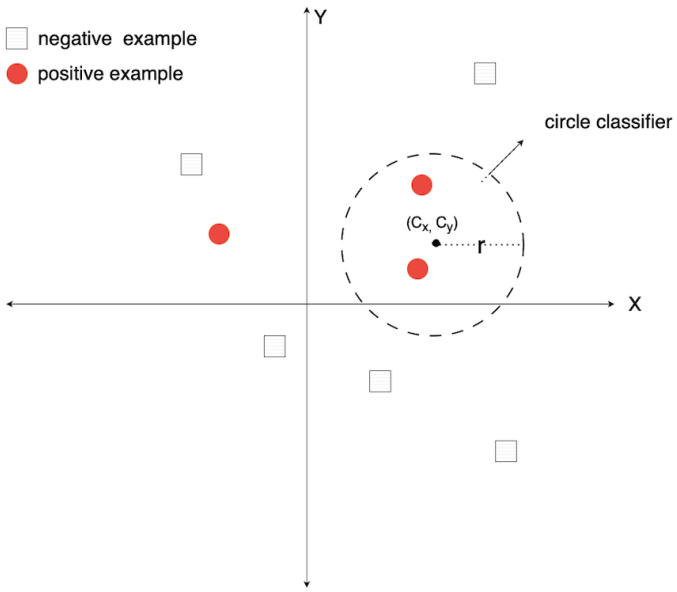
CS 461 Homework 1

Due: Sept. 27 11:59 pm

1. [Right Choice for Empirical Metric] Suppose we want to learn a ML model for binary classification in the 2-D feature domain and the circle is set as the hypothesis function set. It is defined by the two parameters: center (C_x, C_y) and radius (r), and its classification rule $\delta(x, y)$ is defined as below. We plan to train the circle classifier based on empirical error observed on a training set and evaluate its performance on a new dataset.

$$\begin{cases} \delta(x,y) = +1 & (x-c_x)^2 + (y-c_y)^2 \leq r^2 \\ \delta(x,y) = -1 & (x-c_x)^2 + (y-c_y)^2 > r^2 \end{cases} \tag{1}$$

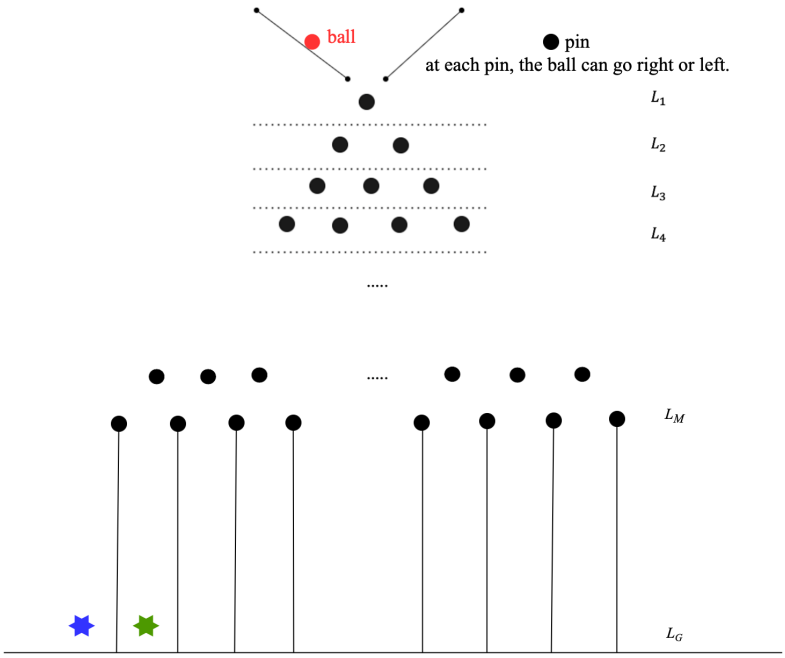


1.1 If precision rate is used to measure the empirical error on the training set, what kind of circle classifier can be trained? Please provide an example. .
sol) a small circle classifier only covers one positive sample.

1.2 If recall rate is used to measure empirical error on the training set, what kind of circle classifier can be trained? Please provide an example. .
sol) a large circle classifier covers all training data samples.

