CSE546 Homework 3

Chuanmudi Qin

May 28, 2020

Problem A1

Proof. a. True

The new coordinate after PCA can be obtained by diagnolized the covariance matirx $C = X^T X = PDP^{-1}$, where P is the new coordinate who has rank(C) = d = rank(X) the full P matrix preserves all the information in X, therefore reconstruction error is 0.

□

Proof. b. False

It really depends on the data to see what are the most suitable linear classifiers.

□

Proof. c. True

Due to the fact that usually we randomly choosing B sets containing k samples, there are chances that x_i could occur multiple times

□

Proof. d. False, under the premise that each **column** of X is a obervation.

the matrix U should be the eignenvectors

□

Proof. e. True

noises in the new coordinates are associated with the directions of smaller eigenvalues. Denoise will be helpful for extracting useful information

□

Proof. f. False.

Supposed that we have n data points and k classe in the original data set. We could take n classes to get zero error but the result will not be meaningfull. □

Proof. g. I should decrease σ

□

Problem A2

Proof. known that:

$$
\begin{aligned}
K(x,x') &= e^{-\frac{(x-x')^2}{2}} = e^{\frac{-x^2+2xx'-x'^2}{2}} = e^{\frac{-x^2-x'^2}{2}} * e^{-2xx'} \\
&= e^{\frac{-x^2-x'^2}{2}}(1 + \frac{2xx'}{1!} + \frac{(2xx')^2}{2!} + \frac{(2xx')^3}{3!} + \ldots) \\
&= e^{\frac{-x^2-x'^2}{2}}(1 * 1 + \sqrt{\frac{2}{1!}}x * \sqrt{\frac{2}{1!}}x' + \sqrt{\frac{2}{2!}}x^2 * \sqrt{\frac{2}{2!}}x'^2 + \sqrt{\frac{2}{3!}}x^3 * \sqrt{\frac{2}{3!}}x'^3 + \ldots) \\
&=< e^{-x^2}[\sqrt{\frac{2}{1!}}x, \sqrt{\frac{2}{2!}}x^2, \sqrt{\frac{2}{3!}}x^3, \ldots], e^{-x'^2}[\sqrt{\frac{2}{1!}}x', \sqrt{\frac{2}{2!}}x'^2, \sqrt{\frac{2}{3!}}x'^3, \ldots] > \\
&= \phi(x)^T\phi(x')
\end{aligned}
$$

□

Problem A3

Proof. A3a)

1) For rbf kernel the hyperparameters are chose from $d = [.01, .1, 1.10, 20, 50, 90, 100, 105, 200, 500]$ and $\lambda = np.arange(0, 1, 0.05)$

the optimal lambda value is 0.19, the optimal d value is 100

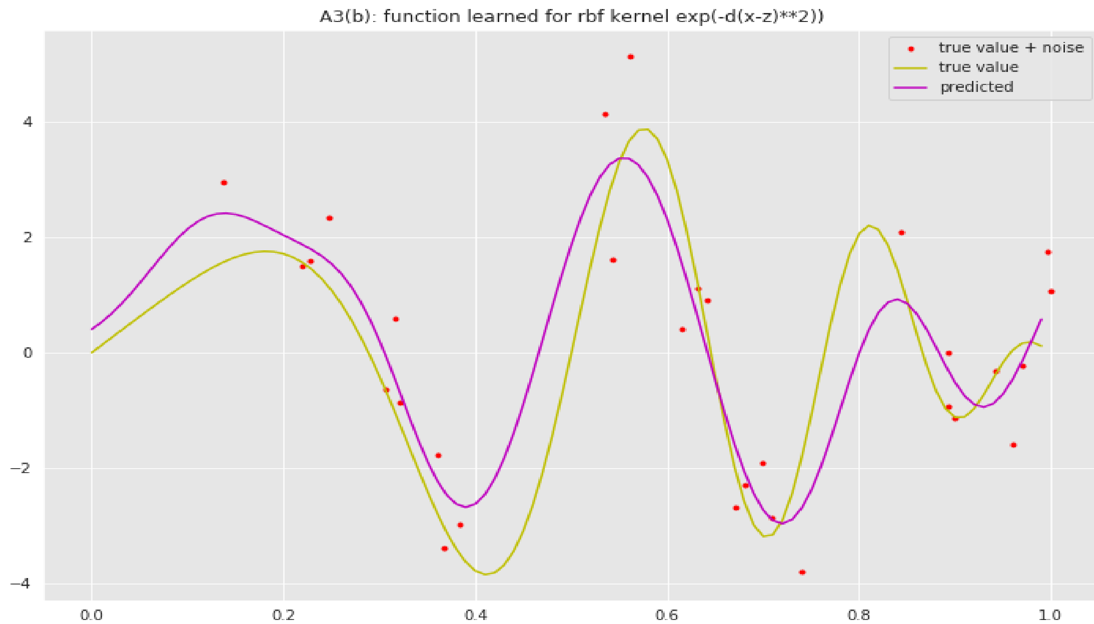
2)For polynomial kernel the hyperparameters are chosen from $d = np.arange(10, 90, 1)$ and $\lambda = np.arange(0.01, .5, .05)$

