Homework2

July 29, 2022

1 1. Collaborative Filtering, Kernels, Linear Regression

In this question, we will use the alternating projections algorithm for low-rank matrix factorization, which aims to minimize

$$J(U,V) = \frac{1}{2}\sum_{(a,i)\in D}(Y_{ai} - [UV^T]_{ai})^2 + \frac{\lambda}{2}\sum_{a=1}^{n}\sum_{j=1}^{k}U_{aj}^2 + \frac{\lambda}{2}\sum_{i=1}^{m}\sum_{j=1}^{k}V_{ij}^2$$

In the following, we will call the first term the squared error term, and the two terms with λ the regularization terms.

Let Y be defined as:

$$Y = \begin{bmatrix} 5 & ? & 7 \\ ? & 2 & ? \\ 4 & ? & ? \\ ? & 3 & 6 \end{bmatrix}$$

D is defined as the set of indices (a,i), where $Y_{(a,i)}$ is not missing. In this problem, we let $k = \lambda = 1$. Additionally, U and V are initialized as $U^{(0)} = [6,0,3,6]$, and $V^{(0)} = [4,2,1]$.

[5]:
```
display_matrix = lambda outer_product: list(map(lambda row: [print(f"{el:>5}",␣
 ↪end='') if el is not None else print(f"{'?':>5}", end='') for el in␣
 ↪row+['\n']], outer_product))
Y = [[5, None, 7],[None, 2, None],[4, None, None],[None, 3, 6]]
U = [6, 0, 3, 6]
V = [4, 2, 1]
lmbda = 1
k = 1

display_matrix(Y);
```

```
    5    ?    7
    ?    2    ?
    4    ?    ?
    ?    3    6
```

1.1 1. (a)

Compute $X^{(0)}$, the matrix of predicted rankings UV^T given the initial values for $U^{(0)}$ and $V^{(0)}$.

[6]:
```
outer_product = lambda v1, v2: list(map(lambda el1: list(map(lambda el2:␣
 ↪el2*el1, v2)), v1) )

X0 = outer_product(U,V)


display_matrix(X0);
print()
display_matrix(Y);
```

```
  24   12    6
   0    0    0
  12    6    3
  24   12    6

   5    ?    7
   ?    2    ?
   4    ?    ?
   ?    3    6
```

1.2 1. (b)

Compute the squared error term, and the regularization terms in for the current estimate X.

Enter the squared error term (including the factor $\frac{1}{2}$):

[7]:
```
squared_error = 1/2 * sum([sum([(el_Y - el_X)**2 for el_Y, el_X in zip(row_y,␣
 ↪row_x) if el_Y is not None]) for row_y, row_x in zip(Y,X0)])
print(f'{squared_error = }')
```

```
squared_error = 255.5
```

Enter the regularization term (the sum of all the regularization terms):

[8]:
```
regularization_term = lmbda/2 * sum(map(lambda x: x**2, [*U, *V]))
print(f'{regularization_term = }')
```

```
regularization_term = 51.0
```

1.3 1. (c)

Suppose V is kept fixed. Run one step of the algorithm to find the new estimate $U^{(1)}$.

[ ]:
```

```

2 2. Feature Vectors Transformation

Consider a sequence of n-dimensional data points, $x^{(1)}, x^{(2)}, ...$, and a sequence of m-dimensional feature vectors, $z^{(1)}, z^{(2)}, ...$, extracted from the x's by a linear transformation, $z^{(i)} = Ax^{(i)}$. If m is much smaller than n, you might expect that it would be easier to learn in the lower dimensional feature space than in the original data space.

2.1 2. (a)

Suppose $n = 6$, $m = 2$, z_1 is the average of the elements of x, and z_2 is the average of the first three elements of x minus the average of fourth through sixth elements of x. Determine A.

