

CSE543/ECE563: Machine Learning (PG)  
Monsoon 2022

Assignment-3 (115 points)

Release Time: October 08, 2022; 5:00 pm

Submission Time: October 30, 2022 11:59 pm

---

## Instructions

- This assignment should be attempted individually. All questions are compulsory.
  - **Theory.pdf:** For conceptual questions, either a typed or hand-written *.pdf* file of solutions is acceptable.
  - **Code Files:** For programming questions, the use of any one programming language throughout this assignment is acceptable. For python, either *.ipynb* or *.py* file is acceptable. For other programming languages, submit the files accordingly. Make sure the submission is self-complete & replicable i.e., you are able to reproduce your results with the submitted files only.
  - **Regarding Coding Exercises:** You can use modules from sklearn or statsmodels or any similar library for writing the code. Use random seed wherever applicable to retain reproducibility.
  - **Report.pdf:** Create a *.pdf* report of programming questions that contains your applied approach, pre-processing, assumptions, analysis, visualizations, etc.. Anything not in the report will not be evaluated. Alternatively, a well-documented *.ipynb* file with answers to all the questions may be submitted as a part of both code file and report.
  - **File Submission:** Submit a *.zip* named A1\_RollNo.zip (e.g., *A1\_PhD22100.zip*) file containing *Theory.pdf*, *Report.pdf*, and Code files.
  - **Submission Policy:** Turn-in your submission as early as possible to avoid late submissions. Expect No Extensions. Besides, submission within 10 min of the passing of the deadline will incur 20% penalty in the total marks of this assignment. Beyond this, late submissions will not be evaluated and hence will be awarded zero marks.
  - **Resource Constraints:** In any question, if there is a resource constraint in terms of computational capabilities at your end, you are allowed to sub-sample the data (must be stratified). Make sure to exclusively mention the same in the report with proper details about the platform that didn't work for you.
  - **Clarifications:** Symbols have their usual meaning. Assume the missing information. You are free to use any libraries and need not do anything from scratch unless specifically stated otherwise. Use Google Classroom for any queries. In order to keep it fair for all, no email queries will be entertained. You may attend office/TA hours for personal resolutions. No queries will be answered in Google Classroom comments when 12 hours or less are left for the submission deadline.
  - **Compliance:** The questions in this assignment are structured to meet the Course Outcomes CO2, CO3, and CO4, as described in the course directory.
  - **Institute Plagiarism Policy Applicable.** Both programming and theoretical questions will be subjected to strict plagiarism check.
  - There could be multiple ways to approach a question. Please explain your approach briefly in the report.
- 

### 1. Support Vector Classifier

(50 points)

#### (a) Linear SVM Classifier

- i. Load MNIST dataset. After necessary data preparation, build a linear SVM classifier. **(5 points)**
- ii. Show the predictions for first twenty values of your test dataset. Display the confusion matrix. **(4 points)**

- iii. Write your own function to calculate class-wise F1 score. (5 points)
  - iv. Check the F1 scores using sklearn inbuilt function and compare the value with the F1 scores returned by your function written from scratch. Also, report the accuracy. (5 points)
- (b) Non linear SVM: (5 points)
- Build non-linear models with the RBF kernel as well as polynomial kernel. Report the accuracy. (5 points)
- (c) Perform (grid search) cross-validation to find the optimal values of cost C and gamma for SVM classifier using RBF kernel. (6 points)
- (d) Choose the best combination of C and gamma and build the final model with chosen hyperparameters. Display the confusion matrix and report the accuracy of the model. (5 points)
- (e) i. Develop a new training set by extracting the support vectors from the SVM fitted above for the chosen hyperparameters. (5 points)
- ii. Now fit another SVM with the new training set and report the accuracies(train, test). (5 points).
- iii. Compare the accuracies with the previous models. State your observations. (5 points)
2. **Support Vector Regressor** (20 points)

Refer to the file '*SVR.ipynb*' and add code to this starter code file.