the optimal lambda value is 0.46, the optimal d value is 45

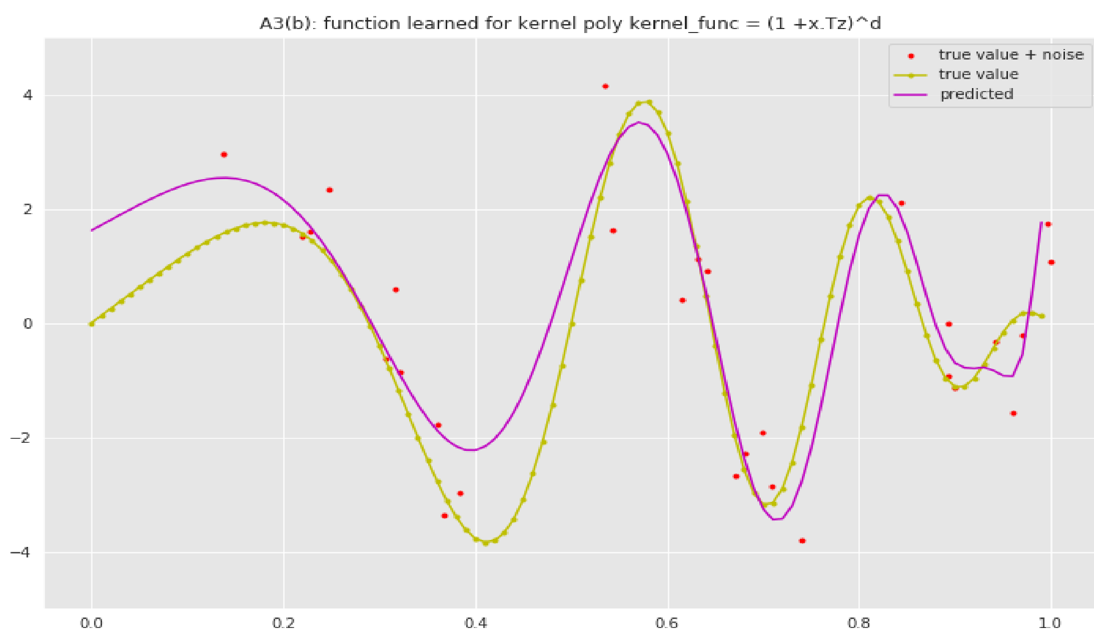
$\square$

Proof. A3b)

1) for RBF kernel, $f(x)$ and $\widehat{f}(x)$ are plotted as follow:



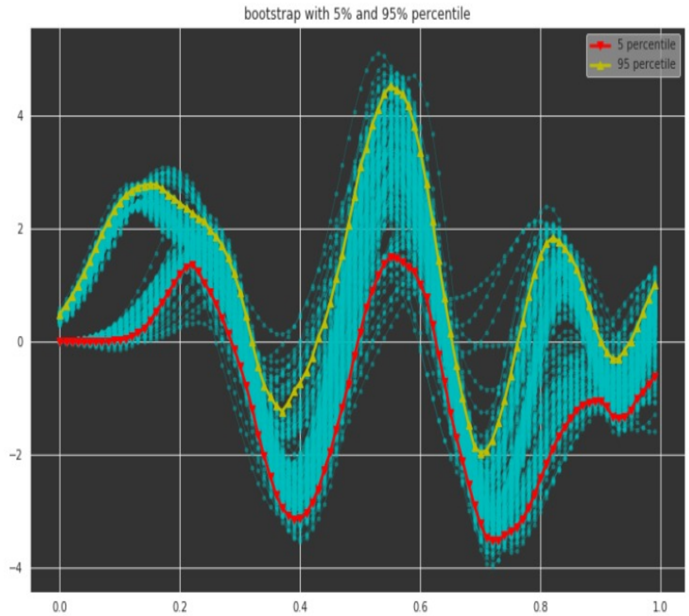
1) for polynomial kernel, $f(x)$ and $\widehat{f}(x)$ are plotted as follow:

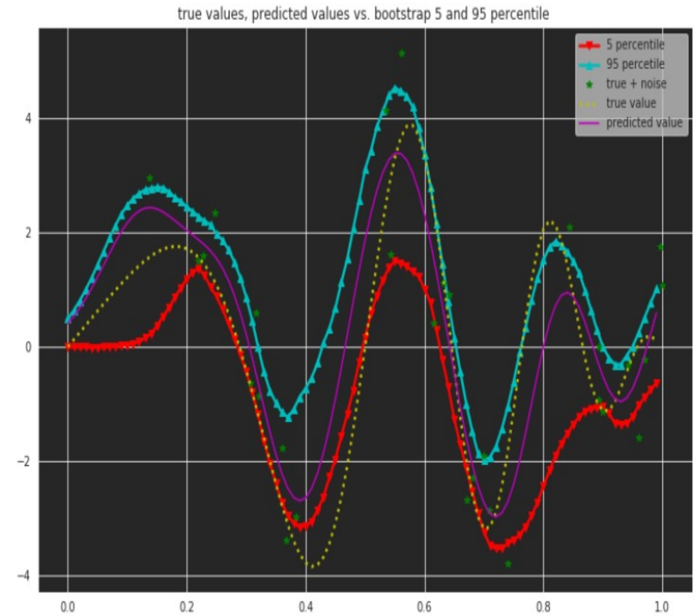


$\square$

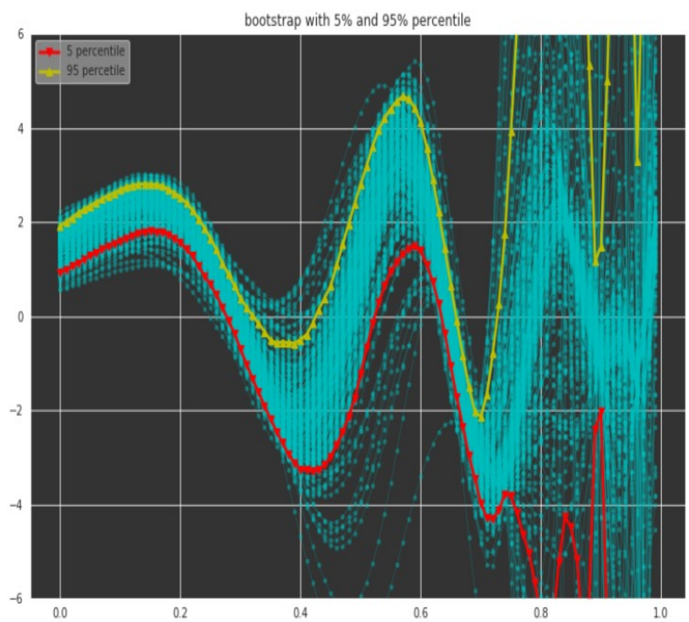
Proof. A3c)

