# Homework 3

Daria Laslo

April 12, 2022

**Exercise 1.** **(a).** In the first part of this exercise, we obtained the rank for optimally applying the SVD imputation. As it can be seen in Figure 1, this method successfully imputes the missing data leading to an image that recovers a substantial amount of the information that was removed from the initial image.


Figure 1: Graphic demonstrating the efficiency of optimised SVD imputation for images (optimal rank = 18). (Left: original image, middle: image after removing 70% of the data, right: imputing information using SVD on the data missing 70%

**(b)(c).** Figure 2 describes the testing errors (mean squared error) for different ranks. We can observe the minimum error is highlighted in orange and obtained for a rank of 18. We can also notice that both a too large or too small rank causes an increase in the testing error. If one of those ranks would have been chosen, the imputation would be of a lower quality as the testing error would be higher and therefore the image would be further away from the original.

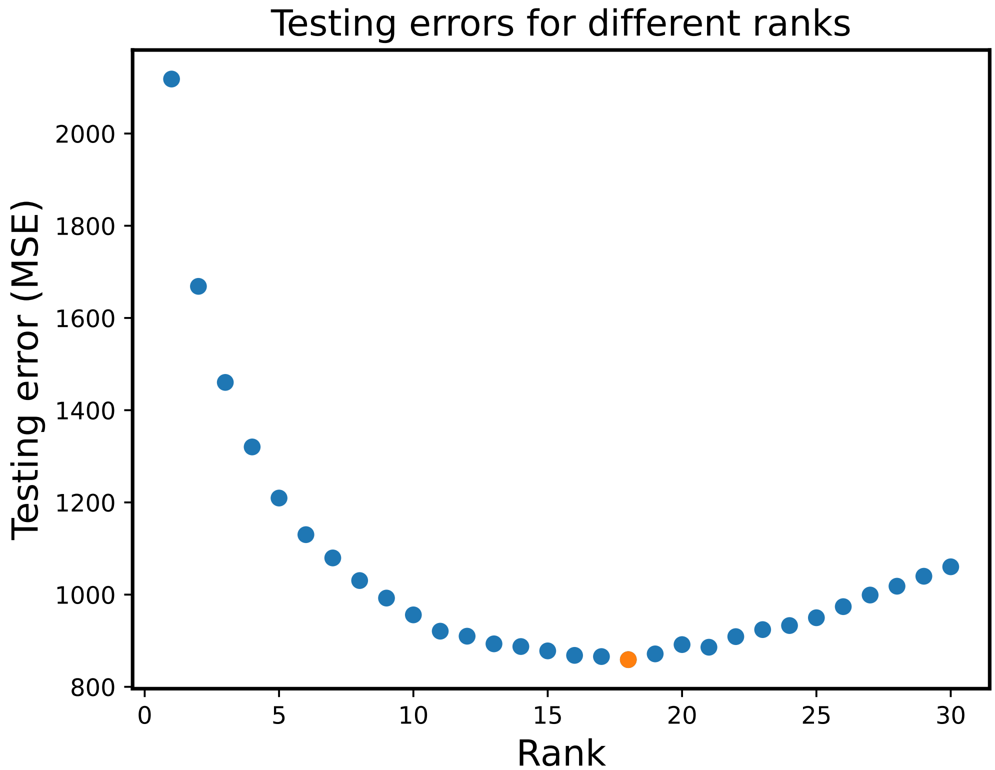


Figure 2: Testing errors for different ranks (optimal rank in orange).

**Exercise 2.** **(a).** After performing PCA on the moons data set, we can observe in Figure 3 that the plot using the two principal components is very close to the original data, only slightly shifted and distorted. Moreover, the two classes are not separable using the transformed data.

