CS/ECE/ME532 Activity 22

Estimated Time: 30 minutes for P1, 30 minutes for P2, 10 minutes for P3

1. **Kernel regression.** Kernel regression predicts a value d corresponding to value x as $\hat{d}(x) = \sum_{i=1}^{N} \alpha_i K(x, x^i)$ where the measured data is $(d^i, x^i), i = 1, 2, \ldots N$ and $K(u, v)$ is the kernel function. We will assume Gaussian kernels, $K(u, v) = \exp\left(-(u-v)^2/(2\sigma^2)\right)$. Scripts are provided to help you explore properties of kernel regression with respect to the kernel parameter σ and ridge regression parameter λ.

   a) Run the regression script with $\sigma = 0.04$ and $\lambda = 0.01$. Figure 1 displays several of the kernels $K(x, x^i)$. What is the value x^i associated with the kernel having the third peak from the left? What property of the kernel is determined by x^i? What property is determined by σ?

   b) Run the regression script for the following choices of regularization and kernel parameters:

   i. $\lambda = 0.01, \sigma = 0.04$
   ii. $\lambda = 0.01, \sigma = 0.2$
   iii. $\lambda = 0.01, \sigma = 1$
   iv. $\lambda = 1, \sigma = 0.04$
   v. $\lambda = 1, \sigma = 0.2$

   (Note that you need to rerun the entire script each time to ensure the random number generator is reset and you obtain identical data.) You may choose additional cases if it helps you understand the nature of the solution. Discuss how λ and σ affect the characteristics of the kernel regression to the measured data, and support your conclusions with rationale and plots.

   c) What principle could you apply to select appropriate values for λ and σ?

2. **Kernel Classification.** The kernel classification script performs classification using the squared error loss using the Gaussian kernel $K(\boldsymbol{u}, \boldsymbol{v}) = \exp\left(-||\boldsymbol{u} - \boldsymbol{v}||_2^2/(2\sigma^2)\right)$. The code is set up to use `N=500` training samples.

   The code creates a contour plot of the predicted class, *before thresholding* (i.e, before applying the sign function).

   Run the code for the following values of the kernel parameter σ.

   a) $\sigma = 5$

   b) $\sigma = 0.05$

c) $\sigma = 0.005$

Use the results to discuss the impact of the kernel parameter σ. Is there a downside to choosing a very small value for σ? Run additional values for σ if needed.

3. SVM. You use a kernel-based support vector machine for binary classification with labels $d^i = \{+1, -1\}$. Given training features and labels $(\boldsymbol{x}^i, d^i), i = 1, 2, \ldots, N$ you use a kernal $K(\boldsymbol{u}, \boldsymbol{v})$ and design the classifier weights $\boldsymbol{\alpha}$ as

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \sum_{i=1}^{N} \left(1 - d^i \sum_{j=1}^{N} \alpha_j K(\boldsymbol{x}^i, \boldsymbol{x}^j)\right)_+ + \lambda \sum_{i=1}^{N} \sum_{j=1}^{n} \alpha_i \alpha_j K(\boldsymbol{x}^i, \boldsymbol{x}^j)$$

a) Assume the optimization problem has been solved to obtain the weights $\boldsymbol{\alpha}$. Express the classification procedure for a measured feature $\boldsymbol{x}$.

b) Suppose $N = 1000$ and $\alpha_i = 0, i = 1, 2, \ldots, 99, 102, 103, \ldots, 1000$. Identify the support vectors and write the classification procedure in terms of the support vectors.

532 Activity 22 - DEVIN BRESSER

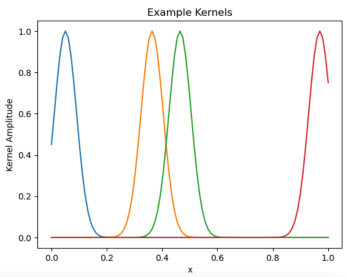
1a.) The x_i associated w/ the green curve is 46.

Adjusting x_i moves the kernes left & right

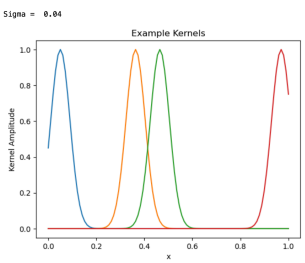
Adjusting σ changes the width of the kernels.

