HW 6

Wednesday, December 8, 2021 1:55 PM

1:

A:

$\sum_{j=1}^{m} w_j K_j : w_j \geq 0$ and K is a valid kernel matrix that is symmetric and positive semi definite

Analyze the summation of 2 arbitrary $w_j K_j$ pairs.

Proof of Symmetric

$w_j K_j + w_i K_i$

$w_j K_j$ & $w_i K_i$ are both still symmetric by definition. And the addition of two symmetric matrices is still symmetric.

Proof of PSD

Consider $w_j \vec{x}^T K_j \vec{x} \geq 0$ and $w_i \vec{x}^T K_i \vec{x} \geq 0$

Now consider

$w_j \vec{x}^T K_j \vec{x} + w_i \vec{x}^T K_i \vec{x} \geq 0$ Both terms are clearly ≥ 0

$\vec{x}^T (w_j K_j + w_i K_i) \vec{x} \geq 0$

Thus we show the summation of 2 arbitrary valid kernel functions is still PSD

Thus the $K = \sum_{j=1}^{m} w_j K_j$ is a valid kernel function.

B:

$K(x_i, x_j) = K_1(x_i, x_j) K_2(x_i, x_j)$

Proof of Symmetry

Choose arbitrary x_i, x_j

$K_1(x_i, x_j) = < \phi_1(x_i), \phi_1(x_j) >$

$K_2(x_i, x_j) = < \phi_2(x_i), \phi_2(x_j) >$

$K_1 * K_2 = < \phi_1(x_i), \phi_1(x_j) > * < \phi_2(x_i), \phi_2(x_j) >$

Now we check if $K(x_i, x_j) = K(x_j, x_i)$

$K_1(x_j, x_i) = < \phi_1(x_j), \phi_1(x_i) > = < \phi_1(x_i), \phi_1(x_j) >$

$K_2(x_j, x_i) = < \phi_2(x_j), \phi_2(x_i) > = < \phi_2(x_i), \phi_2(x_j) >$

$K_1 * K_2 = < \phi_1(x_j), \phi_1(x_i) > * < \phi_2(x_j), \phi_2(x_i) > = < \phi_1(x_i), \phi_1(x_j) > * < \phi_2(x_i), \phi_2(x_j) >$

By nature of inner products they are commutative thus $K(x_i, x_j) = K(x_j, x_i)$

Thus symmetric

Proof of PSD

Consider $\vec{x}^T K_1 \vec{x} \geq 0$ and $\vec{x}^T K_2 \vec{x} \geq 0$

Now consider

$\vec{x}^T K_1 \vec{x} * \vec{x}^T K_2 \vec{x} \geq 0$ Because both terms are greater or equal to zero

$\vec{x}^T (K_1 K_2) \vec{x} \geq 0$ By means of factoring

Thus $K = K_1 * K_2$ is PSD

Thus $K(x_i, x_j) = K_1(x_i, x_j) K_2(x_i, x_j)$ is a valid kernel function.

1 (Continued):

C:

Proof of Symmetry

Choose x_1, x_2 arbitrarily in R

$K(x_1, x_2) = (x_1 x_2 + 1)^{2015}$

Consider $K(x_2, x_1) = (x_2 x_1 + 1)^{2015} = (x_1 x_2 + 1)^{2015}$

Thus the ij position equals the ji position thus symmetric

Proof of PSD

$$K(x, x') = \sum_i \sum_j \left(x_i x_j + 1\right)^{2015}$$

Consider $\vec{z}^T K \vec{z} = \sum_i \sum_j \sum_k z_i \left(x_{i_k} x_{j_k} + 1\right)^{2015} z_i \geq 0$

Thus PSD

Thus $K(x, x') = (xx' + 1)^{2015}$ is a valid kernel function

D:

Proof of Symmetry

$$K(x, x') = e^{\left(-\frac{(x-x')^2}{2}\right)} = e^{\left(-\frac{(x^2 - 2xx' + x'^2)}{2}\right)}$$

Consider $K(x', x) = e^{\left(-\frac{(x'-x)^2}{2}\right)} = e^{\left(-\frac{(x'^2 - 2xx' + x^2)}{2}\right)} = e^{\left(-\frac{(x^2 - 2xx' + x'^2)}{2}\right)}$