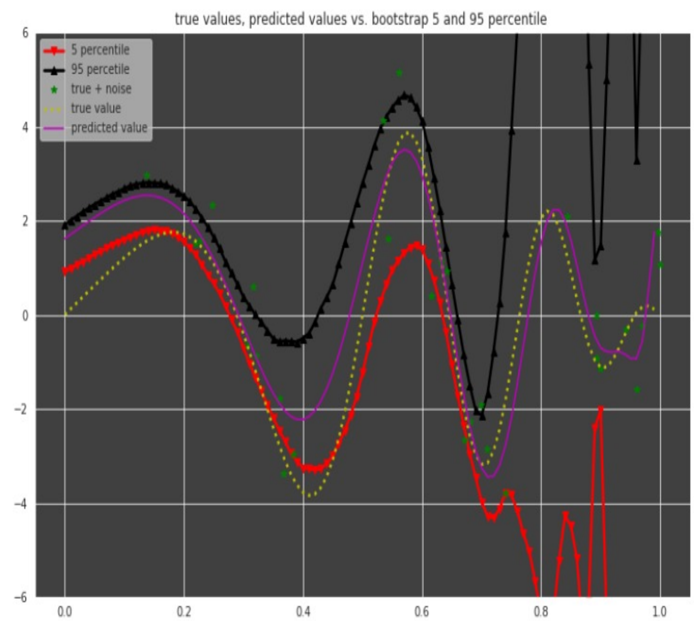
1) for RBF kernel:





1) for polynomial kernel:

