COMP 540 Assignment #3

Yunda Jia yj32
Yu Wu yw92

March 6, 2020

1 Intuitions about support vector machines (10 points)

- a
  For maximizing the margin, we maximize distance between support vectors which means not only we classify two classes but we classify the most hard to classify points. For example, if the margin is extremely small, some points may be misclassified because of small margin.

- b
  No, it would not. SVM's hinge loss function is $J(\theta) = C\sum_{i=1}^{m} max(0, 1 - y^{(i)}h(x^{(i)})) + \frac{1}{2}||\theta||^2$, if the point is not support vector, $max(0, 1 - y^{(i)}h(x^{(i)}))$ is always 0 which does not affect loss function.

2 Fitting an SVM classifier by hand (15 points)

- Write down a vector that is parallel to the optimal vector θ.
  $x^{(1)} = 0$, $x^{(2)} = \sqrt{2}$ and θ is perpendicular to the decision boundary between the two points in the three-dimensional. $\Phi(x^{(2)}) - \Phi(x^{(1)})$ is also perpendicular to the decision boundary. So$\Phi(x^{(2)}) - \Phi(x^{(1)}) = (1, 2, 2) - (1, 0, 0) = (0, 2, 2)$ is a vector that is parallel to the optimal vector.

- What is the value of the margin that is achieved by this θ?

$$\begin{aligned} d &= \frac{\sqrt{0^2 + 2^2 + 2^2}}{2} \\ &= \sqrt{2} \end{aligned}$$

- Solve for the θ given that the margin is equal to $\frac{2}{||\theta||}$.

$$\begin{aligned} \frac{1}{||\theta||} &= \sqrt{2} \\ ||\theta|| &= \frac{1}{\sqrt{2}} \end{aligned}$$

Since θ is parallel to (0, 2, 2), then let $\theta = (0, 2\lambda, 2\lambda)$.

$$\begin{aligned} ||\theta|| &= \sqrt{0 + 4\lambda^2 + 4\lambda^2} = \frac{1}{\sqrt{2}} \\ \lambda &= \frac{1}{4} \end{aligned}$$

So $\theta = (0, \frac{1}{2}, \frac{1}{2})$.

- Solve for the intercept 0 using your value for θ and the inequalities above.

$$\begin{aligned} y^{(1)}(\theta^T\Phi(x^{(1)}) + \theta_0) &\geq 1 \\ y^{(2)}(\theta^T\Phi(x^{(2)}) + \theta_0) &\geq 1 - 1 * (0 + \theta_0) && \geq 1 \\ -1 * (2 + \theta_0) &\geq 1\theta_0 && = -1 \end{aligned}$$

Since both points are support vectors, the inequalities will be tight, $\theta = -1$.

- Write down the equation for the decision boundary in terms of $\theta, \theta_0 and x$.