A = [[1/6, 1/6, 1/6, 1/6, 1/6, 1/6],[1/3, 1/3, 1/3, -1/3, -1/3, -1/3]]

2.2 2. (b)

Using the same relationship between z and x as defined above, suppose $h(z) = sign(\theta_z \cdot z)$ is a linear classifier for the feature vectors, and $g(x) = sign(\theta_x \cdot x)$ is a linear classifier for the original data vectors. Given a θ_z that produces good classifications of the feature vectors, determine a θ_x that will identically classify the associated x's.

$$
\begin{aligned}
sign(\theta_z \cdot z) &= \theta_z^{(1)} \cdot \frac{1}{6}\sum_{i=1}^{6} x_i + \theta_z^{(2)} \frac{1}{3}\sum_{i=1}^{3} x_i - \theta_z^{(2)} \frac{1}{3}\sum_{i=4}^{6} x_i \\
&= sign\left(\begin{bmatrix} \frac{1}{6}\theta_z^{(1)} + \frac{1}{3}\theta_z^{(2)} \\ \vdots \\ \frac{1}{6}\theta_z^{(1)} - \frac{1}{3}\theta_z^{(2)} \\ \vdots \end{bmatrix}^T \cdot x \right) \\
&= sign\left((\theta_z^T \cdot A)^T \cdot x \right)
\end{aligned}
$$

$$\text{Thus:} \theta_x = A^T \theta_z$$

2.3 2. (c)

Given the same classifiers as in (b), if there is a θ_x that produces good classifications of the data vectors, will there **always** be a θ_z that will identically classify the associated z's?

ANSWER No?

2.4 2. (d)

Given the same classifiers as in (b), if there is a θ_x that produces good classifications of the data vectors, will there **always** be a θ_z that will identically classify the associated z's?

Answer Yes.

3 3. Kernels

3.1 3. (a)

Let $x, q \in \mathbb{R}^2$ be two feature vectors, and let $K(x,q) = (x^T q + 1)^2$. This is often known as a polynomial kernel. It's simple to compute: you just take the dot product between two feature vectors, add one, and then square the result. But what kind of feature mapping does this kernel implicitly use?

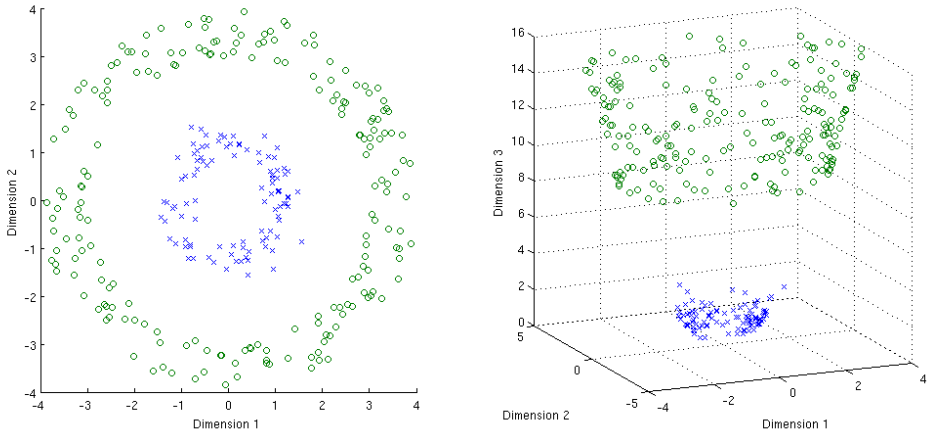
Assuming we can write $K(x,q) = \phi(x)^T \phi(q)$, derive an expression for $\phi(x)$.

$$\begin{aligned} K(x,q) = \phi(x)^T \phi(q) &= (x^T q + 1)^2 \\ &= (x^T q)^2 + 2x^T q + 1 \\ &= [x^T x^T, \sqrt{2}x^T, 1] \begin{bmatrix} qq \\ \sqrt{2}q \\ 1 \end{bmatrix} \end{aligned}$$

4 4. Kernels II

4.1 4. (a)

In the figure below, a set of points in 2-D is shown on the left. On the right, the same points are shown mapped to a 3-D space via some transform $\phi(x)$, where x denotes a point in the 2-D space. Notice that $\phi_1(x) = x_1$ and $\phi_2(x) = x_2$, or in other words, the first and second coordinates are unchanged by the transformation.