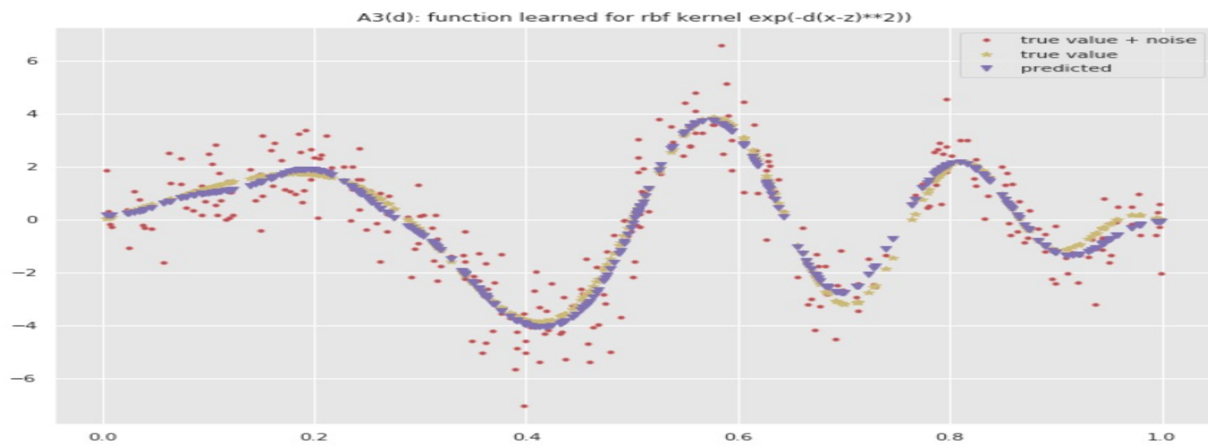


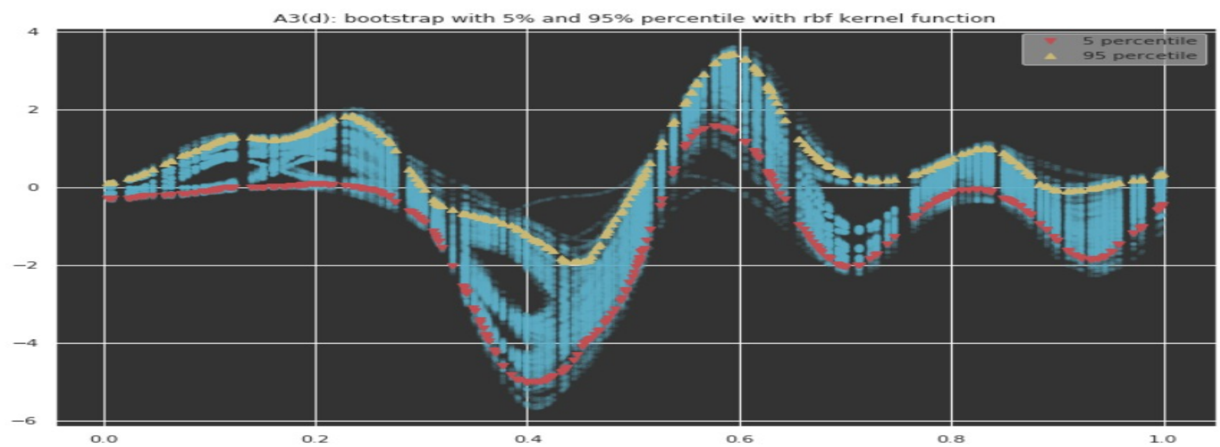


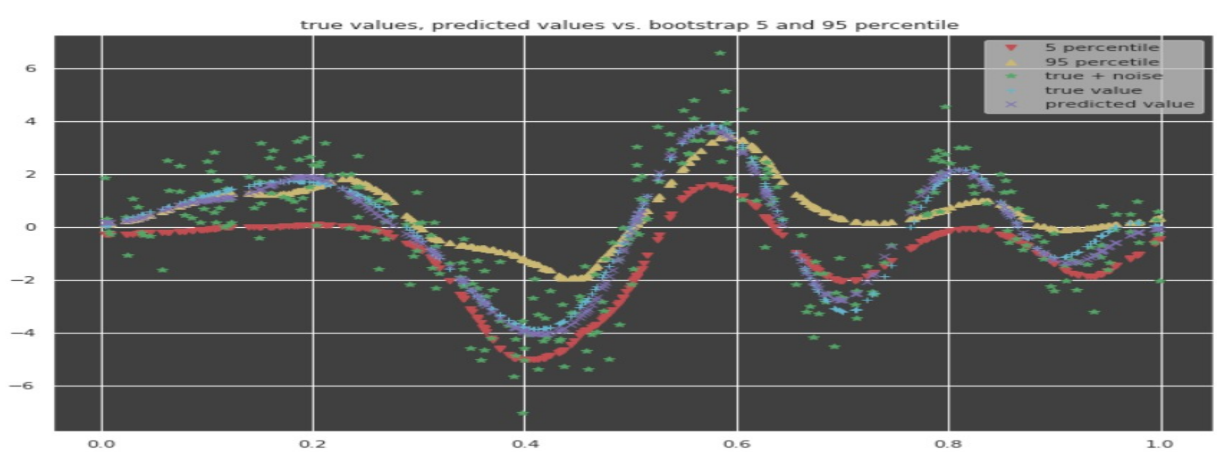
□

Proof. A3d)

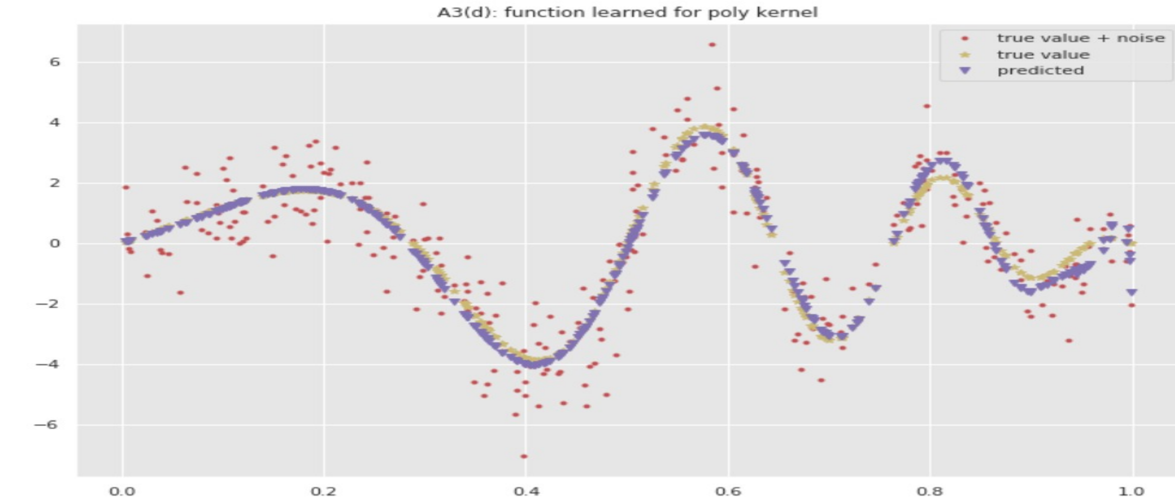
1) RBF kernel:



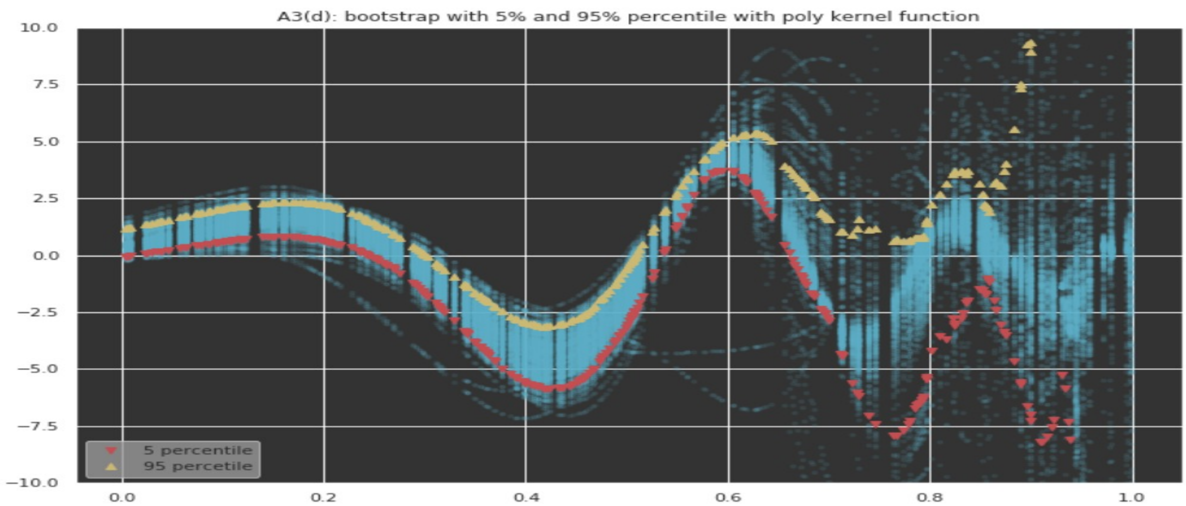




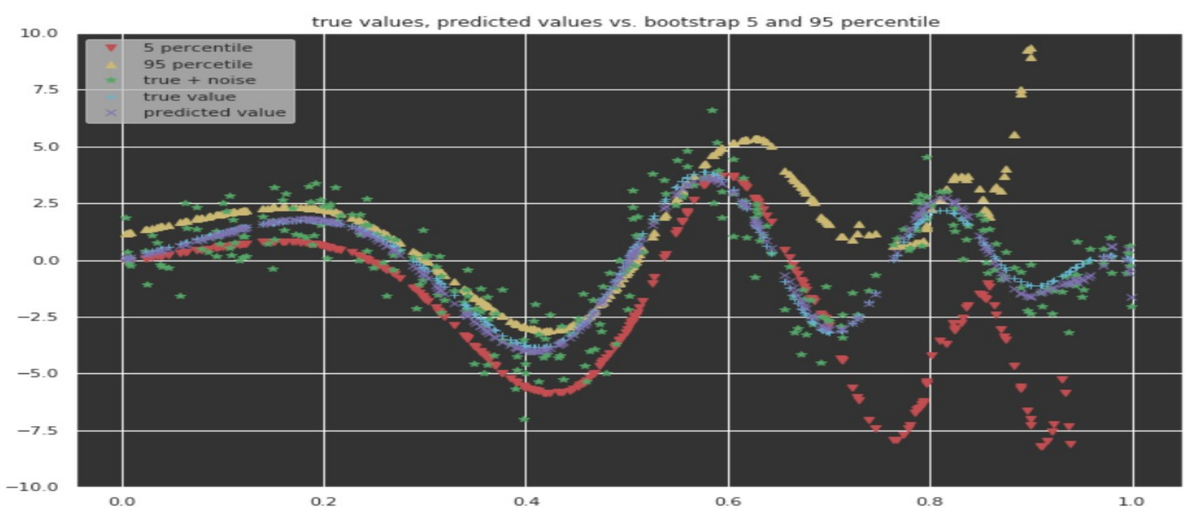
1) Polynomial kernel:



A3(d): function learned for poly kernel
true value + noise
true value
predicted



A3(d): bootstrap with 5% and 95% percentile with poly kernel function
5 percentile
95 percetile



true values, predicted values vs. bootstrap 5 and 95 percentile
5 percentile
95 percetile
true + noise
true value
predicted value

□

Proof. A3e)

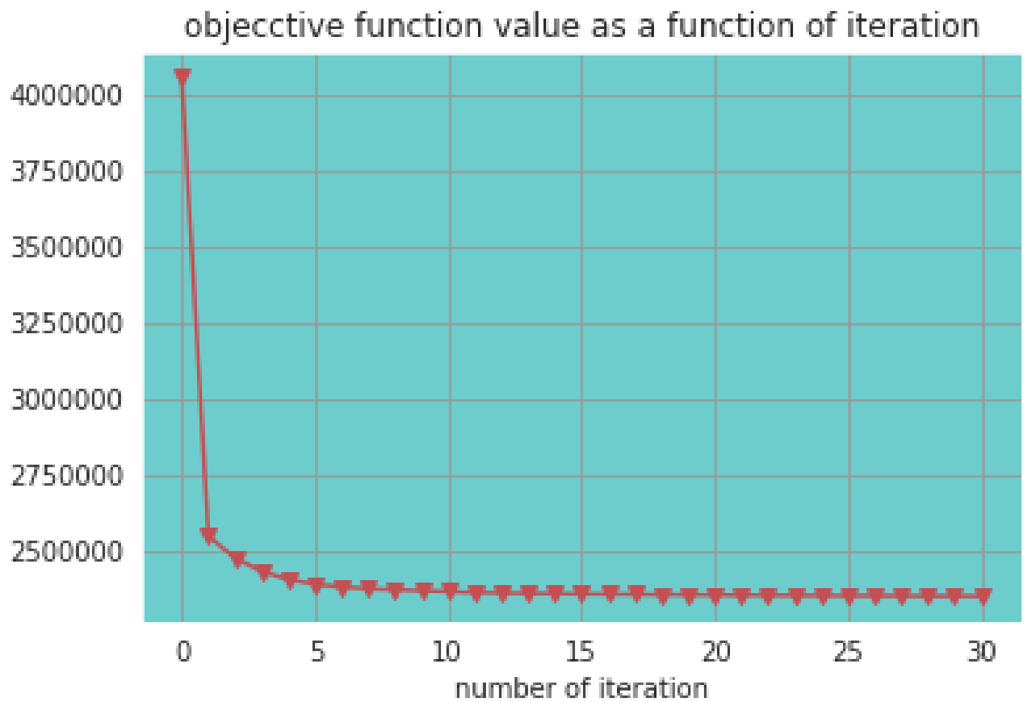
the 5% and 95% percentile is

$$(0.003351860663601623, 0.07656613032802137)$$

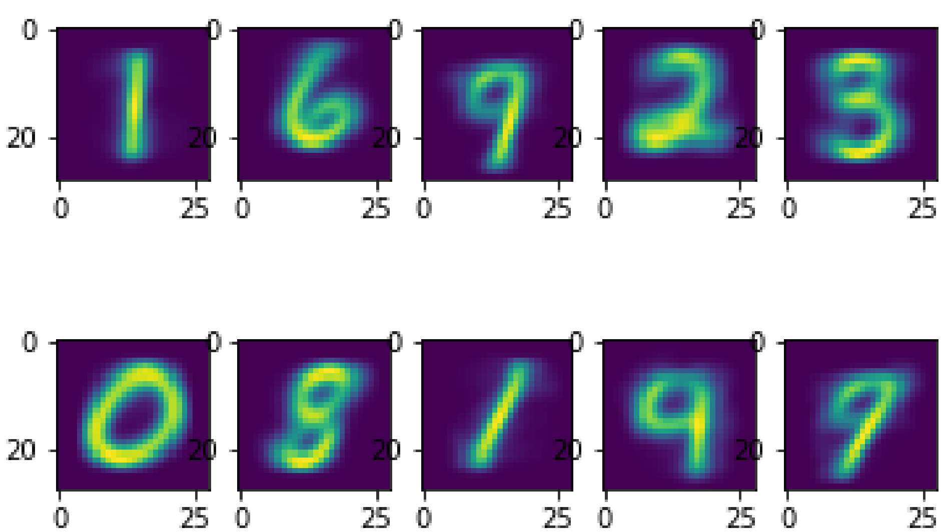
the interval does not contain 0 which suggests that under the current optimal hyper parameters, polynomial has a better performance. However, we can not say for sure that which one of the f_{rbf} or f_{poly} is better, since hyperparamers of one might be tuned better than the other, which means it might not be a fair criteria to judge which one is better. $\square$

