{(0)}$, output=$\vec{a}^{(3)}$

$\therefore$

$\vec{a}^{(3)}$

$= W^{(3)}\vec{a}^{(2)} + \vec{b}^{(3)}$

$= W^{(3)}(W^{(2)}\vec{a}^{(1)}+\vec{b}^{(2)}) + \vec{b}^{(3)}$

$= W^{(3)}W^{(2)}\vec{a}^{(1)} + W^{(3)}\vec{b}^{(2)} + \vec{b}^{(3)}$

$= W^{(3)}W^{(2)}(W^{(1)}\vec{a}^{(0)}+\vec{b}^{(1)}) + W^{(3)}\vec{b}^{(2)} + \vec{b}^{(3)}$

$= W^{(3)}W^{(2)}W^{(1)}\vec{a}^{(0)} + (W^{(3)}W^{(2)}\vec{b}^{(1)} + W^{(3)}\vec{b}^{(2)} + \vec{b}^{(3)}) = \widetilde{W}\vec{a}^{(0)}+\tilde{b}$

$\therefore$

$\widetilde{W} = W^{(3)}W^{(2)}W^{(1)}, \quad \tilde{b} = W^{(3)}W^{(2)}\vec{b}^{(1)} + W^{(3)}\vec{b}^{(2)} + \vec{b}^{(3)}$

3. Multilayer perceptron for regression

I coded in Python and used the *MLPRegressor* function in *sklearn.neural_network* module to constructed the MLP model with 4×4 hidden layers.

(See *code_q3.py*)

4. Connected components

I coded in Python and used the *label* function in *skimage.measure* module. The usage of *label* function is *skimage.measure.label(input, background=None, return_num=False, connectivity=None)*, where *input* means the image to label whose type should be ndarray of dtype int, and *connectivity* means the maximum number of orthogonal hops to consider a pixel/voxel as a neighbor[1].

(See *code_q4.py*)

5. Coloring

I cannot work out this problem.

6. Points inside/outside polygon

I used the ray method[2]: start with the point, make a ray to the right (or left) horizontal direction, and calculate the number of intersection points between the ray and each side of the polygon. if the number of intersection points is odd, the point is inside the polygon; while if the number is even, the point is outside the polygon.

(See *code_q6.py*)

7. Coordinates-to-index & Index-to-coordinates

7.1 2-dimension

a) Derive a mathematical equation converting coordinates to index and derive the inverse equations converting index into coordinates in 2-dimensional grid.

① coordinates→index:

---

[1] Module: measure — skimage v0.19.0.dev0 docs (scikit-image.org)

[2] 判断点是否在多边形内的 Python 实现及小应用（射线法） - 简书 (jianshu.com)

$I = X_2 * L_1 + X_1$

② index→coordinates:

$X_1 = I \% L_1$
$X_2 = I // L_1$

b) Given 2-dimensional grid with sizes (L1, L2) = (50, 57), write a code by yourself to do this.

① Write a code to convert given coordinates to index

② Write a code to convert given index to coordinates

(See *code_q7_1.py*)

7.2 d-dimension

a) Derive a mathematical equation converting coordinates to index and derive the inverse equations converting index into coordinates in d-dimensional grid.

① coordinates→index:

$I = \sum_{i=1}^{d}(X_i * \prod_{j=1}^{i-1} L_j)$

② index→coordinates:

$X_d = I // (\prod_{i=1}^{d} L_i)$
$X_{d-1} = (I \% (\prod_{i=1}^{d} L_i)) // ((\prod_{i=1}^{d-1} L_i)$
…
$X_2 = I \% (\prod_{i=1}^{d} L_i) \% (\prod_{i=1}^{d-1} L_i) \% \ldots // L_1$
$X_1 = I \% (\prod_{i=1}^{d} L_i) \% (\prod_{i=1}^{d-1} L_i) \% \ldots \% L_1$
(The $(X_1, X_2, \ldots, X_{d-1})$ of *Index* are the same like the $(X_1, X_2, \ldots, X_{d-1})$ of *Index* $// (\prod_{i=1}^{d} L_i)$. So I used the recursive method to solve this question.)

b) Given 6-dimensional grid with sizes (L1, L2, L3, L4, L5, L6)=(4, 8, 5, 9, 6, 7), write a code by yourself to do this.