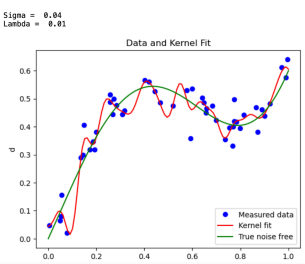
```
[6]: sigma = 0.04 #defines Gaussian kernel width
     p = 100 #number of points on x-axis

     # Display examples of the kernels
     x_test = np.linspace(0,1.00,p)  # uniformly sample interval [0,1]
     j_list = [5, 36, 46, 96]  #list of indices for example kernels

     Kdisplay = np.zeros((p,len(j_list)),dtype=float)

     for i in range(p):
         for j in range(len(j_list)):
             Kdisplay[i,j]= np.exp(-(x_test[i]-x_test[j_list[j]])**2/(2*sigma**2))

     print('Sigma = ',sigma)
     plt.plot(x_test, Kdisplay)
     plt.title('Example Kernels')
     plt.xlabel('x')
     plt.ylabel('Kernel Amplitude')
     plt.show()
```

```
Sigma =  0.04
```

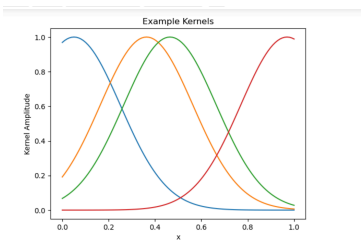


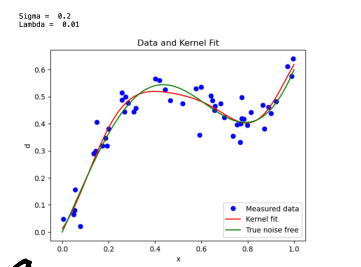
b.)



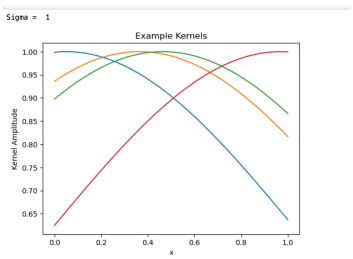


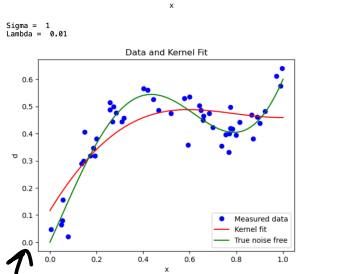
$\lambda = 0.01 \quad \sigma = 0.04$





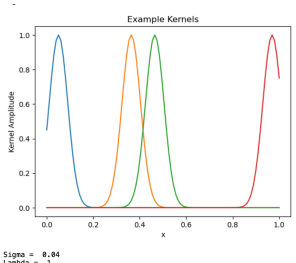
$\lambda = 0.01 \quad \sigma = 0.2$

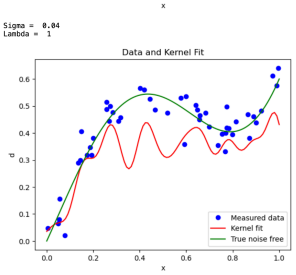




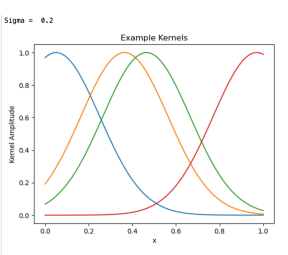
$\lambda = 0.01 \quad \sigma = 1$

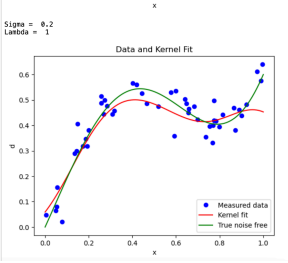
$\lambda = 1 \quad \sigma = 0.04$





$\lambda = 1 \quad \sigma = 0.2$





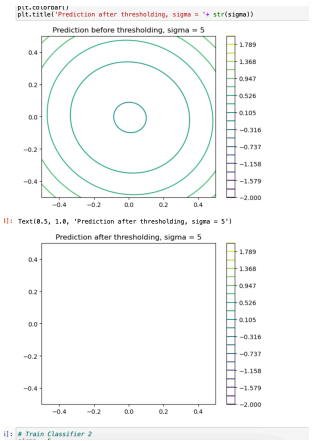
Comment:
If lambda is low and held constant, then increasing sigma reduces the "order" of the solution fit. That is, when sigma is low, the training data is more exactly fit by the curve and resembles a high order polynomial, which could indicate overfitting.

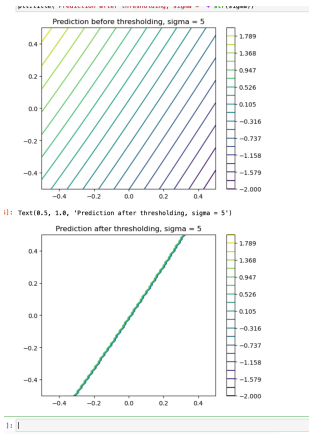
If lambda is high and sigma is low, then the regularization is very aggressive, and the solution is pulled away from a good fit. But, if lambda is high and sigma is also high, the regularization is smoothed over. This is because increasing the width of the kernels reduces each data point's individual contribution.