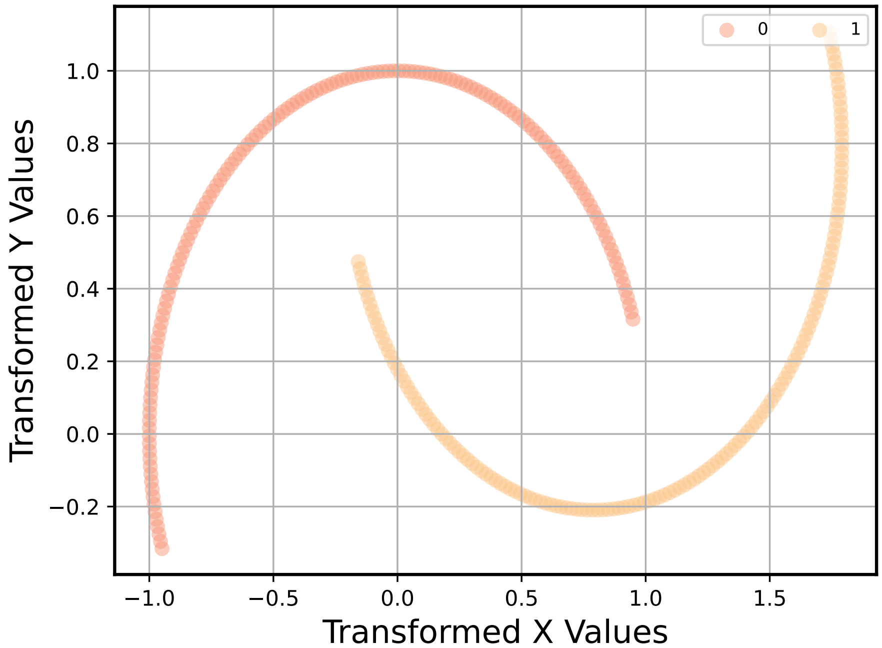


Figure 3: Transformed moons data set using PCA.

**(c).** In Figures 4, 5, 6 and 7 we can see the moons data set being plotted using the transformed data obtained by performing kernel PCA (RBF kernel) using different values for the hyperparameter (gamma = 1, 5, 10, 50). We can observe that even for low values of gamma there is an improvement compared to regular PCA and that as we increase the value of gamma, the classes are starting to slowly separate. For gamma = 50, the two classes are completely linearly separable in the transformed space.

A possible way of finding the optimal value for the hyperparameter is to perform a grid search on a range of values for gamma. This would be similar to what it has been done here where evaluation of the optimality was done be visual inspection. In the case of a classifier we could look at the value of a loss function as a way to understand how the gamma contributes to the separation of the classes. A high value of the loss function would mean that the classes are not separable.

For a more robust approach, this procedure can be applied using k-fold cross validation for all grid values of gamma. Then, the gamma that obtained the lowest value on average for the k-folds is selected as optimal.

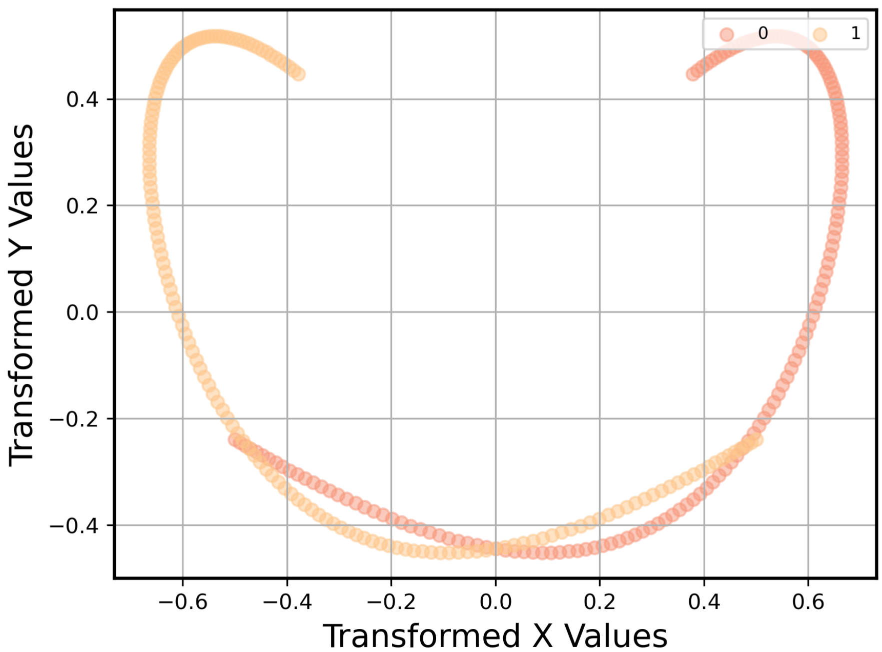


Figure 4: Transformed moons data set using kernel PCA: RBF kernel, gamma = 1.