① Write a code to convert given coordinates to index

② Write a code to convert given index to coordinates

(See *code_q7_2.py*)

8. Enzyme Kinetics

8.1 Using the law of mass action, write down four equations for the rate of changes of the four species, E, S, ES, and P.

$V_{ES} = \frac{d[ES]}{dt} = k_1 * [E] * [S]$

$V_E = \frac{d[E]}{dt} = k_2*[ES]$

$V_S = \frac{d[S]}{dt} = k_2*[ES]$

$V_P = \frac{d[P]}{dt} = k_3*[ES]$

8.2 Write a code to numerically solve these four equations using the fourth-order Runge-Kutta method. For this exercise, assume that the initial concentration of E is 1 μM, the initial concentration of S is 10 μM, and the initial concentrations of ES and P are both 0. The rate constants are: k1=100/μM/min, k2=600/min, k3=150/min.

I cannot work out this problem.

8.3 We define the velocity, V, of the enzymatic reaction to be the rate of change of the product P. Plot the velocity V as a function of the concentration of the substrate S. You should find that, when the concentrations of S are small, the velocity V increases approximately linearly. At large concentrations of S, however, the velocity V saturates to a maximum value, Vm. Find this value Vm from your plot.

I cannot work out this problem.

9. Read the following

(https://www.nature.com/articles/s42256-019-0048-x).

Explain in no more than 2 sides of a page (Arial, font size 12, single spacing) whether the Rashomon set is realistic and can be used to meaningfully capture explainable models. You may include references (these do not count to the 2-page limit). Please avoid the use of excessive generics in your response.

The Rashomon set argument mentioned in the article is actually talking about the possibility of accurate interpretable models existing in many different domains. The Rashomon set means a large number of accurate prediction models permitted by the data, in which at least one interpretable model is included. Therefore, the accurate as well as interpretable models usually exist.

Theoretically, such an inference is realistic and feasible, and it is necessary to discover interpretable models from the Rashomon set. Many of the ML models are black boxes that do not explain their predictions in a way that humans can understand, which may lead to many problems such as the lack of transparency and accountability of predictive models. There are some researches about the ‘explainable ML’, which includes a second (post hoc)

model to explain the first black box model. However, the explanation must be wrong and is only an approximation or important features similar to the original model, which limits the trust in the explanation and the black box that it is trying to explain. Besides, the explanation is incomplete and meaningless, and black box model cannot combine the information outside the database. Therefore, interpretable models need to be constructed. The more interpretable machine learning models are, the easier it is for people to understand why certain decisions or predictions are made. With interpretable models, developers can better understand, optimize and adjust the model; explain the internal mechanism of the model to the business side in the operation stage, and explain the results of the model.

However, how to construct a specific and meaningful workflow to find out the interpretable as well as accurate models according to Rashomon set argument is still a problem, which will face many challenges and difficulties. As mentioned in the article, there are three main challenges: how to construct optimal logical models, how to construct optimal sparse scoring systems, and how to define interpretability for specific domains and create methods accordingly. For those inherently interpretable models such as linear regression or decision tree, which can be directly explained by the results, regression weights and so on, it is a problem to capture the features and get the structed data when data is large and complicated, and the accuracy of simple models will be reduced. There are also several model independent methods, such as permutation feature importance, partial dependency plots and so on[3], but the problem is the lack of accurate definition of the interpretability and the corresponding evaluation methods.

In conclusion, I believe the rationality and feasibility of Rashomon set, but how to find out the accurate and interpretable models is a challenge.

[3] https://blog.csdn.net/linxid/article/details/103294973