FOML Assign 2

Gulshan Hatzade

September 2024

1 Question 1

The margin boundaries in standard support vector machine foumulation are $w \cdot x + b = +1$, $w \cdot x + b = -1$.

The above boundaries contain the points on decision boundary where 2 classes will be placed by SVM.

Distance between two boundaries called as margin is $2/||w||$.
When we consider modified boundaries where in place of +1 and -1 , we are dealing with $+\gamma$ and $-\gamma$ then boundries will be defined as,

$$w \cdot x + b = +\gamma \quad and \quad w \cdot x + b = -\gamma$$

So, then margin will be $2\gamma/||w||$, this margin looks different as compared to above margin.

Objective of support vector machine is maximizing margin but scaling of w and b is arbitrary. By scaling w and b by then equations will be transformed back into their original form.

$$\frac{w}{\gamma} \cdot x + \frac{b}{\gamma} = +1 \quad and \quad \frac{w}{\gamma} \cdot x + \frac{b}{\gamma} = -1$$

From above eqation, we can observe that as the optimization problem is invariant under such scaling so the maximum margin hyperplane remains unchanged.

So, we can conclude that the solution of support vector machine is not affected by scaling margin boundaries by as the optimization and problem formulation will be adapted accordingly by the rescaling of b and w.

2 Question 2

Our task is to prove that half margin of the max margin indicated by which is equal to $1/||w||$ relates to sum of lagrange multipliers α_i by ,

$$\frac{1}{\rho^2} = \sum_{i=1}^{N} \alpha_i$$

The optimal weight vector w is expressed as a linear combination of support vectors in due formulation of support vector machine,

$$w = \sum_{i=1}^{N} \alpha_i y_i x_i$$

Here, $y_i \in \{-1, +1\}$ which are the class labels,
α_i are Lagrange multipliers and
x_i corresponding feature vectors.

Calculating square of $||w||$: For showing the relationship between $||w||$with the Lagrange multiplier, the norm $||w||^2$ need to be calculated.

$$||w||^2 = w \cdot w = \left(\sum_{i=1}^{N} \alpha_i y_i x_i \right) \cdot \left(\sum_{i=1}^{N} \alpha_i y_i x_i \right)$$

Expanding the above equation,

$$||w||^2 = \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j (x_i \cdot x_j)$$

By performing inner product of feature vectors which is made by kernel function in non linear support vector machine.

We can write linear svm for simplicity as follows,

$$||w||^2 = \sum_{i=1}^{N} \alpha_i 1.1$$

This is equation (1.1)

For defining support vectors Langrange multipliers are tied to constant in dual formulation.For non support vectors , lagrange multipliers α_i are zero. Only support vectors are contributing to norm $||w||$.

Half margin for support vector machine is given as follows,

$$\rho = \frac{1}{||w||}$$

Now, square both sides,

$$\rho^2 = \frac{1}{||w||^2}$$

From equation (1.1),

$$\frac{1}{\rho^2} = \sum_{i=1}^{N} \alpha_i$$

This is required relationship between half margin and lagrange mltipliers.

3 Question 3

3.1 Solution a)

It is a Valid kernel.
Lets assume 2 PSD matrices K1 and K2 wich are associated with kernels k1 k2, if v is vector,

$$v^T(K_1 + K_2)v = v^T K_2 v + v^T K_1 v$$

As K_2 and K_1 are PSD:

$$0 \leq v^T K_2 v \quad and \quad 0 \leq v^T K_1 v$$

So we can conclude,

$$0 \leq v^T(K_1 + K_2)v$$

Hence, $K_1 + K_2$ is PSD . Sum of 2 PSD is PSD hence sum of 2 valid kernels is also valid kernel. which shows the given kernelk_1 and k_2 are valid kernel.

3.2 Solution b)

It is a valid kernel.
Let K_1 and K_2 are matrices for 2 valid kernels, so product of K_1 and K_2 which is element wise multiplicationi s PSD.

$$v^T(K_1 \circ K_2)v = \sum_{i,j}(K_{1,ij}K_{2,ij})v_i v_j$$

The hadamard product is also PSD for K_1 and K_2 matrices,

$$v^T(K_1 \circ K_2)v \geq 0$$

If we take product of 2 kernels which are valid then result is also a valid kernel. ELement wiese product of 2 PSD matrices is also PSD is concluded by the Schur product theorem. Hence product of the 2 kernel which are valid kernels, is also a valid kernel.

3.3 Solution c)

It is a valid kernel.
PSD property must be satisfied by positive coeffivents to values of k1 when applying a polynomial function h till the point polynomial is increasing on non -ve points it is non negative.
For proof, we will assume the kernel k1 and K1 is its gram matrix. We need to prove h(K1) is PSD. For increasing function and which have non negative coeficiants of polynomial function h(t), we can say if t is PSD then h(t) is also PSD.
Till the point the h maintains positive definiteness for every inputs, the obtained matrix h(K1) should be PSD, hence the given kernel is a valid kernel.