1.3 What problems are expected for the case of 1.1 and 1.2? Would you suggest another metric to learn a reasonable circle? .
sol) accuracy = (#positive samples within a circle + #negative samples outside of a circle)/ (#all samples)

2. [Galton Board] Suppose we have a Galton board as below and run an experiment to drop a ball into it. At each pin, the ball takes a path left or right with equal probability. Let us define a sample space $\Omega_i = \{L, R\}$ where Ω_i is the set of all possible sides the ball can take at the level L_i.



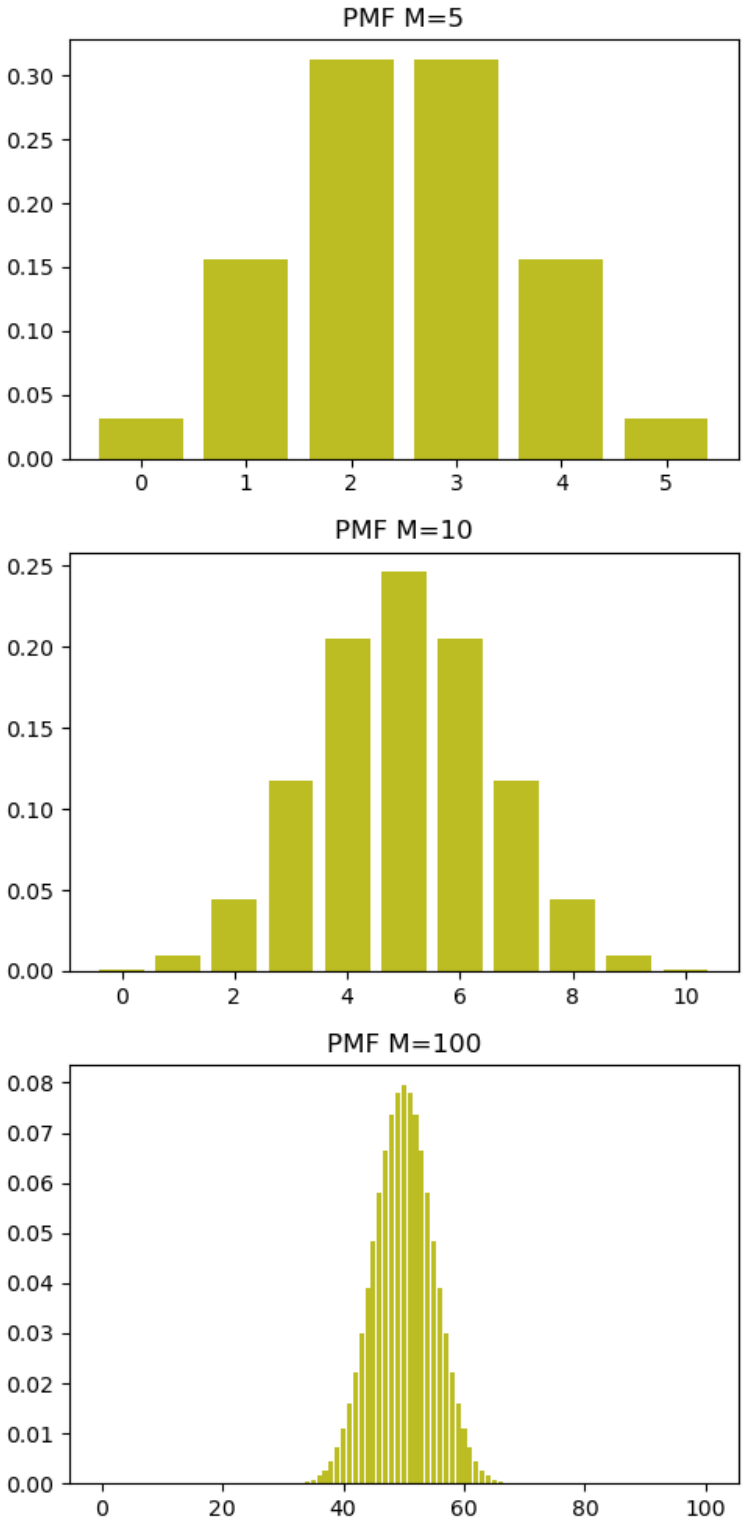
2.1 Define the sample spaces $\Omega_2, \Omega_3, ..., \Omega_M$. .
sol) $\Omega_i = \{L, R\} \quad \forall i$

2.2 Define the sample space $\Omega = \Omega_1 \times \Omega_2 \times ...\Omega_M$. Each element of the set Ω encodes a path the ball could take until it arrives at the ground-level L_G. .
sol) Ω is the Cartesian product for all Ω_i

2.3 What is the meaning of the location at L_G where the ball finally arrives? (hint: think about the possible ball path resulting in the different locations like the leftmost (blue star) or the second left location (green star). .
sol) The location indicates the number of L or R occurrence among M steps.