Problem A4

Proof. A4b) I randomly chose initial centroids. and the result are as follow:



The final centroids are:



□

Proof. A4c)

The trainning error and testing error as a function of k will be:

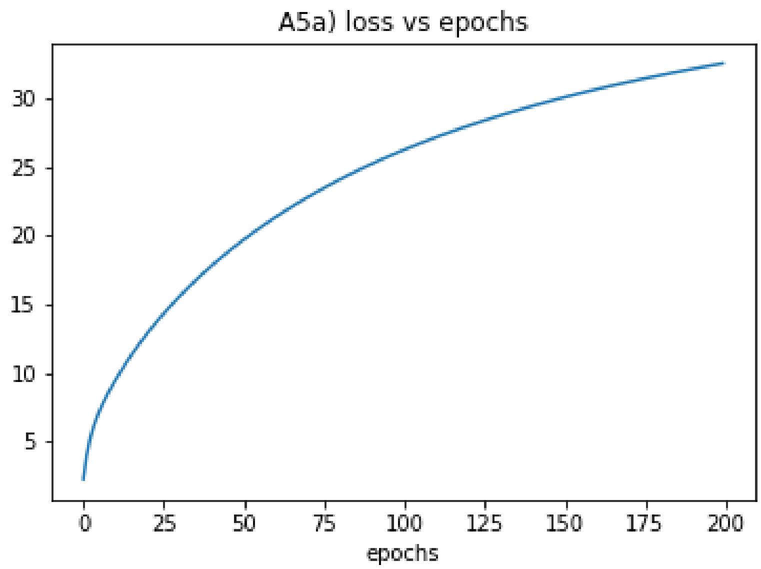
□

Problem A5

Proof. A5a) learning rate is .01

The final training accuracy is 0.99005

The final testing final accuracy is :0.968

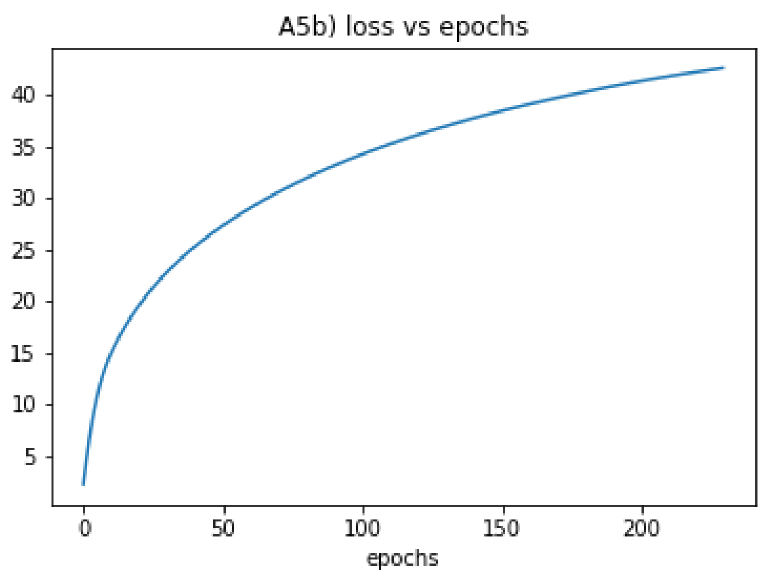


□

Proof. A5b) learning rate is .01

The final training accuracy is 0.9904833333333334

The final testing final accuracy is 0.9595



□

Proof. A5c)
The shallower one takes approximatly 198 iterations to converge, with a learning rate of 0.01. parameters count for each epoch is 784*64 + 64 + 64*10 +10 = 50890.

The deeper one takes approximately 243 iterations to converge with a learning rate of .01. The parameters count for this is 784*32 + 32*32 + 32*10 + 32+32+10 = 26505.

I would say the second one, which is the one with more layers is better. Due to the fact that, it reaches as good accuracy as the first one but is more memory efficient. For every epoach, it stores less coefficient and it reaches to the target accuracy faster. ◻

Problem A6

Proof. A6a)

λ_k represents how much variance of the data are explained in the direction of the k^{th} eignenvector.

For exaple, λ_1 is telling how much variabce is exlpained by the most significant principle component. λ_2 is telling how much variabce is exlpained by the second most significant principle component,....

$\sum_{i=1}^{k} \lambda_i$ means how much variance in the data are explained by the first k eignenvectors. $\square$

Proof. A6b)

It depends on whether X's columns are the observations or rows are the observations.

For this problem, I am defining X to be:

$$X = [x_1, x_2, x_3, ...] = \begin{pmatrix} | & | & | & & | \\ x_1 & x_2 & x_3 & \dots & x_{60000} \\ | & | & | & & | \end{pmatrix}$$

where each column of X is a observation

$$[U, S, V] = svd(X)$$

where U_{d-by-d} and each column of U is the eignenvector. Columns in U, in this case, is the new coordinates.