Thus the ij position equals the ji position thus symmetric

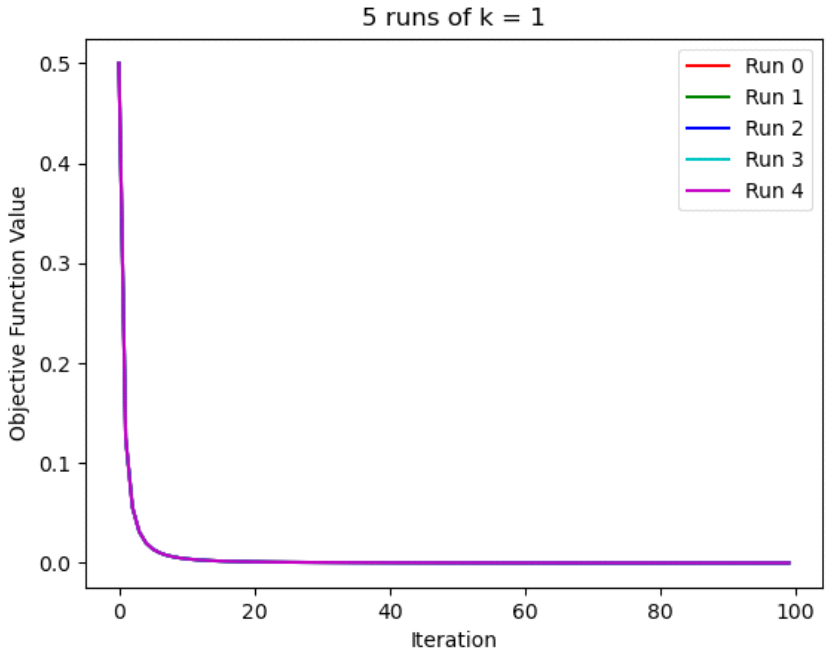
Proof of PSD

Consider $\vec{z}^T K \vec{z} = \sum_i \sum_j \sum_k z_i \left(e^{\left(-\frac{(x_{i_k} - x_{j_k})^2}{2}\right)}\right) z_i = \sum_i \sum_j \sum_k z_i^2 \left(e^{\left(-\frac{(x_{i_k} - x_{j_k})^2}{2}\right)}\right) \geq 0$

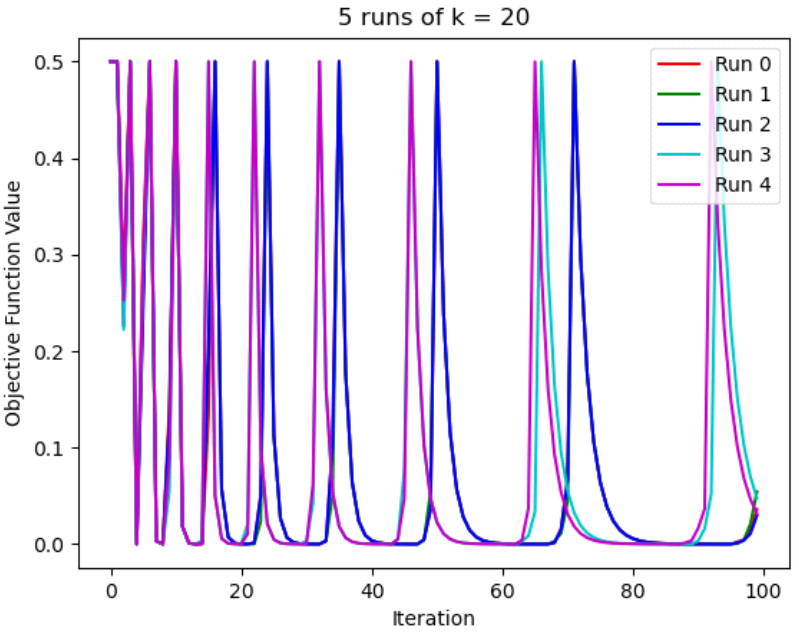
The exp() component will always be positive and z_i^2 is always either positive or 0. Because we show that the breadown of $\vec{z}^T K \vec{z} \geq 0$ the Kernel function K is positive semi definite.