$$\theta^T\Phi(x) + \theta_0 = 0$$

$$\frac{1}{2}x^2 + \frac{\sqrt{2}}{2}x - 1 = 0$$

3 Support vector machines for binary classification (25 points)

3.1 Support vector machines

3.1.1 The hinge loss function and gradient (5 points)

$J = 1.0000$
$grad = [-0.12956186 - 0.00167647]$

3.1.2 Example dataset 1: impact of varying C (2 points)

- When $C = 1$

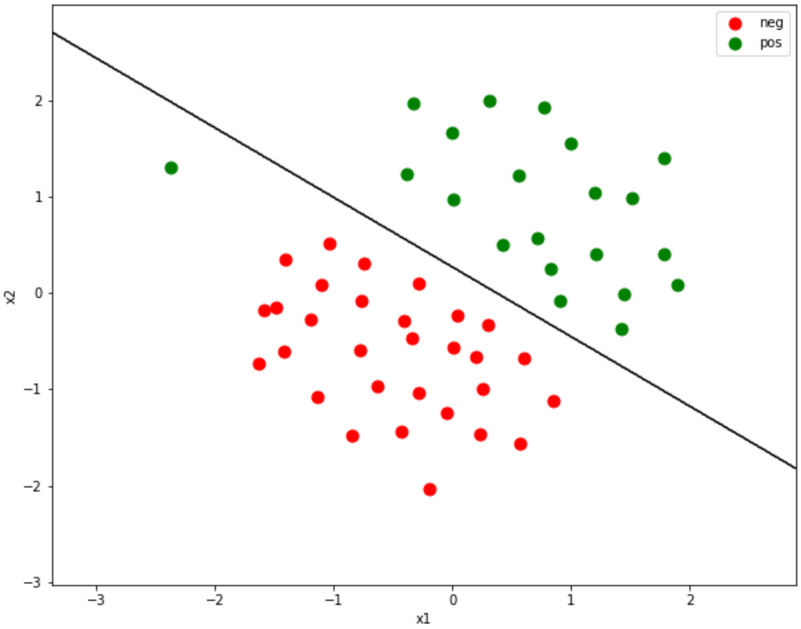

Figure 1: Decision boundary when C = 1

- When $C = 100$

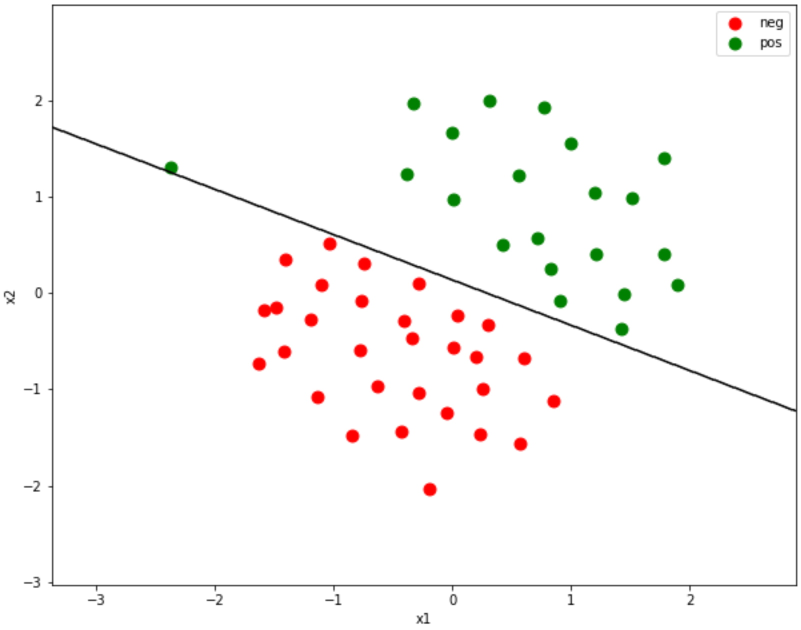

Figure 2: Decision boundary when C = 100

3.1.3 Gaussian kernel (3 points)

Gaussian kernel value (should be around 0.324652) = 0.32465.

3.1.4 Example dataset 2: learning non-linear boundaries

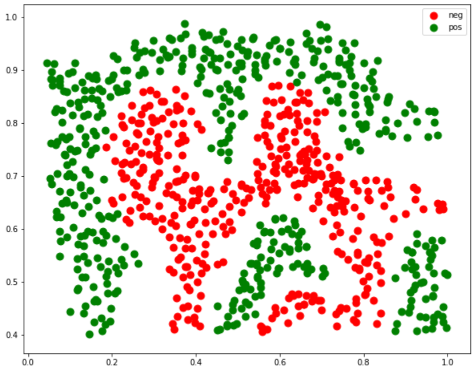

Figure 3: Original Data

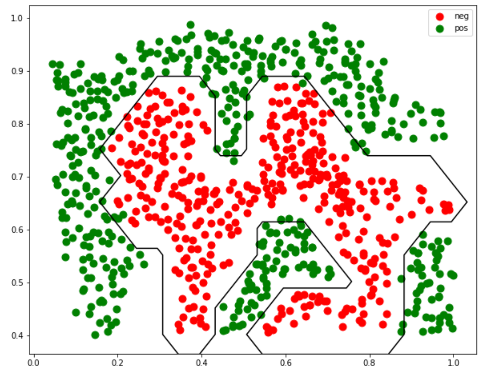

Figure 4: Gaussian kernel decision boundary

3.2 Example dataset 3: selecting hyper parameters for SVMs (5 points)

See my implementation in binary_svm.ipynb.