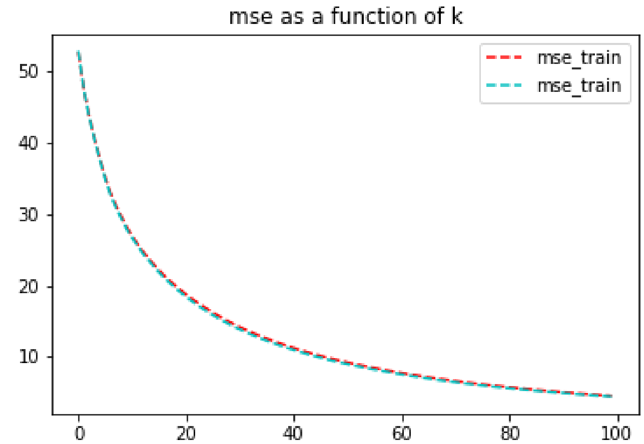
Using the first r vectors to reconstruct:

$$\widehat{Y_r} = U[:, : r](U[:, : r])^T X$$

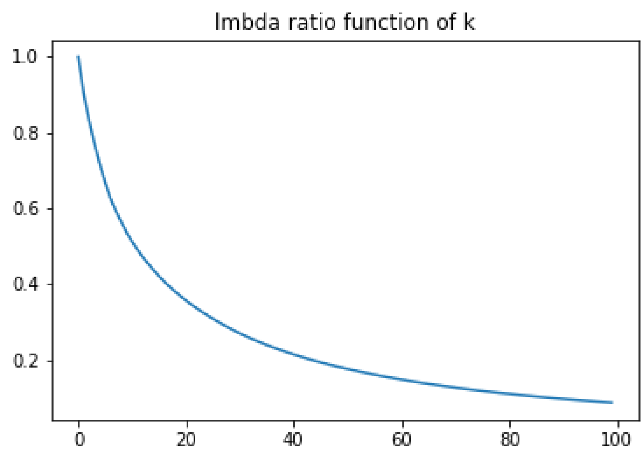
□

Proof. A6c)

MSE as a function of k is plotted as follow:



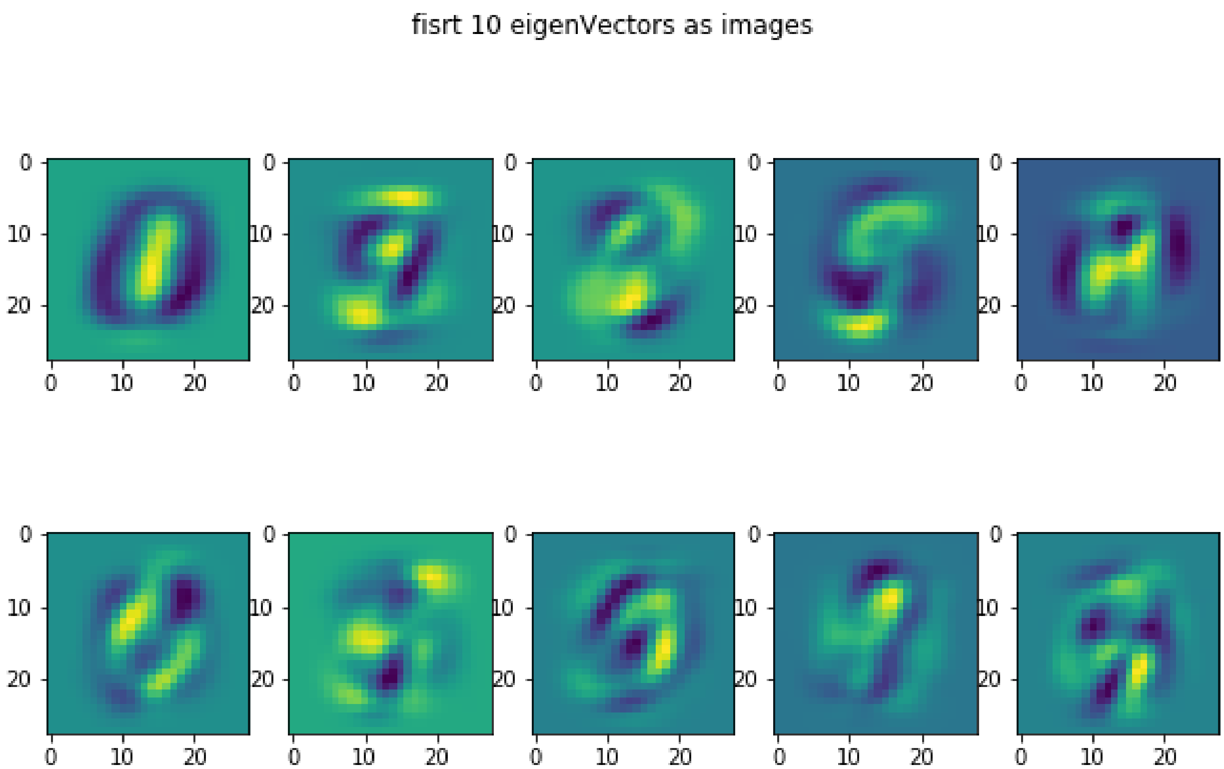
the ratio as a function of k is plotted as follow:



∎

Proof. A6d)

the eigenvectors captures the most prominent/obvious differences of all images. For example, eigenvector that associated with the largest eigenvalue is the images that capetures the pixels which are different the most among all the images.The brighter the pixels are, the more different all the images varies on them.