Thus $K(x, x') = e^{\left(-\frac{(x-x')^2}{2}\right)}$ is a valid kernel function

2:

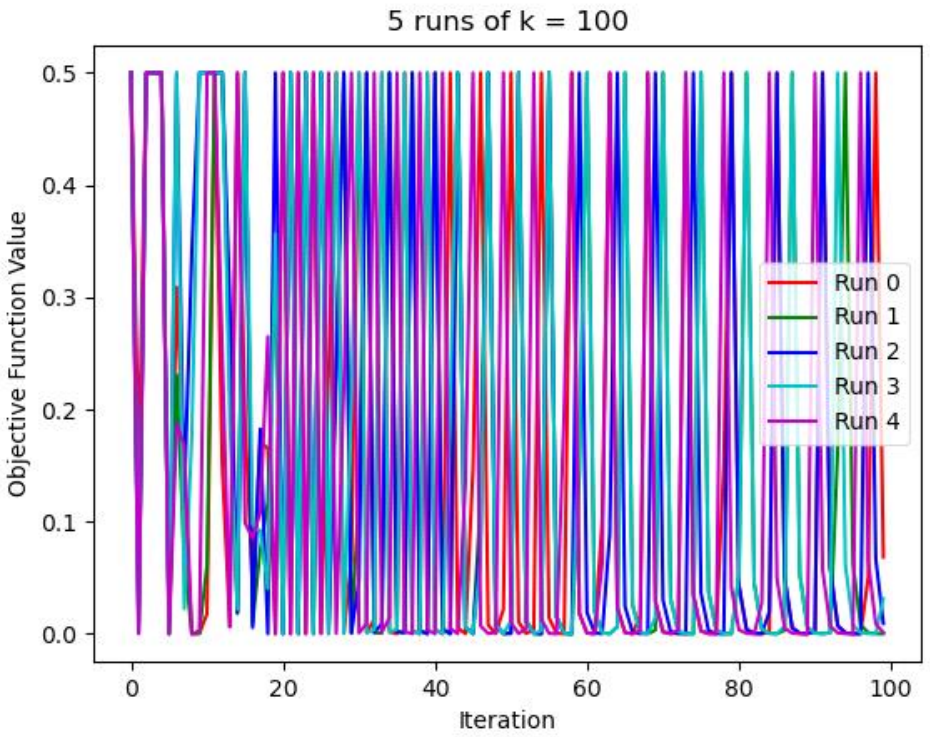



```
The average time per run was: 1.940625s
The Standard Deviation of time per run was: 0.020728904939721248s
```

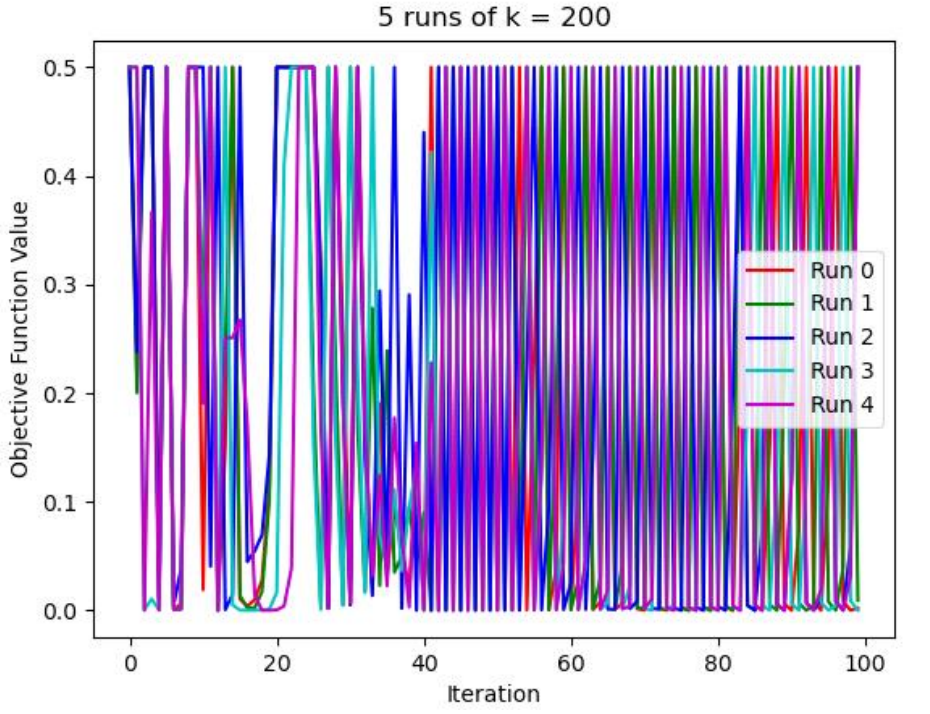



```
The average time per run was: 2.628125s
The Standard Deviation of time per run was: 0.022963966338592295s
```

