Aashsih Kumar pcr902

December 17, 2019

Contents

1 Airline Revisited **1**

2 The Growth Function and the VC-Dimension **2**

3 SVMs **2**
3.1 Data Normalization 2
3.2 Model Selection using Grid Search 3
3.3 Inspecting the Kernel Expansion 3

1 Airline Revisited

Let's consider the following events Event A: In the sample of 10000 passengers, where each passenger shows up with probability p, we observe 95 percent shows up.

$$A = \sum_{i=1}^{10000} X_i = 9500 \tag{1}$$

Event B : In the sample of 100 passengers, where each passenger shows up

$$B = \sum_{i=1}^{100} Y_i = 100 \tag{2}$$

Event C : In the sample of 10100 passengers, where 9600 passenger shows up

$$C = \sum_{i=1}^{10100} Z_i = 9600 \tag{3}$$

We need to bound the probability of Event A and Event B happening given that Event C happens, i.e $P(A \cap B|C)$

$$P(A \cap B|C) \leq P(B|C) \tag{4}$$

so, we can get that probability using sampling without replacement.

$$P(B|C) = \frac{9600}{10100} * \frac{9599}{10099} * ..... * \frac{9501}{10001} \tag{5}$$

$$P(A \cap B|C) \leq \frac{9600}{10100} * \frac{9599}{10099} * ..... * \frac{9501}{10001} \tag{6}$$

By using this we get a bound of 0.0061

2 The Growth Function and the VC-Dimension

not attempted

3 SVMs

3.1 Data Normalization

Mean of training Data set:
[1.55960388e+02, 2.04821194e+02 , 1.15058622e+02, 5.99785714e-03, 4.28877551e-05 , 3.20418367e-03 , 3.31540816e-03, 9.61295918e-03, 2.77400000e-02 , 2.62408163e-01 , 1.46761224e-02, 1.66144898e-02, 2.19880612e-02 , 4.40281633e-02 , 2.26390816e-02 , 2.20007041e+01, 4.94819602e-01 , 7.15689765e-01, -5.76372753e+00 , 2.14795724e-01, 2.36576287e+00 , 1.99708816e-01]

Variance of training Data set:
[1.96280920e+03, 9.63381571e+03 , 2.09357394e+03, 1.56323454e-05, 9.05262911e-10 , 5.59068556e-06, 5.17943912e-06 , 5.03013168e-05, 2.52776890e-04, 2.64697314e-02, 7.48602666e-05, 1.02539466e-04, 1.76683701e-04, 6.73690070e-04, 8.86782604e-04 , 1.65097280e+0 1.03107075e-02, 3.11595818e-03, 1.06174931e+00 , 5.74533305e-03, 1.36459832e-01, 6.65889223e-03]

For normalization, I have used the following linear mapping for feature X, where mean(X), std(X) are mean and standard deviation of the feature X.

$$f(x) = \frac{x - mean(X)}{std(X)} \tag{7}$$

After doing normalization, we get the variance 1 and mean close to 0 for all the features.
Mean of transformed train Data set:
[1.84659544e-16 , -6.06091141e-16, -2.44702218e-16, 1.33679915e-16, -1.28242088e-15 , -3.29668265e-16 , 2.83220159e-16, -1.13288064e-17, 1.29374969e-15 , -7.79421879e-16 , 1.23597278e-15, -4.70145465e-16, 6.41210441e-16, -1.40477199e-16, 1.53505326e-16, -2.27652364e-15, 1.19179043e-15, 3.16257791e-15, -1.57640341e-15 , 7.43169698e-16, -1.43705909e-15 , -3.76116372e-16]

Variance of transformed train Data set:
[1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1. , 1., 1. , 1., 1., 1., 1., 1. , 1., 1. , 1. , 1.]

Mean of transformed test Data set:
[-0.07857931 , -0.15804162, 0.05562311 , 0.11318387 , 0.07157377 , 0.08691489, 0.11567239, 0.08701553 , 0.24898214 , 0.24518734 , 0.2295662 , 0.25089051, 0.31660826, 0.22960283 , 0.14905702, -0.05676346, 0.07356766, 0.08676698, 0.15477245, 0.31069455 , 0.08741643, 0.1685766]

Variance of transformed test Data set:
[0.73218508, 0.71491336, 0.79759033, 1.99040214, 1.66604029, 2.13673681, 1.92226228 , 2.13767335, 1.77195651, 1.82895633, 1.7173149 , 1.77783879, 2.19022855, 1.7174543 , 2.66297002, 1.36090146, 1.08263293, 0.95130846, 1.21651005, 1.36280271 , 1.13351689, 1.41470112]

3.2 Model Selection using Grid Search

I have used scikit-learn library as it has a built in implementation for GridSearchCV. It takes an estimator SVC(), a dictionary of parameters like C, gamma and kernel and creates a new estimator. Then we call fit for fitting the model for grid search. It runs in loop with cross validation to find the best combination of parameters. Then, it runs without cross-validation with the best parameter and build a single model. then we can call prediction using our model.
we passed 'C': [7,8,9,10,11,12,13], 'gamma': [0.001, 0.01, 0.1, 1, 10, 100,1000], 'kernel': ['rbf'] we have used radial basis function kernel.
Following are the best parameters for C and gamma
C: 7, gamma: 0.1

Training error = 0.0
Test error = 0.0928

3.3 Inspecting the Kernel Expansion

not attempted