Problem 1. (KNN, cross-validation [15 points])
A political campaign is on the way for the upcoming election between Party A and Party B in city Z. Party A leader has decided to use machine learning to predict the current trend, based on which it will further adjust its advertisement. To this end, it has surveyed 600 people in the city, recording the coordinate of their home (latitude and longitude), and obtaining their political preference as either A or B. It then has attempted to use the KNN (k-nearest neighbor) approach to predict the potential outcome of the election by estimating the preference for the rest of the residents, having knowledge of the voters' home coordinates.

1. What are the features of the problem? (2 point)

2. Consider the problem of predicting political preference of a new individual based on the above data. Is this a supervised or an unsupervised machine learning problem? Justify your answer. (1 point)

3. Write the formula for standardizing the first feature vector to zero-mean and unit variance. (2 point)

4. You are given a new person's latitude and longitude (x_1, x_2).
   (a) Write the steps for determining the label of this data point based on $K = 3$ neighbors and using the Euclidean distance (3 point).
   (b) Specify which step would change and how if you use the Manhattan distance instead of the Euclidean distance. (1 point)

Party A wants to choose a good K for its prediction. It thus has divided the data into 500 training points and 100 test points. Then, it has done 5-fold cross validation for the values of $K \in \{1, 3, 5, 7, 9\}$. The following table summarizes the resulting mean and variance of the cross validation error.

model	mean cross-validation error	variance
K=1	0.43	14.23
K=3	0.33	11.50
K=5	0.29	10.41
K=7	0.30	12.23
K=9	0.39	15.44

5. Let $\hat{y}^i$ be the predicted label for data (x_1^i, x_2^i) , y^i its true label, and $\mathbf{1}(\hat{y}^i = y^i)$ be the indicator function. Write the formula for the mean (over the 5 folds) validation error rate for $K = 3$. (2 point)

6. For which K is it most likely that the test error will be similar to the validation error, and why? (2 point)

7. Based on KNN classification, how can Party A know in which region it should improve its advertising campaign? (2 point)

Problem 2. (Feature vectors and linear regression [12 **points**])

You have recorded the signal corresponding to the mechanical vibration of a turbine for $1, 2, \ldots, T$ time steps as $(y^1, y^2, \ldots, y^T)$, where $y^t \in \mathbb{R}$. You believe that under normal turbine operating conditions, the signal is periodic. In particular, it is approximately the sum of K sinusoidal signals:

$$y^t \approx \sum_{k=1}^{K} a_k \cos(\omega_k t - \phi_k),$$

where ω_k's are known but a_k and ϕ_k are unknown, for $k = 1, 2, \ldots, K$. Based on the observations $\{y^t\}_{t=1}^T$ you want to estimate a_k, ϕ_k so as to minimize the mean square error between the prediction $\hat{y}^t = \sum_{k=1}^K a_k \cos(\omega_k t - \phi_k)$ and the true value y^t.

1. How many parameters do you need to identify for your approximation? (1 point)

2. Recall the fact that for a sinusoid signal we have:

   $$a\cos(\omega t - \phi) = \alpha \cos(\omega t) + \beta \sin(\omega t),$$

   where $\alpha = a\cos\phi$, , $\beta = a\sin(\phi)$. Based on this fact, formulate your predictor $\hat{y} = \sum_{k=1}^K a_k \cos(\omega_k t - \phi_k)$ as a linear function of the parameters α_k, β_k, $k = 1, 2, \ldots, K$, corresponding to each sinusoid. (3 point).

3. Formulate the mean-square-error loss function. (2 point)

4. Derive the gradient of the loss function with respect to the linear regression parameters α_k, β_k, $k = 1, 2, \ldots, K$. (4 point)

5. Provide an approach to compute the optimal set of parameters without using gradient descent. (2 point)

Problem 3. (Dimensionality reduction using PCA [15 **points**])
Consider a dataset $\{x^i\}_{i=1}^N$, where $x^i \in \mathbb{R}^2$. *Assume the feature vectors are centered so each feature vector has zero mean.* Let σ_1^2, σ_2^2 denote the variance of features 1 and 2, respectively. Suppose that the second feature is a constant multiple of the first feature. So, for any data point $x^i \in \mathbb{R}^2$, we can write $x_2^i = \alpha x_1^i$, where $\alpha \in \mathbb{R}$ is an unknown constant, here assumed to be positive[1].

Here, we derive the visually obvious fact that the PCA projects the data onto the line $x_2 = cx_1$.

1. Derive the variance of $\{x_2^i\}_{i=1}^N$ denoted by σ_2^2, in terms of α and σ_1^2. (2 point)
   Using the variance calculation, show that the correlation between the two features is 1 irrespective of value of α. (2 point).
2. Derive the matrix $C = X^T X \in \mathbb{R}^{2\times 2}$ in terms of σ_1^2 and α. (1 point)
3. Let $c_1, c_2 \in \mathbb{R}^2$ denote the columns of C. Verify that $c_2 = \alpha c_1$. (1 point)
4. Using the result of Part 3, show that the vector $(1, \alpha) \in \mathbb{R}^2$ is an eigenvector of the matrix C and derive its corresponding eigenvalue. (2 point)
5. Using the result of Part 3, show that the vector $(-\alpha, 1) \in \mathbb{R}^2$ is an eigenvector of the matrix C corresponding to an eigenvalue at 0. (1 point)
6. Scale the first principal component of C to have a unit Euclidean norm. (1 point)
7. The projection of the data matrix $X \in \mathbb{R}^{N\times 2}$ onto its first principal component is given by $X\Theta \in \mathbb{R}^N$. Write what Θ is in this problem. (1 point)
8. Verify that the projection of each data point (x_1^i, x_2^i) onto the first principal component gives the Euclidean distance of (x_1^i, x_2^i) from the origin $(0, 0)$ and draw a visualization of the result. (3 point).
9. What would look different in your projection if you first standardize the features to unit variance before PCA computations above? (1 point)

[1] In practice, due to noisy measurements, x_2 would only approximately be a constant multiple of x_1 in the dataset. Here, to simplify the computations we discard the impact of the noise and assume to have access to "perfect" data.

