Assignment 1
Classical Machine Learning methods

Part 1: Regressing Problems

Note:

1. Solve all the problems using MATLAB or any other computer language. However, try by hand problem 1.
2. After calculating the best model given your data, review the provided data and you may remove any possible outliers (**a value with a possible error or high noise**), then recalculate the model for cleaned data. Compare the two models.
3. In all your answers to each question, write down the equations of your solutions (after calculating their parameters).
4. Hint: If the gain in R^2 is not high enough, it is better to take the model with a lower number of parameters.

1. The numbers of insured persons "y" with an insurance company for the years 1987 to 1996 are shown in the table.

Year	1987	1988	1989	1990	1991	1992	1993	1994	1995	1996
y	13300	12400	10900	10100	10150	10000	800	9000	8750	8100

Make a scatterplot of the data, letting x represent the number of years since 1987.

a) Fit linear, quadratic, and cubic, by comparing the values of R^2. Determine the function that best fits the data. (Hint: take care of note 4 above)

b) You may review the data and remove what is outside the reasonable range (outlier), then recalculate the results and compare.

c) Graph the function of best fit with the scatterplot of the data.

d) With the best function found in part (b), predict the average number of insured persons in 1997.

2. Develop a model for estimating heating oil used for a single-family home in January based on average temperature and amount of insulation in inches.

Oil	Temp F	Insulation
270	40	4
362	27	4
162	40	10
45	73	6
91	65	7
233	65	40
372	10	6
305	9	10
234	24	10
122	65	4
25	66	10
210	41	6
450	22	4
325	40	4
52	60	10

a) Fit linear, and quadratic functions to the data. By comparing the values R^2, determine the function that best fits the data.

b) Review the data and remove what is outside the reasonable range (outlier), then recalculate the results and compare.

c) Then use the regression models for the functions in b to predict the needed oil if the temperature is 15 Fahrenheit and the insulation is 5 attic insulations inches.

Part 2: some classical ML modeling

Use the ReducedMNIST which is a reduced version of the MNIST data set.

- **ReducedMNIST training**: 1000 examples for each digit.
- **ReducedMNIST test**: 200 examples for each digit.

1. Use the ReducedMNIST data to generate these features for each of the images of the training and testing sets:
    a. DCT features (225 dimensions)
    b. PCA (use several dimensions so that the total variance is at least 95% of the total variance when using all the 784 dimensions).
2. Then train these classifiers using the training set of the **ReducedMNIST** data for each of the above features:
    a. K-means for each class. Try 1, 4, 16, and 32 clusters for each class.
    b. SVM. You may try the linear and the nonlinear kernels (like RBF). In this case **state clearly**, what kernel have you used.

Then use the resulting models to classify the test set. Compare the different features and the different classifiers:
You must add a final table to summarize all your results (accuracy and processing time) comparatively, as in the table that is shown below.

1. **Only for the best result of each classifier show it in a confusion matrix among the 10 digits.**
2. **Add your conclusions.**

		Features			
		DCT		PCA	
		Accuracy	Processing Time	Accuracy	Processing Time
Classifier					
K-means Clustering	1				
	4				
	16				
	32				
SVM	Linear				
	nonlinear*				

* Mention the kernel name and its specs

Part 3: ways to fasten the Labeling Process

Problem Statement

Manual data labeling is both costly and time-consuming. This assignment explores methods to reduce the amount of manual effort required, thereby decreasing the overall labeling cost and time.

For the "ReducedMNIST training" dataset, first generate a version with no labels. Then, design a pipeline - a series of steps - that automatically labels the data with minimal human involvement. The dataset contains 10,000 images (10 classes with 1,000 examples each). Below are two example pipelines that can help reduce manual labeling efforts. You are free to modify these pipelines; if you do, please clearly document your changes.

Pipeline 1:

1. **Clustering and Majority Vote:**
    - Cluster the 10,000 training images into many clusters (for example, 100 or more clusters).
    - For each cluster, randomly sample 5 images and determine the majority class among them.
    - Label all images in the cluster with the majority class.
2. **Initial Classification:**
    - Develop an SVM classifier to automatically classify all images.
3. **Refinement via Clustering:**
    - Use the output from the SVM classifier to generate improved clusters.
4. **Iteration and Reporting:**
    - Repeat steps 2 and 3 until no significant improvement is observed.
    - Report the final classification accuracy and estimate the total human time spent verifying the labels (assume that checking the class of one image takes 10 seconds).

Pipeline 2:

1. **Random Sampling:**
    - Randomly select a small set of examples from each class (e.g., 40 examples per class).
2. **Data Augmentation:**
    - Augment the selected images by applying transformations such as small rotations (e.g., 5° to the right, then 5° to the left), adding noise, or shifting the digit within the image (up, down, left, or right).
3. **Initial Model Training:**
    - Train an SVM model (SVM-1) using the combination of the randomly selected images and the augmented data.

   - Use SVM-1 to classify all the images in the dataset.

4. **Iterative Refinement:**
   - Train a new SVM model (SVM-2) using the data classified by SVM-1.
   - Repeat this process until further iterations do not improve the model's performance.
5. **Evaluation:**
   - Report the final classification accuracy of the model.
   - Compare the performance and estimated labeling time of your pipeline with that of a model trained on the full dataset (1,000 examples per class).
   - (Hint: Use the "ReducedMNIST test" dataset for accuracy evaluation, and estimate the total time to label 10,000 images by assuming 10 seconds per image evaluation)

Pipeline 1	Manual time	Accuracy
iteration-1 results		
iteration-2 results		
iteration-3 results		
Pipeline 2		
iteration-1 results		
iteration-2 results		
iteration-3 results		

Pipeline 3 (Optional Bonus):

Alternative Pipeline for Reducing Manual Labeling

You are free to propose any additional pipeline that minimizes manual work while maintaining high labeling accuracy. For example, you might design a pipeline that leverages active learning, or semi-supervised learning, or even consult a large language model (LLM) for suggestions. In your submission, clearly outline your pipeline's steps, explain how it reduces human effort, and provide empirical evidence of its effectiveness.