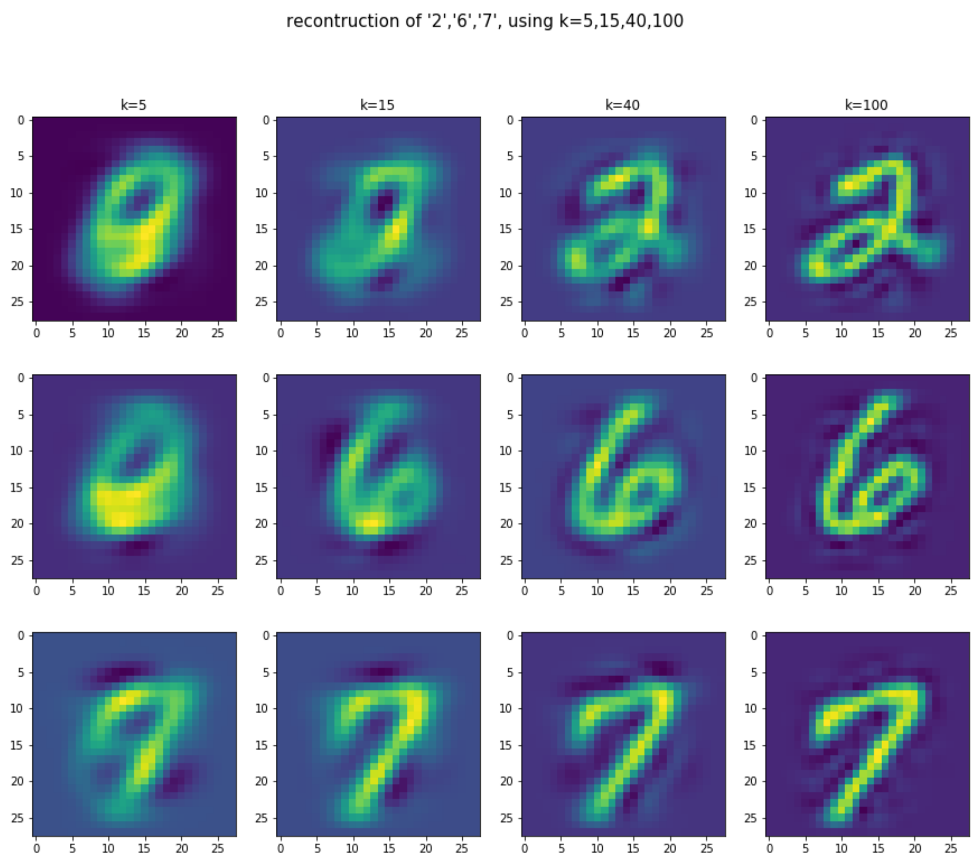


□

Proof. A6e)

As we already known, the eigenvectors in U are othogonal and span R^d, therefore, we have a new coordinates/basis, where we can represent everything in R^d with the linear combination of the set of eigenvectors

We can see that the images in the first column is the reconstructed with only the first eigenvector. the k=15 is reconstructed with the linear combination of the first 15 eigenvectors. Same argument with k=40 and k = 100, the image is the linear combination of the first 40 ir 100 eigenvectors



□

Problem B1

Proof. a)

To show that $P(\widehat{R_n}(f) = 0) \leq e^{-n\sigma}$, First I realize that :

$$\begin{aligned} E_{XY}[1\{f(X) \neq Y\}] &= E_X[E_{Y|X}[1\{f(X) \neq Y\}|X = x]] \\ &= E_{Y|X}[1\{f(X) \neq Y\}|X = x] \\ &= 1 - P(\{f(X) \neq Y\}|X = x) \geq \sigma \end{aligned}$$

which implies that: $P(\{f(X) = Y\}|X = x) \leq 1 - \sigma$

$$\begin{aligned} P(\widehat{R}_n(f) = 0) &= P(\sum_{i=1}^{n} 1\{f(x_i) \neq y_i\}) \\ &= P(\{f(x_1) = y_i\}|X = x_1) * P(\{f(x_2) = y_2\}|X = x_2) * ... * P(\{f(x_n) = y_n\}|X = x_n) \\ &\leq (1 - \sigma)^n \\ &= e^{-n\sigma} \end{aligned}$$

□

Proof. b)

Supposed that the set $\mathcal{F} = \mathcal{A} \cup \mathcal{D}$, where $\mathcal{A} \cap \mathcal{D} = \emptyset$ and $\mathcal{A} = \{f \in \mathcal{F} | R(f) > \sigma, \widehat{R_n}(f) = 0\}$

noticed $\mathcal{A} \subset \mathcal{F} \implies |\mathcal{A}| \leq |\mathcal{F}|$ then our objective:

$$\begin{aligned} &P(f \in \mathcal{F} s.t R(f) > \sigma, \widehat{R_n}(f) = 0) \\ &= P(\mathcal{A}) = P(\{f_1 | f_1 \in \mathcal{A}\} \cup \{f_2 | f_2 \in \mathcal{A}\} \cup ... \cup \{f_k | f_k \in \mathcal{A}\}) \\ &= |\mathcal{A}| e^{-n\sigma} \leq |\mathcal{F}| e^{-n\sigma} \end{aligned}$$

□

Proof. c)

$$\begin{gathered} |\mathcal{F}| e^{-n\sigma} \leq \delta \\ \implies log(e^{-n\sigma}) \leq log(\frac{\delta}{|\mathcal{F}|}) \\ \implies \sigma \geq -\frac{log(\frac{\delta}{|\mathcal{F}|})}{n} \\ \implies \sigma \geq \frac{log(\frac{|\mathcal{F}|}{\delta})}{n} \end{gathered}$$

therefore, $\sigma_{min} = \frac{log(\frac{|\mathcal{F}|}{\delta})}{n}$

$\square$

Proof. B1d)

Let $M = |\mathcal{F}|$, and c to be a number

suppose I have 2 sets $A = \{R(\widehat{f}) < c\}$ and $B = \{R(\widehat{f}) - R(f^*)) < c\}$

I know that $R(f^*) \geq 0$, then

$$(*) = R(\widehat{f}) - R(f^*) \leq R(\widehat{f}) \leq c$$

To prove that $c = \frac{log(M/\delta)}{n}$, I claim that $P(R(\widehat{f}) \leq c) > 1 - \delta$.

This implies that: $P(R(\widehat{f}) > c) \leq 1 - \delta$

As has been proven in part (b),

$$\begin{aligned} P(R(\widehat{f}) > c) =& P(\cup_{i=1}^{M}\{\widehat{R}(f_i) = 0, R(f_i) > c\}) \\ &\leq \sum P(\{\widehat{R}(f_i) = 0, R(f_i) > c\}) \\ &\leq Me^{-nc} = \delta \end{aligned}$$

As proven in part(c),

$$(**) = c_{min} = \frac{log(\frac{|\mathcal{F}|}{\delta})}{n}$$

combining (*) and (**), I know

$$R(\widehat{f}) - R(f^*) \leq R(\widehat{f}) \leq \frac{log(\frac{|\mathcal{F}|}{\delta})}{n}$$

$\square$

Collaborator: Zidan Luo