Problem 4. (Neural networks [13 points])

1. You want to construct a neural network that takes two binary inputs $x_1, x_2 \in \{0, 1\}$ and gives the output $y = 0$ if $x_1 = x_2 = 0$ and $y = 1$, otherwise. In other words, the neural network is implementing the logical "OR" function.

   Construct this network using a single hidden layer, with one node in the hidden layer and the unit step activation function, that is, $\sigma(z) = 1$ if $z \geq 0$ and $\sigma(z) = 0$, otherwise. Draw your network and specify its weight and bias values. (5 point)

Now, we consider a robotic obstacle detection problem. You aim to control a robot car so that it can maneuver in a room while avoiding obstacles. Thus, you first want to design a neural network classifier that takes a camera image and labels objects identified as *obstacle* or *free space*. The car camera produces images of size 240×240 pixels over three $\{red, green, blue\}$ channels. For training your neural network, you take 200 images and manually label the objects in the image as *obstacle* or *free space*.

2. In your first attempt, you use a neural network with 4 hidden layers, where each hidden layer has 10 nodes and an output layer with 1 node. You use a *tanh* activation function and the logistic loss function for the output layer. You flatten each image before giving it to your neural network as an input. For the *first* layer of the network, how many weights and biases need to be determined? (2 point)

You find that the accuracy of your classifier above is not very good. So, you decide to use a pre-trained network and simply modify it for your task. You use the EfficientNet network, developed by Google, which is a convolutional neural network for classifying images into 1000 classes. The original network has 7.8 million parameters to train, and thankfully, the network has been trained already. The last layer of EfficientNet has 1000 outputs and the *softmax* activation function. You simply append one layer with a single output node to the last layer of EfficientNet, and you add no activation function to this node.

3. You use the logistic loss function for the last layer you added. Write the logistic loss function in terms of the output of the last layer of the EfficientNet and your newly added parameters. (2 point)

4. Do you expect to be able to find the optimal parameters for the above problem using gradient descent? Justify your answer.s (1 point)

5. You observe the training loss oscillating throughout the iterations of the gradient descent. Explain why a decrease in the learning rate could result in a monotone decrease of the loss. (1 point)

6. Suppose that a filter of size 3×3 is used for convolution in the first layer and on the red channel of EfficientNet. The filter parameters are as shown in the matrix F below. Consider a 3×3 segment of the image over the channel *red* as shown in the matrix S below. What is the result of convolution of F with S? (2 point)

$$F = \begin{pmatrix} -1/4 & 0 & 1/4 \\ 0 & 0 & 0 \\ 1/4 & 0 & -1/4 \end{pmatrix}, \qquad S = \begin{pmatrix} 0.3 & 0.1 & 0.2 \\ 0.4 & .01 & 0.2 \\ 0.4 & 0.2 & 0.3 \end{pmatrix}$$

Problem 5. (Probabilistic classification [15 **points**])
At last, you understood decision trees and decided to make a youtube video to explain it to all those students who might be confused. Your video is posted and you are receiving many comments. You want to design an automatic way of analyzing the "sentiment" of the comments, that is, to understand if the watchers found your video helpful (positive +) or unhelpful (negative -).

So, you classify a few comments as below, and plan to let the machine learning algorithms do the analysis for the rest of the comments.

class label	comment
+	clear and engaging
-	confusing but funny
-	boring and many typos
+	engaging and funny
+	highly helpful and clear

Note: For the three points below, ignore the common word "and". Hence, the 5 comments above contain 10 distinct words.

1. Write the prior probability of positive and negative classes. (1 point)
2. Write the probability of the word "funny" given class positive and the probability of the word "funny" given class negative (2 point).
3. Write the probability of "helpful and funny" given class positive using the Naive Bayes assumption. (1 points)

Now, suppose that you use 100 comments for training so that you encounter many more positive and negative comments and many more distinct words.

4. Your Naive Bayes approach classifies 3 of the positive comments as negative and 7 of the negative comments as positive. What is its error rate? (1 point)
5. Your classifier has much more false positives than false negatives. Since you care more about the critical comments to improve your work, you decrease the *prior* probability of class positive. Show that this change could reduce the number of false positives. (2 point)

Suppose that you now replaced your Naive Bayes classifier with a decision tree, trained on the same 100 comments above and using the gini criterion. You found that the presence of the feature "helpful" had the lowest gini index. In particular, all the 20 comments that had the word "helpful" in them were of class positive. For the remaining 80 comments, you found that the word "funny" had the lowest gini index. In particular, 20 of the 80 comments did have the word "funny", and of these 20, 15 were of class positive; whereas 60 of the 80 comments did not have the word "funny" and of these 60, 20 were of class positive. Now, you stopped growing your tree, at depth 2.

6. Draw your decision tree. (2 point)
7. What was the gini index of splitting the "not helpful" node into a leaf node corresponding to "funny" and "not funny"? (2 point)
8. What is the false positive and false negative rate of this depth 2 decision tree? (2 point)
9. Comment on the advantages and disadvantages of the two classifiers above for this problem. (2 point)