Which functions could have been used to compute the value of the 3rd coordinate:

ANSWER

$\phi_3(x) = x_1^2 + x_2^2$

Think about how a linear decision boundary in the 3 dimensional space ($\{\phi \in \mathbb{R}^3 : \theta \cdot \phi + \theta_0 = 0\}$) might appear in the original 2 dimensional space.

For example, suppose the decision boundary in the 3 dimensional space is $z = 4$.

Provide an equation $f(x_1, x_2) = 0$ in the 2 dimensional space such that all the points (x_1, x_2) with $f(x_1, x_2) > 0$ correspond to $z > 4$ in the 3 dimensional space.

$f(x_1, x_2) = 0 = x_1^2 + x_2^2 - 4$

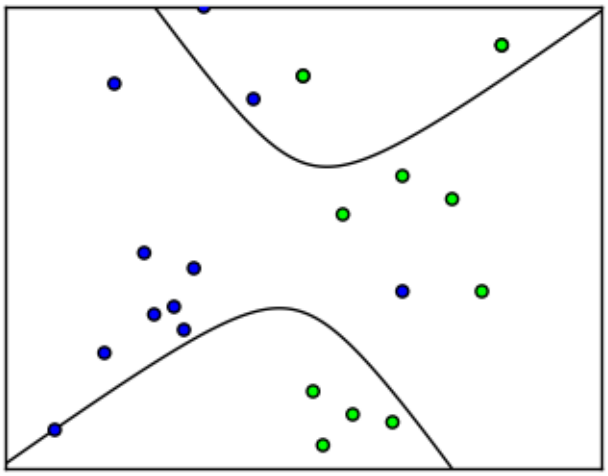
4.2 4. (b)

Consider fitting a kernelized SVM to a dataset $(x^{(i)}, y^{(i)})$ where $x^{(i)} \in \mathbb{R}^2$ and $y^{(i)} \in \{1, -1\}$ for all $i = 1, \ldots, n$. To fit the parameters of this model, one computes θ and θ_0 to minimize the following objective:

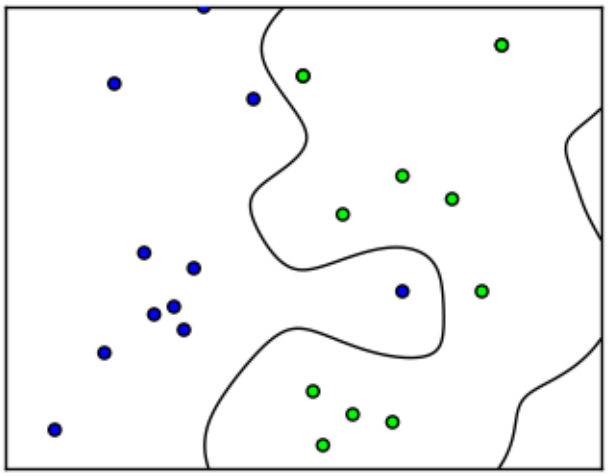
$$L(\theta, \theta_0) = \frac{1}{n} \sum_{i=1}^{n} \text{Loss}_h \left(y^{(i)} (\theta \cdot \phi(x^{(i)}) + \theta_0) \right) + \frac{\lambda}{2} |\theta|^2$$

where ϕ is the feature vector associated with the kernel function. Note that, in a kernel method, the optimization problem for training would be typically expressed solely in terms of the kernel function $K(x, x')$ (dual) rather than using the associated feature vectors $\phi(x)$ (primal). We use the primal only to highlight the classification problem solved.

