Midterm 2 Recitation

10-601: Introduction to Machine Learning

11/08/2019

1 Regularization and Optimization

1. **True or false:** For a classification task, as model complexity increases, (ie. more features are engineered), training error can decrease to 0

○ True

○ False

True. Features can always be engineered to perfectly fit on the training data.

2. **Select all that apply:** Which of the following are correct regarding Gradient Descent (GD). Assume data log-likelihood is $L(\theta|X)$ which is a function of the parameter θ and the objective function is negative log-likelihood.

□ GD requires that $L(\theta|X)$ is concave with respect to parameter θ in order to converge

□ GD requires that $L(\theta|X)$ is convex with respect to parameter θ in order to converge

□ GD update rule is $\theta \leftarrow \theta - \alpha \nabla_\theta L(\theta|X)$

□ Given a fixed small learning rate (say $\alpha = 10^{-10}$), GD will always reach the optimum after infinite iterations (assume that the objective function satisfies the convergence condition)

Option A. C is incorrect as $L(\theta|X)$ is the log-likelihood and not negative log-likelihood. D is wrong as GD is not guaranteed to reach optimum even with a small step size.

3. **Select one:** Which of the following is true about the regularization parameter λ (the parameter that controls the extent of regularization)

○ Large values of λ can overfit the data

○ Larger λ does not affect the performance of your hypothesis

○ Adding a regularization term to a classifier, ($\lambda \neq 0$), may cause some training examples to be classified incorrectly

Option C. A is incorrect, larger values of lambda help to generalize predictions or under-fit the data. B is incorrect as it does affect your hypothesis.

2 Classification, Linear Models, Feature Engineering

1. [Fall 2016 Midterm Practice, Problem 5.2.3] For logistic regression, we need to resort to iterative methods such as gradient descent to compute the $\hat{w}$ that maximizes the conditional log likelihood. Why?

There is no closed-form solution.

2. [Fall 2016 Midterm Practice, Problem 5.2.5] For a binary logistic regression model, we predict $y = 1$, when $p(y = 1|x) \geq 0.5$. Show that this is a linear classifier.

We predict $y = 1$ when $p \geq 0.5 \leq 1 - p$. That is, when $\frac{p}{1-p} \geq 1 \Rightarrow \log \frac{p}{1-p} \geq 0$. The LHS is known as log-odds and this is nothing but $\log e^{w^T x} = w^T x$. Therefore, we predict $y = 1$ when $w^T x \geq 0$ and $y = 0$ otherwise. Therefore, logistic regression is a linear classifier.

3. [Spring 2019 Final Practice, Problem 1.3.2] **True or False:** A multi-layer perceptron model with linear activation is equivalent to linear regression model.

○ True

○ False

True

4. [Spring 2018 Final Worksheet, Problem 6.1]We have a set of 2-dimensional data points $\{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, ..., \mathbf{x}^{(n)}\}$ with $\mathbf{x}^{(i)} \in \mathbb{R}^2$ and their corresponding labels $\{\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_n\}$ with $\mathbf{y}_i \in \{0, 1\}$ shown below in figure 1. Let "o" denote 1 and "x" denote 0.

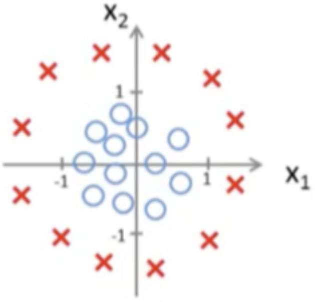

Figure 1: Non-Linear classification boundary

(i) True or False: We can engineer a set of features such that a binary logistic regres- sion model trained on the transformed features perfectly categorizes this dataset. Notice, the unmodified feature space is x(i) $= [1, x_1, x_2]^T$ after folding in the bias term, as introduced in class.

○ True

○ False

True, we can perfectly categorize dataset with feature space such as x = $[x_1^2, x_2^2]^T$.

(ii) If true, please provide the engineered feature space using which a binary logistic regression can perfectly categorize the above dataset. If false, please explain why.

A model such as y = $\sigma(\theta_1 x_1^2 + \theta_2 x_2^2 + \theta_3)$, $\theta_i \in \mathbb{R}$, can perfectly categorize such model.

5. [Spring 2019 Final Worksheet, Problem 1.7] (2 points) Consider the input data as shown in figure 2. (Select all that apply) Which of the following sets of features would you use to apply logistic regression to this data?

