1. In each of the following scenarios, determine whether supervised, unsupervised, semi-supervised, or reinforcement learning is needed (18 pts).

   (a) We have a dataset of images. There are more than one million distinct colors in the images. We wish to find 256 colors that represent those colors the best.

   (b) We have a large corpus of webpages labeled by their topics. We also used a crawler to download a large number of webpages from the internet. We whish to use both datasets to build a machine learning algorithm that determines the topic of a webpage.

   (c) We have a dataset of photo IDs. We want to build a machine learning algorithm that estimates the age of a person based on their ID portrait.

   (d) We wish to build a vacuum cleaner that learns to clean a room and avoid obstacles.

   (e) We have electronic health records of 2,000,000 patients. Each patient has 200 numeric features. We wish to build an algorithm that summarizes the numeric features of each patient into 20 features.

   (f) We have a dataset of insurance claims that includes the time each claim has been processed. We want to build a model that predicts insurance claim processing time.

2. In the following, determine regression and classification problems (15 pts):

   (a) Separating seabass and salmon based on their lightness and length in a food factory.

   (b) Estimating the price of a house based on features such as location, number of bedrooms, etc.

   (c) Diagnosis of diabetes based on electronic health records.

   (d) Determining the MPG of a car based on its specifications.

   (e) Determining the Myers-Briggs personality type [1] of a person based on their writing style.

3. Explain if the following cases defy the no free lunch theorem (20 pts).

   (a) Five different algorithms perform approximately similar on a data set on pregnancy diabetes.

   (b) A special type of classifier (called the Naïve Bayes Classifier) often works particularly well with text data.

4. A Machine Learning engineer is designing an expert system for simple diagnosis tasks. The data set contains information about persons with a number of features describing their symptoms and the labels are the diagnosis. The data set contains the seven cases provided in the table below.

[1] Research what it is, if you like!

Person	Fever	Vomiting	Diarrhea	Shivering	Classification
1	No	No	No	No	Healthy
2	Low	No	No	No	Flu
3	High	No	No	Yes	Flu
4	High	Yes	Yes	No	Food Poisoning
5	Low	No	Yes	No	Food Poisoning
6	No	Yes	Yes	No	Stomach Flu
7	Low	Yes	Yes	No	Stomach Flu

Sometimes, instead of a distance measure between two instances, we use a similarity measure between two instances. The higher the similarity between two instances, the lower the distance between them. The Machine Learning Engineer has determined a similarity measure according to her expertise, using local similarity measures as specified in the tables below and feature weights that are given in the sequel.

Query \ Instance	No	Low	High
No	1	0.7	0.2
Low	0.5	1	0.8
High	0	0.3	1

Table: Local similarity for feature *fever*.

Query \ Instance	No	Yes
No	1	0
Yes	0	1

Table: Local similarity for features *vomiting*, *diarrhea*, and *shivering*.

(a) Compute the similarity between all instances and the query $(High, No, Yes, Yes)$ according to the formula:

$$\text{sim}(I, Q) = w_F s\text{sim}_F + w_V \text{sim}_V + w_D \text{sim}_D + w_S \text{sim}_S$$

where $\text{sim}(I, Q)$ is the total similarity between the instance I and the query, i.e. the test point, and $\text{sim}_F, \text{sim}_V, \text{sim}_D, \text{sim}_S$ are respectively the local similarities for features fever, vomiting, diarrhea, and shivering, and w_F, w_V, w_D , and w_S are their corresponding weights. Use $w_F = 0.25, w_V = .2, w_D = 0.3$, and $w_S = 0.25$.

(b) How can you calculate the similarity of the training instances with the test instance $(*, Yes, Yes, No)$, which is a patient whose fever level is unknown/missing? Calculate the similarity between the training instances and this test query using the weights in 4a.

(c) Determine the k-nearest neighbors of the test instances in 4a and 4b and the determine the diagnosis using $k = 3$.

5. The conditional probability distribution function of the weight W (in kg) given the height H (in cm) in a population is Gaussian (normal), and (35 pts)

$$p_{W|H}(w|h) = \frac{1}{\sqrt{2\pi} \times 10} \exp\left(-\frac{(w - 0.5 * h^{1.001})^2}{200}\right)$$

   (a) What is the best estimate of the weight of a person as a function of their height in the sense of *mean squared error*? (Hint: look up the Gaussian distribution, its mean, and variance. This problem is asking you to determine the regression function $w = f(h)$, which was shown in the lecture to be a statistical property of the conditional distribution of the output W given a particular value h of the input H.)

   (b) Use the result you obtained in 5a to estimate the weight of people whose heights are 155, 165, and 190 cm.

   (c) Would your answer to 5a and 5b change if the conditional variance of W given $H = h$ were a function of h, say, $\sigma(h)$, i.e.:

$$p_{W|H}(w|h) = \frac{1}{\sqrt{2\pi} \times \sigma(h)} \exp\left(-\frac{(w - 0.5 * h^{1.001})^2}{2[\sigma(h)]^2}\right)$$

   (d) Assume that instead of the conditional distribution, you have the following sample. Estimate the weight of people whose heights are 150, 155, 165, and 190 cm, using KNN with $k = 3$:

$$\hat{y}_{KNN} = \frac{y_1 + y_2 + \cdots + y_k}{k}$$

   where $y_1, y_2, \cdots, y_k$ are the labels of the k nearest neighbors to your test instance.

Person	Height (cm)	Weight (kg)
1	171	80
2	168	78
3	191	100
4	182	80
5	150	65
6	178	83

   (e) Repeat 5d, but instead of using the simple average of the labels of k nearest neighbors, which is use the following weighted average:

$$\hat{y}_{KNN} = \frac{w_1 y_1 + w_2 y_2 + \cdots + w_k y_k}{w_1 + w_2 + \cdots + w_k}$$

   where the weight w_i for the label y_i of instance i is determined as $1/d_i$, where d_i the distance between the instance i and the test instance.