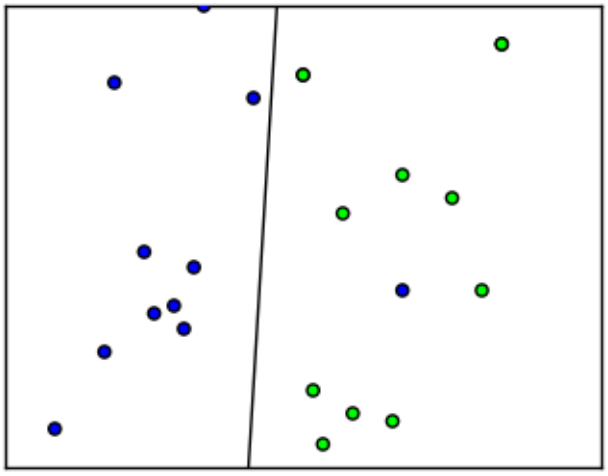
The plots below show 4 different kernelized SVM models estimated from the same 11 data points. We used a different kernel to obtain each plot but got confused about which plot corresponds to which kernel. Help us out by assigning each plot to one of the following models: linear kernel, quadratic kernel, order 3 kernel, and RBF kernel.



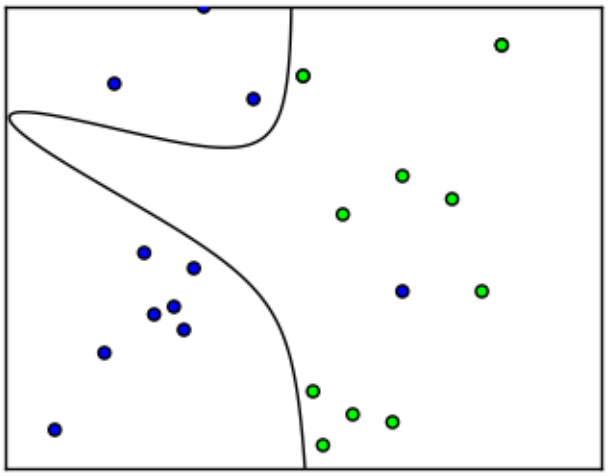
Kernel: Quadratic



Kernel: Radial basis function



Kernel: Linear



Kernel: Third order

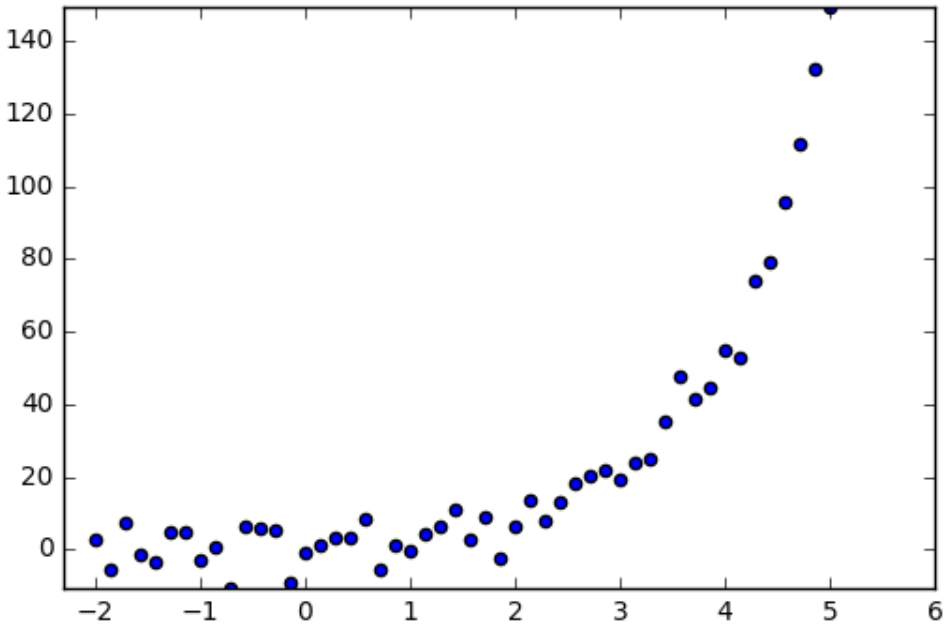
How would you describe qualitatively how the resulting classifiers vary with the value of λ? If the value of λ is increased, the fitting of model would be:

ANSWER Worse fit on training data, smoother curves (flatter decision boundary)

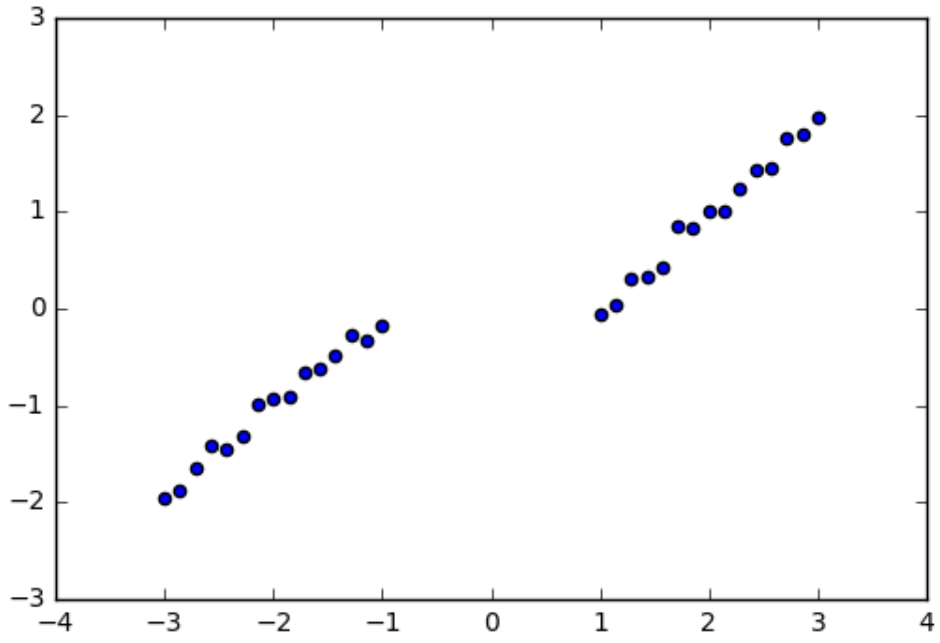
5 5. Linear Regression and Regularization

5.1 5. (a)

For each of the datasets below, provide a simple feature mapping ϕ such that the transformed data $(\phi(x^{(i)}), y^{(i)})$ would be well modeled by linear regression.



$\phi(x) = \exp(x)$



$\phi(x) = x - sign(x)$

5.2 5. (b)

Consider fitting a ℓ_2-regularized linear regression model to data $(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})$ where $x^{(t)}, y^{(t)} \in \mathbb{R}$ are scalar values for each $t = 1, \ldots, n$. To fit the parameters of this model, one solves

$$\min_{\theta \in \mathbb{R}, \theta_0 \in \mathbb{R}} L(\theta, \theta_0)$$

where

$$L(\theta, \theta_0) = \sum_{t=1}^{n} (y^{(t)} - \theta x^{(t)} - \theta_0)^2 + \lambda \theta^2$$

Here $\lambda \geq 0$ is a pre-specified fixed constant, so your solutions below should be expressed as functions of and the data. This model is typically referred to as ridge regression .

Write down an expression for the gradient of the above objective function in terms of θ.

$$\frac{\partial L(\theta, \theta_0)}{\partial \theta} = 2 \sum_{t=1}^{n} (\theta x^{(t)} + \theta_0 - y^{(t)}) x^{(t)} + 2\lambda\theta$$

$$\frac{\partial L(\theta, \theta_0)}{\partial \theta_0} = 2 \sum_{t=1}^{n} (\theta x^{(t)} + \theta_0 - y^{(t)})$$

5.3 5. (c)

Find the closed form expression for θ and θ_0 which solves the ridge regression minimization above.

Assume θ is fixed, write down an expression for the optimal $\hat{\theta}_0$ in terms of $\theta, x^{(t)}, y^{(t)}, n$.

$$\hat{\theta}_0 = \frac{1}{n} \sum_{t=1}^{n} y^{(t)} - \theta \frac{1}{n} \sum_{t=1}^{n} x^{(t)}$$

bla bla…