MLSM 2023 - Quiz 2

15/09/2023 - 27/09/2023 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 $Pr\left(R(h_S) \leq 0.2\right) \geq 0.9$, which of the statements is **true**?

- (a) It is not possible to run the algorithm and obtain a hypothesis with only 70% accuracy.
- (b) Always at least 20% of the test samples are wrongly classified.
- (c) It is possible to run the algorithm and obtain a hypothesis with 95% accuracy.
- (d) It is guaranteed that 80% of the time the generalisation error is not greater than 10%

(2) Alice wants to learn which spice combinations work for her in a chili recipe, i.e. if she likes the result or not. She has five spices to try out in different combinations. How many different combinations should she try out in order to be able to build a boolean conjunctions classifier to give 95% accuracy with 0.9 confidence?

Answer: ______ combinations

Questions based on lecture 3: Learning with infinite hypothesis classes

(3) (1.0 pt.) A classifier has a VC dimension of 4. Which of the following claims is true?

- (a) Given any four data points, it is possible to shatter them with the classifier.
- (b) It is not possible to shatter any collection of five data points with the classifier.
- (c) It is not possible to shatter any collection of three data points with the classifier.

(4) (1.0 pt.) [*Programming exercise*] The provided code template and data files contain a simple dataset and a binary classifier. How many training samples are needed for the difference between the test error and the generalisation bound based on Rademacher complexity to drop below 0.4 (for the first time) when $\delta = 0.05$? To train with n samples, use the first n samples of the training set.

Note: randomness from calculating the empirical Rademacher bound can result in small variation in the results; choose the closest one

- (a) About 58-63 samples
- (b) About 77-80 samples
- (c) More than 300 samples

(5) (1.0 pt) [*Programming exercise*] Continuing with the same data, how many training samples are needed for the difference between the test error and the generalisation bound based on VC dimension complexity to drop below 0.4 (for the first time) when $\delta = 0.05$? The VC-dimension of perceptron is $d + 1$, where d is the number of features in the data.

- (a) 100
- (b) 145
- (c) 269

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


# ============================================================================
# dataset

"""
n_tot = 800
n = int(n_tot/2)
# two moons, not really linearly separable
X, y = make_moons(n_tot, noise=0.15, random_state=0)

# divide data into training and testing
np.random.seed(42)
order = np.random.permutation(n_tot)
train = order[:n]
test = order[n:]

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

np.save("quiz2_datafiles/Xtr.npy", Xtr)
np.save("quiz2_datafiles/Xtst.npy", Xtst)
np.save("quiz2_datafiles/ytr.npy", ytr)
np.save("quiz2_datafiles/ytst.npy", ytst)
"""

Xtr = np.load("quiz2_datafiles/Xtr.npy")
Xtst = np.load("quiz2_datafiles/Xtst.npy")
ytr = np.load("quiz2_datafiles/ytr.npy")
ytst = np.load("quiz2_datafiles/ytst.npy")
n = len(ytr)

plt.figure()
colors = ["g", "b"]
for (X, y) in [(Xtr, ytr), (Xtst, ytst)]:
    for ii in range(2):
        class_indices = np.where(y==ii)[0]
        plt.scatter(X[class_indices, 0], X[class_indices, 1], c=colors[ii])
plt.title("full dataset")
```

```python
plt.show()

# ==============================================================================
# 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)  # this is how to train the classifier on training data
preds = bc.predict(Xtst)  # this is how to obtain predictions on test data
```