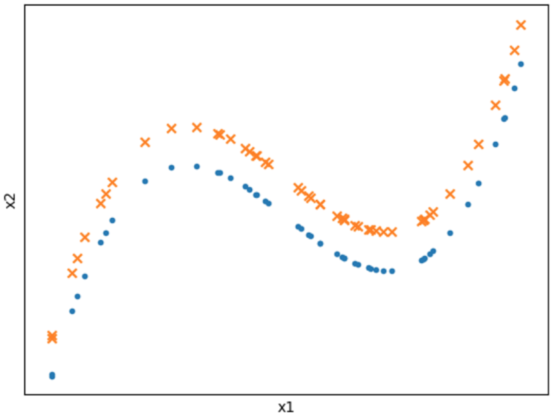

Figure 2: Non-Linear classification boundary

○ x_1, x_2

○ x_1, x_2, x_1^2, x_2^2

○ $x_1, x_2, x_1^2, x_2^2, x_1^3, x_2^3$

○ $x_1, x_2, x_1^2, x_2^2, x_1^3, x_2^3, x_1^4, x_2^4$

C. A and B would undrfit while D would overfit.

6. (1 point) True or False: Logistic regression gives a probability distribution over the output labels for an input example.

○ True

○ False

True

7. (1 point) True or False: Before applying Logistic regression, normalizing the input data i.e., rescaling inputs so that each feature has values between a fixed range [a, b], is necessary for the algorithm to converge even on linearly separable data.

○ True

○ False

False

8. (2 points) Select all that apply. Suppose you train a logistic regression module for binary classification with 2 outputs, use a softmax activation over it and use the cross entropy loss with non-negligible L2 regularization. Which activation function can I use to replace the softmax with and still learn the same decision boundary. Assume the data is linearly separable

□ softmax activation with 1 output unit

□ sigmoid activation with 1 output unit

□ softmax activation with 3 output units

□ sigmoid activation with 2 output units, that is applied indepently to each unit

B

3 Deep Learning

[S19-10601-Worksheet 2, Problem 1.10] Assume you have a Neural Network made up of a hidden layer, with 4 neurons all with a Perceptron non-linearity (return 1 if input is greater than 0 and 0 otherwise) applied and a single output neuron.

(a) Draw the decision boundaries that your neurons may find on the below graph and be sure to show the orientation of these decision boundaries.

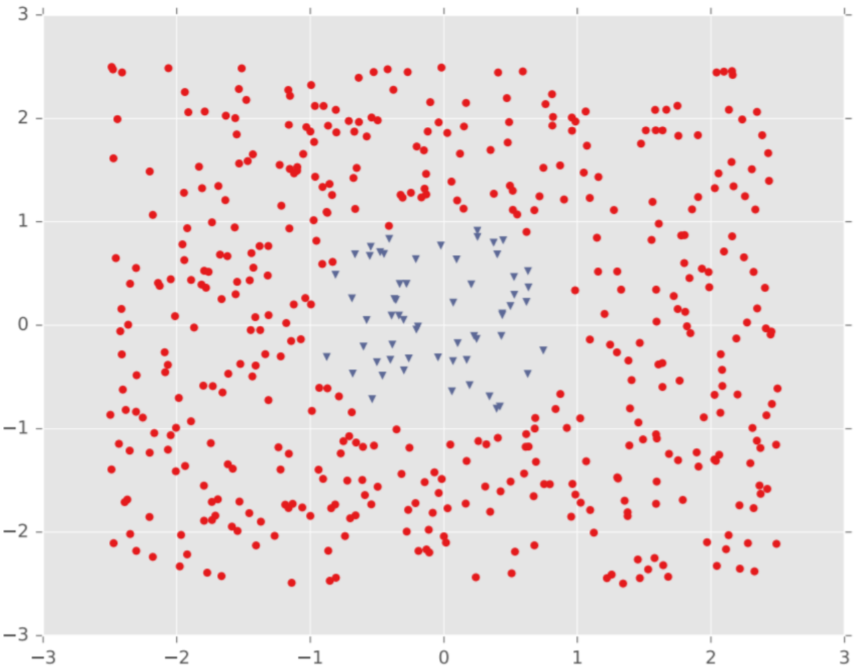

Figure 3: Neural Network Data

(b) Explain what kind of function the output neuron represents.

(c) **True or false:** Gradient descent would be suitable to help us find the optimal solution.

(a)

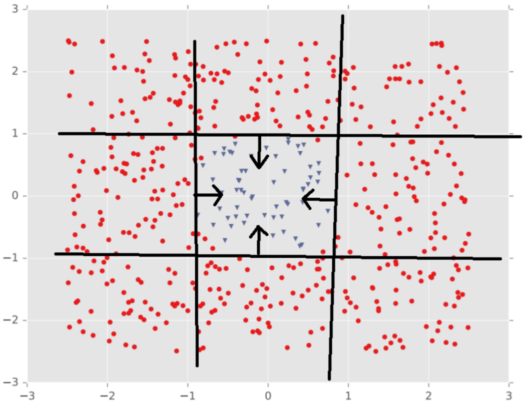

Figure 4: Solution

(b) Output 0 if any of the hidden neurons output 0, and output 1 if all of the hidden neurons output 1.

(c) No, perceptron function isn't differentiable.