3.4 Solution d)

It is not valid kernel.
For kernel k1(x, z) which is constant function, matrix K1 corresponding to that kernel is suppose is PSD.
Depending upon in what way exponential function affects the elements in matrix, , applying exponential will not guarantee PSD. THe given kernel may not guaranteed and can't be surely said that it is a valid kernel. So, when we apply kernel, the exponential function is not satiesfiying the PSD property.
Hence it is not sure for preserving the property os PSD.

3.5 Solution e)

If a valid kernel.
By observing this, we can say that this is RBF (radial basis function) kernel. Consider K is gram matrix corresponding to RBF (radial basis function) which will always be PSD. If v is any vectors then,

$$\mathbf{v}^T\mathbf{K}\mathbf{v} = \sum_{i,j} \exp\left(\frac{-\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right) v_i v_j$$

Due to properties of Gaussian function, the above matrix is surely a PSD. Hence the given kernel is a valid kernel.

4 Question 4

4.1 Solution 4)a

Accuracy over the entire dataset is 0.9787735849056604 and no. of Support Vectors are 28.

4.2 Solution 4)b

If the model trained using using first 50 samples in linear kernel-
Training with first 50 samples-
Sample Size is 50
Accuracy is 0.9811320754716981
No. of Support Vectors are 2

If the model trained using using first 100 samples in linear kernel-
Training with first 100 samples-
Sample Size is 100
Accuracy is 0.9811320754716981
No. of Support Vectors are 4

If the model trained using using first 200 samples in linear kernel-
Training with first 200 samples-
Sample Size is 200
Accuracy is 0.9811320754716981
No. of Support Vectors are 8

If the model trained using using first 800 samples in linear kernel-
Training with first 800 samples-
Sample Size is 800
Accuracy is 0.9811320754716981
No. of Support Vectors are 14

4.3 Solution 4)c

Solution 1)

For C=0.0001 Degree=2:
Training Error is 0.3408.
Testing error is 0.3467.
No. of support vectors are 1112.

or C=0.0001 Degree=5:
Training Error is 0.0519.
Testing error is 0.0755.
No. of support vectors are 374.

False. Training error is lower at Q=5 compared to Q=2.

Solution 2)

For C=0.001 Degree=2:
Training Error is 0.0250.
Testing error is 0.0354.
No. of support vectors are 558.

For C=0.001 Degree=5:
Training Error is 0.0211.
Testing error is 0.0307.
No. of support vectors are 158.

True. The no. of support vectors at Q=5 are 158 which are lower than support vectors at Q=2 which have 558 vectors.

Solution 3)

For C=0.01 Degree=2:
Training Error is 0.0083.
Testing error is 0.0212.
No. of support vectors are 164.

For C=0.01 Degree=5:
Training Error is 0.0083.
Testing error is 0.0212.
No. of support vectors are 68.

False. Training error at Q= 5 is 0.0083 which is same as training error at Q=2.

Solution 4)
For C=1 Degree=2:
Training Error is 0.0045.
Testing error is 0.0189.
No. of support vectors are 30.

For C=1 Degree=5:
Training Error is 0.0045.
Testing error is 0.0165.
No. of support vectors are 26.

True. Testing error at Q=5 is 0.0165 which is lower than testing error at Q=2 which is 0.0189.

4.4 Solution 4)d)

Evaluating for C = 0.01
Training error- 0.005124919923126248
Test error- 0.01650943396226412

Evaluating for C = 1
Training error- 0.004484304932735439
Test error- 0.021226415094339646

Evaluating for C = 100
Training error- 0.0032030749519538215
Test error- 0.018867924528301883

Evaluating for C = 10000.0
Training error- 0.002562459961563124
Test error- 0.018867924528301883

Evaluating for C = 1000000.0
Training error- 0.002562459961563124
Test error- 0.0235849056603774

The training error is lowest for C= pow(10,6) and C= pow(10,4)
Testing error is lowest at C = 0.01

5 Question 5

5.1 Solution a)

Training Error for linear Support Vector Classifier is 0.00000.
Testing Error for linear
Support Vector Classifier is 0.02400.
No. of Support Vectors are 1084.

5.2 Solution b)

Running Support Vector Classifier ....
Training Error for RBF Support Vector Classifier is 0.00000.
Testing Error for RBF Support Vector Classifier is 0.50000.

```
No. of Support Vectors are 6000.


Running Polynomial SVC (degree=2) ....
Polynomial SVC (degree=2):
Training Error for polynomial Support Vector Classifier is 0.00050.
Testing Error for polynomial Support Vector Classifier is 0.02000.
Number of Support Vectors: 1332
```