2.4 How would you represent the location numerically? Please define a random variable that maps Ω to the numerical values. .
sol) Random Variable X is binomial. X indicates the number of occurrence of L among M steps. For example, the blue star location will be encoded as 0 and the green star will be encoded as 1.

2.5 Define the PMF of your random variable for the depth M = 5, M = 10, M=100. Plot them and check how the PMFs change as M goes to large. Please explain the phenomenon in relation to Central Limit Theorem. .



sol) As M goes to ∞, the PMF of X converges to Gaussian Density. This phenomenon can be explained by Central Limit Theorem: the CDF of the sum of any iid random variables with finite mean and variance approaches the CDF of a Gaussian R.V. The binomial R.V X is the sum of Bernoulli R.V B_i indicating whether the ball takes the side L or R at level i. $X = \sum_{i=1}^{M} B_i$.

3. [Bayes Rule] Before going on vacation, you ask your friend to water your ailing plant. Without water, the plant has an 80 percent chance of dying. Even with proper watering, it has a 20 percent chance of dying. And the probability that your friend will forget to water it is 30 percent .

$$P[D = +|W = -] = 0.8$$
$$P[D = +|W = +] = 0.2$$
$$P[W = -] = 0.3$$

3.1 What's the chance that your plant will survive the week? .
sol)

$$\begin{aligned} P[D = -] &= P[D = -|W = -]P[W = -] + P[D = -|W = +]P[W = +] \\ &= (1 - 0.8) \cdot (0.3) + (1 - 0.2) \cdot (1 - 0.3) \\ &= 0.2 \cdot 0.3 + 0.8 \cdot 0.7 = 0.62 \end{aligned}$$

3.2 If your friend forgot to water it, what's the chance it'll be dead when you return?

$$\begin{aligned} P[D = +|W = -] &= \frac{P[D = +, W = -]}{P[W = -]} = \frac{P[D = +|W = -]P[W = -]}{P[W = -]} \\ &= \frac{0.8 \cdot 0.3}{0.3} = 0.8 \end{aligned}$$

3.3 If it's dead when you return, what's the chance your friend forgot to water it? .
sol)

$$\begin{aligned} P[W = -|D = +] &= \frac{P[W = -, D = +]}{P[D = +]} = \frac{P[D = +|W = -] \cdot P[W = -]}{1 - P[D = -]} \\ &= \frac{0.8 \cdot 0.3}{1 - 0.62} = 0.63 \end{aligned}$$

4. [Naïve Bayes] You will build a Naive Bayes Classifier to recognize diabetes based on the two factors of glucose and blood pressure. Let the random variable D be the indicator of diabetes, and the two random variables G and B represent glucose and blood pressure each. (note: you can use numpy or math but please do not use other libraries like SciPY.

4.1 Naïve Bayes is a probabilistic model based on Bayes Theorem. Rewrite the formula below as G and B are conditionally independent given $D+$. Also, write about $P[D = -|G = g, B = b]$. How would you use the two formulas to determine diabetes when you have a glucose and blood pressure record (g, b)?

$$P[D = +|G = g, B = b] = \frac{P[D = +, G = g, B = b]}{P[G = g, B = b]} = \frac{P[G = g, B = b|D = +] \cdot P[D = +]}{P[G = g, B = b]}$$

sol)

$$P[D = +|G = g, B = b] = \frac{P[D = +, G = g, B = b]}{P[G = g, B = b]} = \frac{P[G = g|D = +] \cdot P[B = b|D = +] \cdot P[D = +]}{P[G = g, B = b]}$$
$$P[D = -|G = g, B = b] = \frac{P[D = -, G = g, B = b]}{P[G = g, B = b]} = \frac{P[G = g|D = -] \cdot P[B = b|D = -] \cdot P[D = -]}{P[G = g, B = b]}$$

$$\begin{cases} \delta(g, b) = D+ & P[D = +|G = g, B = b] > P[D = -|G = g, B = b] \\ \delta(g, b) = D- & P[D = +|G = g, B = b] \leq P[D = -|G = g, B = b] \end{cases} \tag{2}$$

4.2 The data file "train.csv" contains patient records: glucose level, blood pressure, and diagnosis outcomes. Based on the formula you wrote in 4.1, estimate the necessary densities based on "train.csv". Glucose and blood pressure conditioned on diabetes follow Gaussian densities and use ML (Maximum Likelihood) estimators presented in the textbook Bishop section 1.2.4. Your file outcome will be "yourname-nb-train.py".

