Problem 1 (20 Points): Independence and Law of Total Probability

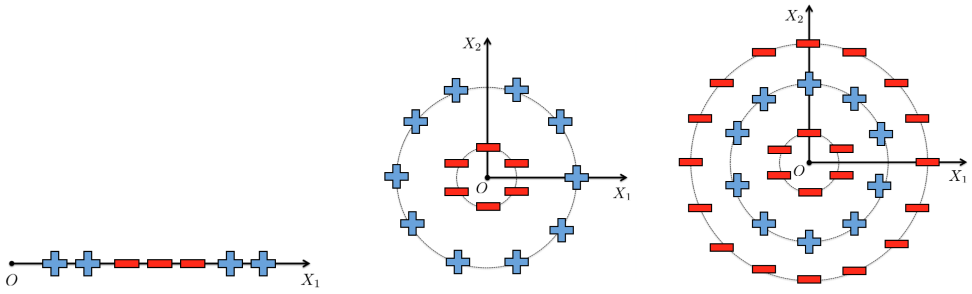
Let X, Y, Z all be binary variables, taking values either 0 or 1. Assume Y and Z are independent, and $P(Y = 1) = 0.9$ while $P(Z = 1) = 0.8$. Further, $P(X = 1|Y = 1, Z = 1) = 0.6$, and $P(X = 1|Y = 1, Z = 0) = 0.1$, and $P(X = 1|Y = 0) = 0.2$.

1. **10 Points.** Compute $P(X = 1)$. (Hint: use the law of total probability)
2. **5 Points.** Compute the expected value $\mathbb{E}[Y]$.
3. **5 Points.** Suppose that instead of Y attaining values 0 and 1, it takes one of two values 115 and 20, where $P(Y = 115) = 0.9$. Compute the expected value $\mathbb{E}[Y]$.

Problem 2 (20 Points): Bayes Rule

Alex owns a retail store for selling phones. The phones are manufactured at three different factories, A, B, C, where factory A, B, and C produces 20%, 30%, and 50% of the phone being sold at Alex's store. The probabilities of the defective phones from stores A, B, and C are 2%, 1%, and 0.05%, respectively. The total number of phones being sold at Alex's store is 10,000. One day, a customer walks up to Alex's store, and ask for a refund for a defective phone.

1. **5 Points.** What is the probability of a phone being defective?
2. **5 Points.** What is the probability that this defective phone is manufactured at factory A?
3. **5 Points.** What is the probability that this defective phone is manufactured at factory B?
4. **5 Points.** What is the probability that this defective phone is manufactured at factory C?



(a) (b) (c)

Figure 1: Dataset for (a) Question 1 and 2; (b) Question 3; and (c) Question 4.

Problem 3 (20 Points): Feature Transformation & Kernels

Designing transformations. In this problem, you will design some transformations of the original data points, i.e., derive features, to try to make a dataset linearly separable.
Note: If your answer is 'Yes', write out the expression for the transformation; Otherwise, briefly explain why

1. **4 Points.** Consider the 1-D dataset as shown in Figure 1(a). Can you think of a 1-D transformation that will make the points linearly separable?
2. **3 Points.** Still consider the above 1-D dataset (as shown in Figure 1(a)). Can you come up with a 2-D transformation that makes the points linearly separable?
3. **3 Points.** You may not always need to map to a higher dimensional space to make the data linearly separable. Consider the 2-D dataset as shown in Figure 1(b). Can you suggest a 1-D transformation that will make the data linearly separable?
4. **4 Points.** Using ideas from the above two datasets, can you suggest a 2-D transformation of the dataset, as shown in Figure 1(c), that makes it linearly separable?

Kernel of Not: For the following two functions, prove or disprove that it is a valid kernel.

1. **3 Points.** $k(x, z) = (xz + 1)^2$
2. **3 Points.** $k(x, z) = (xz - 1)^3$

Problem 4 (20 Points): Exponential Family & Geometric Distribution

1. **10 Points.** Consider the geometric distribution parameterized by ϕ

$$p(y;\phi) = (1-\phi)^{y-1}\phi, y = 1, 2, 3, \cdots$$

   Show that the geometric distribution is in the exponential family, and give $b(y)$, η, $T(y)$, and $a(\eta)$.

2. **10 Points.** Given a training set $\{(x_n, y_n)\}_{n=1}^N$ and let the log-likelihood of an example be $\log p(y_n|x_n; \mathbf{w})$. By taking the derivative of the log-likelihood with respect to $\mathbf{w}$, derive the stochastic gradient ascent rule for learning using a GLM model with goemetric responses y.

   Hint: Remember the three assumptions to derive a GLM model.

Problem 5 (20 Points): Implementation of the Perceptron Algorithm

In this problem, we will implement the Perceptron algorithm on synthetic training data.

1. **5 Points.** Suppose that the data dimension $d = 2$. Generate two classes of data points with 100 points each, by sampling from Gaussian distributions centered at $(0.5, 0.5)$ and $(-0.5, -0.5)$. Choose the variance of the Gaussian to be small enough so that the data points are sufficiently well separated.
2. **10 Points.** Implement the Perceptron algorithm as discussed in class. Choose the initial weights to be zero, the maximum number of epochs as $T = 100$, and the learning rate $\alpha = 1$. How quickly does your implementation converge?
3. **5 Points.** Now, repeat the above experiment with a second synthetic dataset; this time, increase the variance of the Gaussians such that the generated data points from different classes now overlap. What happens to the behavior of the algorithm?