Link to '*SVR.ipynb*':

[https://drive.google.com/file/d/1obJlV7QWFpBM2Jic\\_fuPSZGsl57ZseP-/view?usp=sharing](https://drive.google.com/file/d/1obJlV7QWFpBM2Jic_fuPSZGsl57ZseP-/view?usp=sharing)

3. **Theory Question** (10 points)
- (a) Maximize  $f(x, y) = xy$  subject to  $x + y^2 \leq 2$  and  $x, y > 0$  using KKT conditions (7 points)
- (b) **True or False (with justification)** Given a linearly separable data, the margin of the decision boundary produced by SVM will always be greater than or equal to the margin of the decision boundary produced by any other hyperplane that perfectly classifies that data (hyperplane) for the given training dataset. (3 points)

4. **Theory Question** (16 points)

A key benefit of SVM training is the ability to use kernel functions  $K(x, x')$  as opposed to explicit basis functions  $\phi(x)$ . Kernels make it possible to implicitly express large or even infinite dimensional basis features. We do this by computing  $\phi(x)^T \phi(x')$  directly, without ever computing  $\phi(x)$ .

When training SVMs, we begin by computing the kernel matrix  $K$ , over our training data  $\{x_1, \dots, x_n\}$ . The kernel matrix, defined as  $K_{i,i'} = K(x_i, x_{i'})$ , expresses the kernel function applied between all pairs of training points.

The Mercer's theorem tells us that any function  $K$  that yields a positive semi-definite kernel matrix forms a valid kernel, i.e. corresponds to a matrix of dot-products under some basis  $\phi$ . Therefore, instead of using an explicit basis, we can build kernel functions directly that fulfill this property. A particularly nice benefit of this theorem is that it allows us to build more expressive kernels by composition.

In this problem, you are tasked with using Mercer's theorem and the definition of a kernel matrix to prove that the following compositions are valid kernels, assuming  $K^{(1)}$  and  $K^{(2)}$  are valid kernels.

Recall that a positive semi-definite matrix  $\mathbf{K}$  requires  $z^T K z \geq 0, \forall z \in \mathcal{R}^n$

- (a)  $K(x, x') = cK^{(1)}(x, x'); c > 0$  (4 points)
- (b)  $K(x, x') = K^{(1)}(x, x') + K^{(2)}(x, x')$  (4 points)
- (c)  $K(x, x') = f(x)K^{(1)}(x, x')f(x')$ ; Where  $f$  is any function from  $\mathcal{R}^m$  to  $\mathcal{R}$  (4 points)
- (d)  $K(x, x') = K^{(1)}(x, x')K^{(2)}(x, x')$  (4 points)

Hint: Use the property that for any  $\phi(x), K(x, x') = \phi(x)^T \phi(x')$  forms a positive semi-definite kernel matrix.

5. **Theory Question** (19 points)

Consider the following dataset that has 3 points in 1 D.

(class)	$x$
+	0
-	-1
-	+1

- (a) Are the classes  $\{+, -\}$  linearly separable? **(1 point)**
- (b) Map each point to 3-D using new feature vectors  $\phi(x) = [1, \sqrt{2}x, x^2]^T$ . Are the classes now linearly separable? If yes, find a separating hyperplane. **(3 points)**
- (c) Define a class variable  $y_i \in \{1, +1\}$  which denotes the class of  $x_i$  and let  $w = (w_1, w_2, w_3)^T$ . The max-margin SVM classifier solves the following problem.

$$\min_{w,b} \frac{1}{2} \|w\|_2^2 \quad s.t.$$

$$y_i(w^T \phi(x_i) + b) \geq 1, \quad i = 1, 2, 3$$

Using the method of Lagrange multipliers show that the solution is  $\hat{w} = (0, 0, -2)^T$ ,  $b = 1$  and the margin is  $\frac{1}{\|\hat{w}\|_2}$ . **(8 points)**

- (d) Show that the solution remains the same if the constraints are changed to

$$y_i(w^T \phi(x_i) + b) \geq \rho, \quad i = 1, 2, 3$$

for any  $\rho \geq 1$ .

**(3 points)**

- (e) Is your answer to (d) also true for any dataset and  $\rho \geq 1$ ? Provide a counter-example or give a short proof. **(4 points)**