- for example, $f_G(g|D+) = N(\mu_{g-pos}, \sigma^2_{g-pos})$
- $f_B(b|D-) = N(\mu_{b-neg}, \sigma^2_{b-neg})$, and so on.

4.3 Write a Naïve Bayes classifier code "yourname-nb-cls.py" based on your estimated densities.

4.4 Evaluate your classifier using "test.csv". Use accuracy rate .
sol) about 93% accuracy rate (please check there is a reasonble implementation on their codes)

4.5 Do you think the standardization for data was necessary when building your Naïve Bayes classifier? If yes, then why? If not, why we don't need to?

$$std(x) = \frac{x - \mu}{\sigma} \tag{3}$$

sol) We don't need standardization process. It does not make any difference in the comparison between $P[D = +|G = g, B = b]$ and $P[D = -|G = g, B = b]$

4.6 Do you think the data reflects reality well? Which part of the previous steps would you like to change if we cannot collect the data again? How would you change? .
sol) It may not reflect our reality well for its prior probability estimated as $P[D+] = 0.51125$ and$P[D-] = 0.48875$. The probability of diabetes is known as around 10% in general. Hence, we need to adjust the prior probability distribution based on general knowledge.

5. [Data Whitening] Whitening data is a prepossessing step in machine learning. Suppose we have 10,000 3-D data points and computed mean and covariance information as below.

$$E[X] = \begin{bmatrix} 0.2 \\ 0.3 \\ 0.1 \end{bmatrix} \quad COV[X,X] = \begin{bmatrix} 2.75 & 0.43 & 0 \\ 0.43 & 2.25 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

5.1 Let Y be a random vector defined by $\vec{Y} = A\vec{X} + \vec{b}$. Express $E[Y]$ and $COV[Y,Y]$ in terms of $E[X]$ and $COV[X,X]$.
sol)

$$\begin{aligned} COV(Y,Y) &= E[(Y - E[Y])(Y - E[Y])^t] \\ &= E[(AX + b - AE[X] - b)(AX + b - AE[X] - b)^t] \\ &= AE[(X - E[X])(X - E[X])^t]A^t \\ &= A \cdot COV(X,X) \cdot A^t \end{aligned}$$

$$E[Y] = E[AX + b] = AE[X] + b$$

5.2 Design A and b to whiten Y. i.e. $E[Y] = 0$ and $COV[Y,Y] = I$.
Sol) By Spectral Decomposition Theorem (Some students may not use spectral decomposition. it is okay if that gives whitening),

$$COV[X,X] = U\Lambda U^t = \begin{bmatrix} -0.87 & -0.5 & 0 \\ -0.5 & 0.87 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 3 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -0.87 & -0.5 & 0 \\ -0.5 & 0.87 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$A = \Lambda^{1/2} \cdot U^t = \begin{bmatrix} -0.5 & -0.28867249 & 0 \\ -0.35354853 & 0.612373 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4}$$

$$b = -AE[X] = [0.1866023, -0.11300219, -0.1]^t \tag{5}$$

6. [Extra 20 Points, ML and MAP Estimation] Suppose we want to measure a length, for example, the water depth μ at 79.137°(N) and 2.817°(E). The depth was measured repeatedly and recorded as $x_1, x_2, ..., x_n$. For device imperfection, the samples were varied by ε where $\varepsilon \sim N(0, \sigma^2)$.i.e $x = \mu + \varepsilon$. Bayes rule allows us to evaluate the uncertainty in μ after observing $\vec{x}$ in the posterior probability $p(\mu|\vec{x})$ as below.

$$\mu^* = \arg\max_{\mu} p(\mu|\vec{x}) \propto p(\vec{x}|\mu) \cdot p(\mu)$$

6.1 Given the observations $x_1, x_2, ..., x_n$, derive a formula to estimate μ^*_{ML} when μ is a fixed value. We assume the observations are i.i.d (independent and identically distributed) and follow multivariate Gaussian.

