Homework 3 for #70240413
"Statistical Machine Learning"

Instructor: Prof. Jun Zhu

May 24, 2018

1 Dimension Reduction, PCA

1.1 Minimum Error Formulation (1 point)

Complete the proof on the lecture slide (page 34) which shows that PCA can be equivalently formulated as minimizing the mean-squared-error of a low-dimensional approximation from a subset of orthonormal bases.

1.2 MNIST (1.5 points)

Implement the PCA algorithm and run it on the whole MNIST training set[1]. Choose d to preserve 30%, 60%, 90% information respectively. Fix two images of different digits and show the resulting images for each d.

Rerun your PCA implementation without centering the dataset (substracting the sample mean) and show the difference between the results with the same two images you choose before.

You should plot the images in the report and also submit your code alongside in a zip file.

2 Learning Theory

2.1 VC Dimension (1.5 points)

Consider the instance space X to be $\mathbb{R}^2$. Derive the VC dimension of the following hypothesis space (lecture slide page 53):

$H = \{$All the axes-parallel rectangles in $\mathbb{R}^2$, where points inside the rectangle are classified as positive.$\}$

You should first re-state VC ≥ 3, VC ≥ 4 in words (address at least why VC ≥ 4 even though some placement of 4 points can't be shattered), and then address the case with 5 points with a mandatory simple illustrative figure.

[1] `http://yann.lecun.com/exdb/mnist/`, or `http://www.cs.nyu.edu/~roweis/data.html` for a MATLAB version

2.2 Generalization Bound (1.5 points)

Consider a learning problem in which instances $X = \mathbb{R}$ are all the real numbers, and the hypothesis space $H = \{(a < x < b) | a, b \in \mathbb{R}\}$ is composed of all the intervals in $\mathbb{R}$, where all points inside the interval is classified as positive. What is the upper bound of the probability that a hypothesis $h \in H$ consistent with m instances $x_1, \cdots, x_m$ has an error of at least ϵ? You can use the theoretical results from the lecture slides directly.

3 Reinforcement Learning

3.1 Value Iteration (1 point)

Consider the bottom-right table on page 25 (Value Iteration Demonstration) of the RL lecture slides. It already gives you the formula to calculate 27.935 for the first iteration. Please list the formula to calculate the remaining 5 entries in the table.

3.2 Policy Improvement Theorem (1 point)

Prove the Policy Improvement Theorem (on page 28, Policy Improvement Theorem, of the RL lecture slides).

Hint: one way is to show that $V^{\pi_k} \leq V^{\pi_{k+1}}$, where V^{π_k} is the V value for some fixed policy π_k.

3.3 TianShou (2.5 points)

Implement ONE of the following algorithms on a simple but applicable Gym environment with TianShou:

- Deep Q-learning
- Proximal Policy Optimization

Also, state whether it's an on-policy or off-policy, value-based or policy-based algorithm.

You may reference ddpg.py and actor_critic.py under the examples directory of TianShou. You should be able to get fair results on simple environments in tens of minutes just running on CPU. Please submit both your running log file showing improvement and your code in a zip file.