Problem 1c comment: The principle of cross validation across a variety of lambdas and sigmas is a good approach to determine optimal values for them.

p2.) a)

$\sigma = 5$





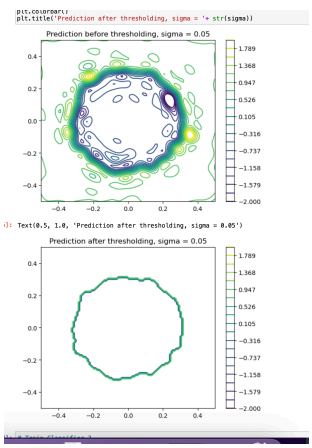
As the kernel parameter sigma is reduced, the decision boundary is increased in fidelity and becomes more complex. It exhibits zig-zagging geometry around individual data points in the training data, indicating an over-fit to the training data and a high sensitivity to noise. This would lead to a higher error rate on real world data.

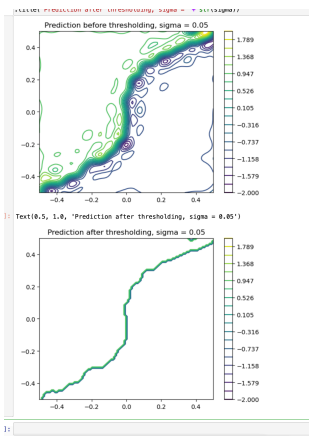
Also, decreasing Sigma increases the computational complexity of the model, increasing training times.

Regularization can be incorporated to mitigate some of the effects of a small sigma.

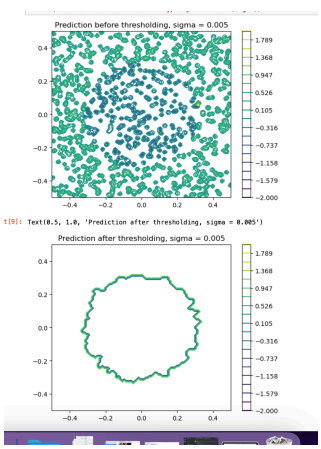
Sigma and lambda should be again tuned via cross validation to ensure a balanced model.

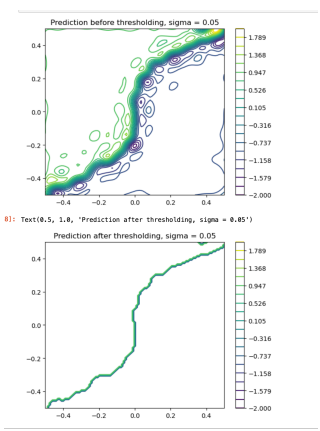
$\sigma = 0.05$





$\sigma = 0.005$





3. SVM. You use a kernel-based support vector machine for binary classification with labels $d^i = \{+1, -1\}$. Given training features and labels $(\boldsymbol{x}^i, d^i), i = 1, 2, \ldots, N$ you use a kernal $K(\boldsymbol{u}, \boldsymbol{v})$ and design the classifier weights $\boldsymbol{\alpha}$ as

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \sum_{i=1}^{N} \left(1 - d^i \sum_{j=1}^{N} \alpha_j K(\boldsymbol{x}^i, \boldsymbol{x}^j)\right)_+ + \lambda \sum_{i=1}^{N} \sum_{j=1}^{n} \alpha_i \alpha_j K(\boldsymbol{x}^i, \boldsymbol{x}^j)$$

a) Assume the optimization problem has been solved to obtain the weights $\boldsymbol{\alpha}$. Express the classification procedure for a measured feature $\boldsymbol{x}$.

b) Suppose $N = 1000$ and $\alpha_i = 0, i = 1, 2, \ldots, 99, 102, 103, \ldots, 1000$. Identify the support vectors and write the classification procedure in terms of the support vectors.

a.) We know the decision boundary is $d(\underline{x}) = 0$.

And $d(\underline{x}) = \underline{\varphi}^T(\underline{x})\,\underline{w} = \sum_{j=1}^{n} \alpha_j K(\underline{x}, \underline{x}_j)$

So we need to compute $d(\underline{x})$ using $\uparrow$ and take the sign. If below decision boundary, it will be negative w/ label -1. If above boundary, label $+1$.

b.) The support vectors are the only training data points w/ nonzero corresponding α value. So in this case it's the training samples associated w/

$\alpha_{100}, \alpha_{101}$

So to classify compute $d(\underline{x}) = \sum_{j \in \{100, 101\}} \alpha_j K(\underline{x}, \underline{x}_j)$