$$f(\vec{x}) = \frac{1}{(2\pi\sigma^2)^{N/2}} \exp\{\frac{-1}{2\sigma^2}\sum_{n=1}^{N}(x_n - \mu)^2\}$$

sol)

$$\mu^*_{ML} = \arg\min_{\mu} \frac{1}{(2\pi\sigma^2)^{N/2}} \exp\{\frac{-1}{2\sigma^2}\sum_{n=1}^{N}(x_n - \mu)^2\}$$

$$\frac{\partial J(\mu*)}{\partial \mu} = 0, \quad \text{where} \quad J(\mu) = \frac{1}{(2\pi\sigma^2)^{N/2}} \exp\{\frac{-1}{2\sigma^2}\sum_{n=1}^{N}(x_n - \mu)^2\}$$

$$\sum_{n=1}^{N}(x_n - \mu^*_{ML}) = 0$$

$$N \cdot \mu^*_{ML} = \sum_{n=1}^{N} x_n$$

$$\mu^*_{ML} = \frac{1}{N}\sum_{n=1}^{N} x_n$$

6.2 Given the observations $x_1, x_2, ..., x_n$, derive a formula to estimate μ^*_{MAP} when μ is known to follow Gaussian $p(\mu) \sim N(\mu_0, \sigma_0^2)$. (hint: the posterior $p(\mu|\vec{X}) \propto p(X|\mu) \cdot p(\mu)$ is also Gaussian like $\sim N(\mu_N, \sigma_N^2)$. Then, it will have a mode at μ_N.) .
sol)

- In posterior, the quadratic function of μ: $\frac{-1}{2\sigma_N^2}(\mu - \mu_N)^2$ is same as $\frac{-1}{2\sigma^2}\sum_{n=1}^{N}(x_n - \mu)^2 + \frac{-1}{2\sigma_0^2}(\mu - \mu_0)^2$
- μ^2 term $= -\frac{1}{2}(\frac{N}{\sigma^2} + \frac{1}{\sigma_0^2}) \quad \therefore \frac{1}{\sigma_N^2} = (\frac{N}{\sigma^2} + \frac{1}{\sigma_0^2})$
- μ term is equal to $\frac{\mu_N}{\sigma_N^2}$: $\quad \frac{1}{\sigma^2}\sum_{n=1}^{N} x_n + \frac{1}{\sigma_0^2}\mu_0 = \frac{\mu_N}{\sigma_N^2} \quad \therefore \mu_N = \frac{\sigma^2}{N\sigma_0^2 + \sigma^2} \cdot \mu_0 + \frac{N\sigma_0^2}{N\sigma_0^2 + \sigma^2} \cdot \mu^*_{ML}$

6.3 The μ^*_{MAP} estimated in 6.2 can be expressed as follows. Examine how the behavior of μ^*_{MAP} changes for the two cases $N \to \infty$ and $N \to 0$ in relation to μ_0 and μ^*_{ML}. If you have enough data points, what method would you choose between ML and MAP?

$$\mu^*_{MAP} = \frac{\sigma^2}{N\sigma_0^2 + \sigma^2} \cdot \mu_0 + \frac{N\sigma_0^2}{N\sigma_0^2 + \sigma^2} \cdot \mu^*_{ML}$$

sol)

- $N \to \infty$: $\frac{\sigma^2/N}{\sigma_0^2 + \sigma^2/N} \cdot \mu_0 + \frac{\sigma_0^2}{\sigma_0^2 + \sigma^2/N} \cdot \mu^*_{ML} = \mu^*_{ML}$
- $N \to 0$: $\frac{\sigma^2}{0 \cdot \sigma_0^2 + \sigma^2} \cdot \mu_0 + \frac{0 \cdot \sigma_0^2}{0 \cdot \sigma_0^2 + \sigma^2} \cdot \mu^*_{ML} = \mu_0$

The formula demonstrates how the posterior estimation balances the influence of data and prior knowledge based on the number of data points. As N goes to ∞, the posterior estimation converges to μ^*_{ML}. On the other hand, N goes 0, the estimator is just the mean value of prior density and ML estimator does not contribute to the final estimation. The analysis shows that it is a safe to use MLE as we have enough data points without considering prior knowledge. In general, estimating prior density is costly.