MLSM 2020 - Quiz 1

08/09/2020 - 16/09/2020 at 23:59

---

(1) (1.0 pt.) Assume that there is a labeling function $f : \mathcal{X} \to \mathcal{Y}$, where probability distribution over $\mathcal{X}$ is denoted by $\mathcal{D}$. The Error of a classifier $h : \mathcal{X} \to \mathcal{Y}$ can be defined as:
$L_{\mathcal{D},f}(h) \overset{\text{def}}{=} \mathbb{P}_{x\sim ?}[h(x) \neq f(x)]$, where in the learner, x is assumed to be a randomly chosen example from:
1- known distribution $\mathcal{D}$
2- unknown distribution $\mathcal{D}$
3- unknown distribution $\mathcal{D}' \neq \mathcal{D}$

(2) (1.0 pt.) What is the probability of having more than 10 noisy examples, in a sample of 35 drawn uniformly from a distribution with 20% inherent noise.
Hint 1: you need to first compute probability of having 10 or less noisy examples in a sample of size 35
Hint 2: you will need to sum up Binomial trials, where each drawn example can be noisy or not
Hint 3: you can compute it by hand, but it could be faster computationally
1- 0.07
2- 0.02
3. 0.10

(3) (1.0 pt.) In each of the following problems which measure should be prioritized for evaluating a classifier.
A) Cancer prediction (positive: cancer, negative: healthy). The patients with positive prediction will go through more analysis, and the negative ones will be sent home. Hint: sending a cancer patient home, can be dangerous.
B) Spam email prediction (positive:spam, negative: non-spam). The spam detected emails would be automatically removed. Hint: we do not want to miss any important email.
$precision = \frac{TP}{TP+FP}$
$recall = \frac{TP}{TP+FN}$
1- A: precision, B: recall
2- A: recall, B: precision
3- A: precision, B: precision

(4) (1.0 pt.) Import the given files *X.csv* and *Y.csv* in the Materials section, as the inputs and targets. The provided files can also be imported from *Boston* dataset available in *sklearn*, where the features *LSTAT* and *RM* are used as inputs and *target* is used as output. Split the dataset to training and test set using the following Python code:

```python
from sklearn.model_selection import train_test_split
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size = 0.3, random_state=1)
```

function description

using Linear Regression (LR) without considering bias (intercept), which option is the root-mean-square error (RMSE) over test data:
1- 5.26
2- 6.34
3- 4.62

(5) (1.0 pt.) In the previous question, again using LR without bias, compute average (K-folds cross-validation error) and variance of the root-mean-square error (RMSE) on test folds, for $K = 2$ and $K = 5$. To generate the folds, use the following Python code:

```
from sklearn.model_selection import KFold
KFold(n_splits=k, random_state=1, shuffle=True)
```

function description

1- K=2 : average = 6.23, variance = 0.21 / k=5: average = 6.12 , variance = 0.64

2- K=2 : average = 5.38 , variance = 0.43 / k=5: average = 5.53, variance = 0.38

3- K=2 : average = 6.03 , variance = 0.37 / k=5: average = 5.72 , variance = 0.50