2:

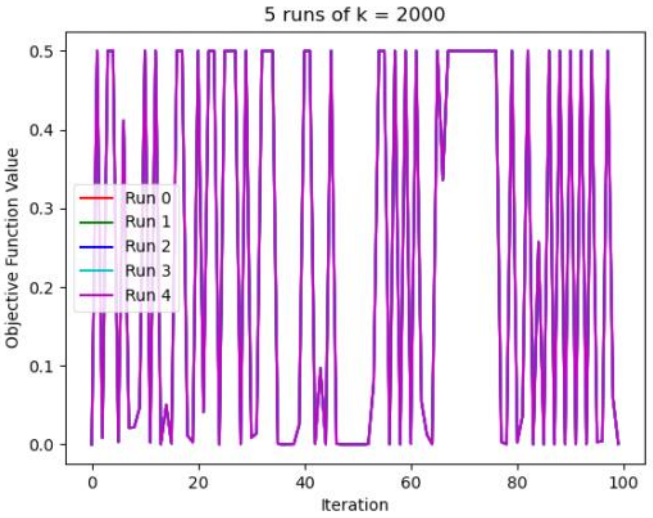



```
The average time per run was: 2.690625s
The Standard Deviation of time per run was: 0.018221724671391566s
```




```
The average time per run was: 2.8125s
The Standard Deviation of time per run was: 0.038273277230987154s
```

2:




```
The average time per run was: 3.475s
The Standard Deviation of time per run was: 0.015934435979977454s
```

Approach for Algorithm:

INPUT: S, λ, T, k
INITIALIZE: Choose $\mathbf{w}_1$ s.t. $\|\mathbf{w}_1\| \leq 1/\sqrt{\lambda}$
FOR $t = 1, 2, \ldots, T$
Choose $A_t \subseteq S$, where $|A_t| = k$
Set $A_t^+ = \{(\mathbf{x}, y) \in A_t : y \langle \mathbf{w}_t, \mathbf{x} \rangle < 1\}$
Set $\eta_t = \frac{1}{\lambda t}$
Set $\mathbf{w}_{t+\frac{1}{2}} = (1 - \eta_t \lambda)\mathbf{w}_t + \frac{\eta_t}{k} \sum_{(\mathbf{x},y) \in A_t^+} y\mathbf{x}$
Set $\mathbf{w}_{t+1} = \min\left\{1, \frac{1/\sqrt{\lambda}}{\|\mathbf{w}_{t+\frac{1}{2}}\|}\right\} \mathbf{w}_{t+\frac{1}{2}}$
OUTPUT: $\mathbf{w}_{T+1}$

Implemented exactly from the paper
Laplace smoothing was added for the full dataset run.
k was chosen to be either 1, 20, 100, 200, or N. In this case N=2000

Results:

$k = 1$ for a rudimentary stochastic gradient descent worked very well. The objective function was minimized well. For all other versions of k it did not work as well. What was interesting was that the algorithm was capable of minimizing the objective function very well. It even reached 10^{-7} territory. However it kept on swinging between high error and low error. It should be noted though that at the end of each runs of $k = 20,100,200,2000$ the last outputed $\vec{w}$ did minimize the objective function well.

The algorithm was decently fast. It was much faster than other mini batch methods that we implemented in the past, and this time we had a larger dataset. And as expected the larger the k the longer the algorithm took but not by much. It scales very well as N gets large

Remark to grader:

$k = 20,100,200,2000$ were implemented very poorly. I have followed the formula exactly as shown in the paper and instructions for $k = 20,100,200,2000$ in the homework but for some reason the gradient descent does not descend properly. I have no other explanation for my results other than that. My guess is that either some pre-processing was needed or the bounding of the pegasos algorithm (the min function and the λ constant) were poorly implemented.