MLSM 2021 - Quiz 2

24/09/2021 - 06/10/2021 at 23:59

Questions based on lecture 2: Statistical learning theory

(1) (1.0 pt.) With a PAC learnable class $\mathcal{C}$, with algorithm $\mathcal{A}$ and sample S, if the generalisation error satisfies $Pr\left(R(h_S) \leq 0.08\right) \geq 0.94$, which of the statements is **false**?

(a) When running the algorithm, it is possible that 6% of the time the learned hypothesis is not good

(b) If we obtain a good hypothesis, it is still possible that up to 8% of the new data points are misclassified

(c) It is not possible to obtain a hypothesis with a worse accuracy than 92%

(2) (1.0 pt.) Consider classification problem on dataset containing 3 binary features, and a binary label. Using a rule-based classifier with boolean conjunctions, the sample complexity bound is

$$m \geq \frac{1}{\epsilon}\left(\log\left(|\mathcal{H}|\right) + \log\left(\frac{1}{\delta}\right)\right).$$

How many data samples are needed in order to guarantee 90% accuracy 95% of time?

(a) 52

(b) 63

(c) 112

Questions based on lecture 3: Learning with infinite hypothesis classes

(3) (1.0 pt.) You are investigating the VC dimension of a classifier, and have found a configuration of 5 points that you can shatter with it. What does this tell you?

(a) VCdim < 5

(b) VCdim $= 5$

(c) VCdim ≥ 5

(4) (1.0 pt.) [*Programming exercise*] The attached example code contains a simple dataset and a binary classifier.

What is the value of the generalisation bound based on Rademacher complexity for training set sizes n=20, n=50 and n=100 (use the first n samples of the training set)? Choose the closest values (randomness from calculating the empirical Rademacher bound can result in small variation in the results). Use $\delta = 0.05$.

(a) $20 \approx 1.00$; $50 \approx 0.61$; $100 \approx 0.42$

(b) $20 \approx 1.20$; $50 \approx 0.81$; $100 \approx 0.50$

(c) $20 \approx 1.52$; $50 \approx 1.24$; $100 \approx 0.99$

(5) (1.0 pt) [*Programming exercise*] Using the same setting as the previous exercise, calculate the value of the generalisation bound based on VC-dimension. The VC-dimension of perceptron is $d + 1$, where d is the number of features.

(a) $20 = 1.21$; $50 = 0.85$; $100 = 0.64$

(b) $20 = 1.29$; $50 = 0.95$; $100 = 0.66$

(c) $20 = 1.41$; $50 = 1.05$; $100 = 0.73$

```python
import numpy as np
import matplotlib.pyplot as plt
from sklearn.datasets import make_blobs

#=====================================================================
# dataset
n_tot = 200
# two blobs, not completely separated
X, y = make_blobs(n_tot, centers=2, cluster_std=3.0, random_state=2)

# divide data into training and testing
#NOTE! Test data is not needed in solving the exercise
# But it can be interesting to investigate how that behaves w.r.t. training set
# performance and the bounds :)
np.random.seed(42)
order = np.random.permutation(n_tot)
train = order[:100]
# test = order[100:]

Xtr = X[train, :]
ytr = y[train]
# Xtst = X[test, :]
# ytst = y[test]

#=====================================================================
# classifier

# The perceptron algorithm will be encountered later in the course
# How exactly it works is not relevant yet, it's enough to just know it's a binary classifier
from sklearn.linear_model import Perceptron as binary_classifier

## It can be used like this:
# bc = binary_classifier()
# bc.fit(Xtr, ytr)  # train the classifier on training data
# preds = bc.predict(Xtst)  # predict with test data

#=====================================================================
# setup for analysing the Rademacher complexity

# consider these sample sizes
print_at_n = [20, 50, 100]  # take always n first samples from training set
delta = 0.05
```