4 Reinforcement Learning

1. A maze environment and the rules of a maze are defined as such:

State1	G
State2	H
State3	State4

Table 1: Map of the Maze

- At each valid state (State 1-4), the agent can move North, South, West, and East.
- **Stochastic Environment**

  After the agent selects an action, it has a 50% chance of moving in the intended direction, a 25% chance of moving to the left of the intended direction, and a 25% chance of moving to the right of the intended direction. For example, If an agent is at state 3, and decides to go right. It has a 50% chance of ending up at state 4, a 25% chance of moving North and going to state 2, and a 25% chance of staying in its previous state (by hitting the wall by going south).
- If the agent hits a wall (edge of the maze) or an obstacle(H), it stays in its previous state.

- The agent receives a reward of 100 when entering the goal state (G), 0 otherwise.
- Discount Factor = 1

Please perform two rounds of value iteration, and report the state value function (V) after each round. The initial state values (including G) are 0.

Fill in the tables:

0	0
0	H
0	0

Table 2: Initial state

	H

Table 3: Round 1

	H

Table 4: Round 2

Solution:
ROUND 1: STATE 1: 50, STATE 2: 0, STATE 3: 0, STATE 4: 0, GOAL: 0
ROUND 2: STATE 1: 62.5, STATE 2: 25, STATE 3: 0, STATE 4: 0, GOAL: 0

2. **Short Answer** For MDPs, recall the value function of a state s under a policy π, denoted $v_\pi(s)$, is the expected discounted return when starting in s and following π thereafter:

$$v_\pi(s) = E\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \middle| S_t = s\right]$$

Similarly, the value of taking action a in state s under a policy π, denoted $q_\pi(s,a)$, is the expected discounted return starting from s, taking the action a, and thereafter following policy π:

$$q_\pi(s,a) = E\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \middle| S_t = s, A_t = a\right]$$

Finally note that if the agent is following policy π at time step t, then $\pi(a|s)$ is the probability that $At = a$ if $St = s$, that is we take action a at time step t given we are at state s with probability $\pi(a|s)$. This allows our policy to be stochastic.

Write down the equation for $v_\pi(s)$ in terms of $q_\pi(s,a)$ and $\pi(a|s)$:

$$\begin{aligned} v_\pi(s) &= E\left[\sum_{k=0}^{\infty}\gamma^k R_{t+k+1}\middle| S_t = s\right] \\ &= E\left[\sum_{k=0}^{\infty}\gamma^k R_{t+k+1}\middle| S_t = s, A_t = a\right]\pi(a|s) \\ &= \sum_{a\epsilon A} q_\pi(s,a)\pi(a|s) \end{aligned}$$

5 Information Theory

An experiment consists of simultaneously flipping a fair dime and a fair penny. You are asked to predict whether the penny ends up heads or tails, rank these values. Let P denote the random variable corresponding to the outcome of flipping the penny.

1. The irreducible entropy of P.
2. The cross-entropy from the truth to your prediction when you (rationally) predicted that the penny would come up heads with probability 0.5.
3. The average entropy of your prediction when you are given the total number of "tails" (this could be 0, 1, 2)
4. The average entropy of your prediction when you are given whether the total number of "tails" was odd or even.

This question is adapted from Fall18 10601 section A/C practice midterm.

1. $H(P) = 1$
2. $CH(P,Q) = H(P) = 1$
3. $H(Q|T) = P(T=0)*H(Q|T=0) + P(T=1)*H(Q|T=1) + P(T=2)*H(Q|T=2) = 0.25*H(0,1) + 0.5*H(0.5,0.5) + 0.25*H(0,1) = 0.5$
4. $H(Q|odd(\#T)) = P(T = odd)*H(Q|T = odd) + P(T = even)*H(Q|T = even) = 0.5*H(0.5,0.5) + 0.5*H(0.5,0.5) = 1$. Note P(T=1) is 0.5.

$2 < 1 = 3 = 4$

Working with Languages

Suppose you are working on a language processing task in Japanese. You know that there are 46 basic characters in Japanese (like there are 26 characters in English). In this question, assume log is of base 2.

1. (1 point) Suppose uniform character distribution and independence between successive characters, how many bits do you need to encode a basic Japanese character? Do NOT

simplify the answer.

□

$\log 46$ This is adapted from Roni's tutorial on information theory.

2. (1 point) However, the true distribution of characters is not uniform. In this case, how does the number of bits needed to encode a character change in comparison to your answer in the previous question? Assume all other assumptions remain the same.

   ○ $>$ answer in 1

   ○ $=$ answer in 1

   ○ $<$ answer in 1

   ○ None of the above

   C

3. (1 point) **True or False:** Suppose we have two languages A and B. A has 26 characters and B has 46 characters. Is it true that on average language A needs more bits to encode a word than language B? Assume all the words in both languages are composed by the characters.

   ○ True

   ○ False

   False. Words may have different lengths.