3.3 Spam Classification with SVMs (10 points)

3.3.1 Training SVMs for spam classification (10 points)

From the begining code, we already seperate the test set 3600 and validation set is 400. We scale the data matrix by using StandardScaler() method. For learning rate, we have choose serval number in [1e-5,3e-5,1e-4,3e-4,1e-3,3e-3,1e-2] and find the best learning rate are 0.003. We choose tne number of itrations is 30000 and panalty parameter is 300. Besides, for the kernal we choose Gaussian kernal and the sigma for it is 10 . Finally, Accuracy on the training set is 0.99925 and accuracy on the testing set is 0.988
The top 15 words of spam are listed below:
remot
clearli
otherwis
mondai
wife
info
with
dollarac
doesn
gt
human
militari
mark
bush
similar

4 Support vector machines for multi-class classification (35 points)

4.1 Loss and gradient function for multi-class SVM - naive version (5 points)

See my implementation in linear_svm.ipynb.

4.1.1 Loss and gradient function for multi-class SVM - vectorized version (10 points)

See my implementation in linear_svm.ipynb.

4.1.2 Prediction function for multi-class SVM (5 points)

My results are shown below.
training accuracy: 0.400510
validation accuracy: 0.377000

4.1.3 Tuning hyper parameters for training a multi-class SVM (10 points)

My best accuracy is 0.387500 which is shown on jupyter notebook.

4.1.4 Comparing the performance of multi-class SVM and softmax regression (5 points)

For Cifar-10, multi-class SVM takes longer to train but softmax achieves higher test accuracy. Test accuracy on SVM is 0.387500, but best test accuracy on softmax achieves 0.413.
For softmax

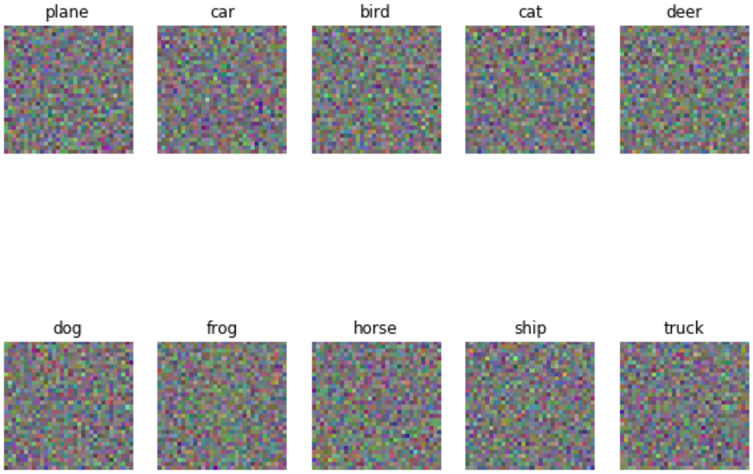

Figure 5: Softmax theta

For SVM

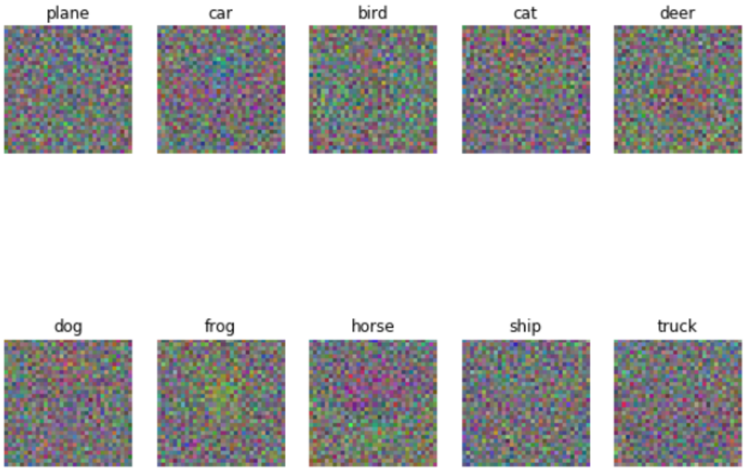

Figure 6: SVM theta

For visualization of theta, there is no much differentiation.
For softmax, batch size is 2000, number of iteration is 1700, learning rate is 5e-7, regularization term is 5e5.
For multi-class SVM, number of iteration is 5000, learning rate is 1e-7, regularization term is 5e+5. The hyper-parameter choice is similar between these two models.