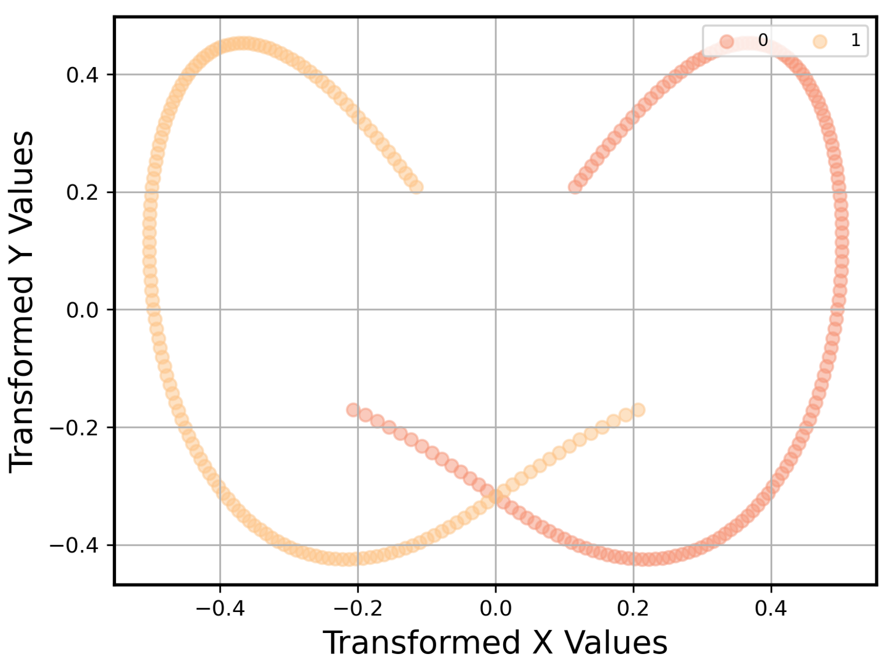


Figure 5: Transformed moons data set using kernel PCA: RBF kernel, gamma = 5.

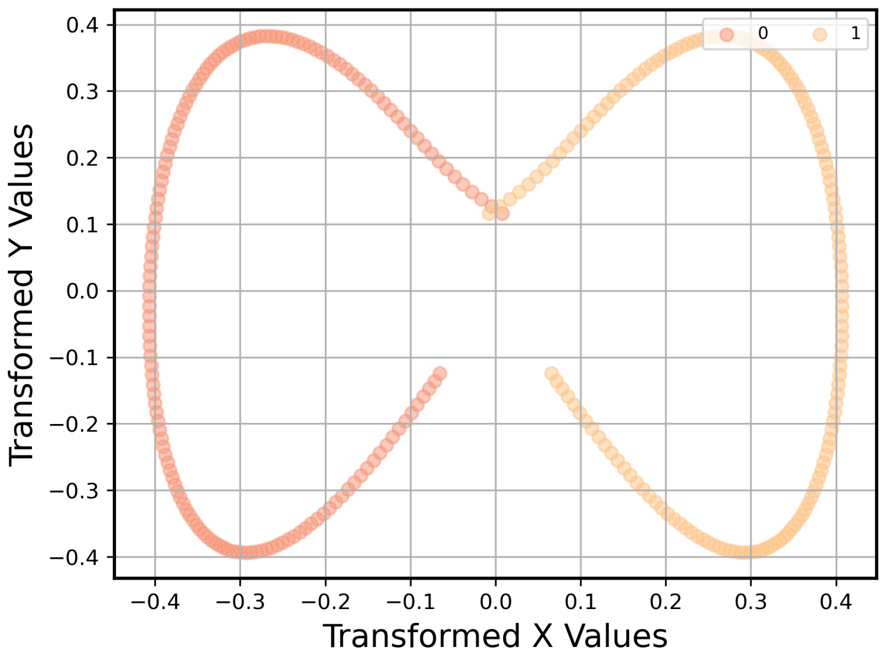


Figure 6: Transformed moons data set using kernel PCA: RBF kernel, gamma = 10.

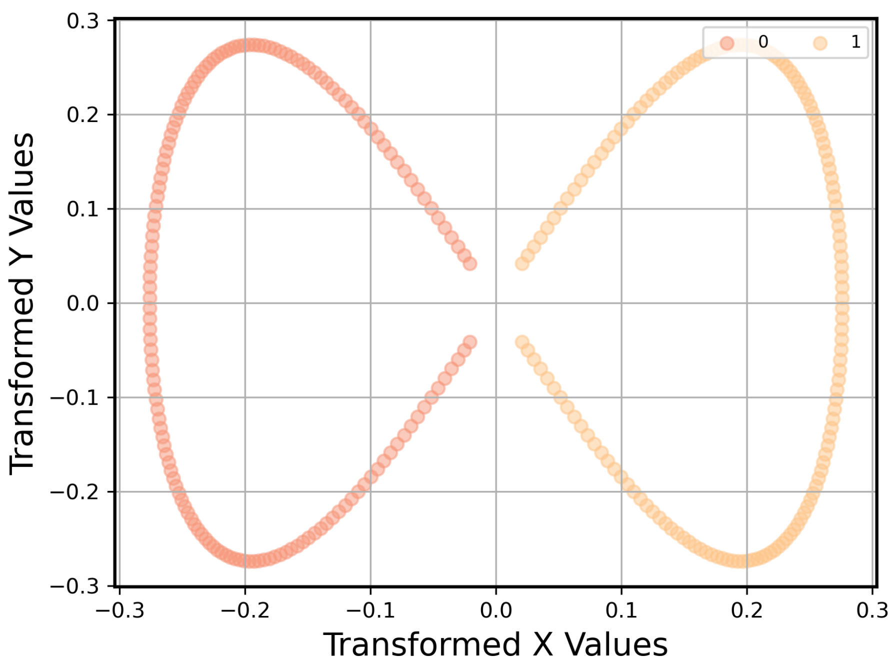


Figure 7: Transformed moons data set using kernel PCA: RBF kernel, gamma = 20.