CS-215 Assignment-2 Report

22B0914 - Utkarsh Pant
22B0994 - Jaladhi Joshi
22B1005 - Anirudh Garg

Contents

Question 1

PART

I

Y_1

CDF

The CDF of Y_1 is defined as:

$$P(Y_1 \leq x) = P(\max(X_1, X_2, \dots, X_n) \leq x) \tag{0.1}$$

If $\max(X_1, X_2, \dots, X_n) \leq x$, this means each X_i for $1 \leq i \leq n$. Hence, $Y_1 \leq x \implies X_i \leq x$ for $1 \leq i \leq n$.

So, we can write:

$$P(Y_1 \leq x) = P(X_1 \leq x) \cap P(X_2 \leq x) \dots P(X_n \leq x) \tag{0.2}$$

Since $X_1, X_2, \dots, X_n$ are independent and have CDF $F_X(x)$, we can use the independence property:

$$\begin{aligned} P(X_1 \leq x) \cap P(X_2 \leq x) \cap \dots \cap P(X_n \leq x) &= F_X(X_1 \leq x, X_2 \leq x, \dots, X_n \leq x) \\ &= P(X_1 \leq x) \times P(X_2 \leq x) \times \dots \times P(X_n \leq x) \end{aligned} \tag{0.3}$$

So, we have:

$$P(Y_1 \leq x) = (F_X(x))^n \tag{0.4}$$

This implies the CDF of Y_1:

$$F_{Y_1}(x) = (F_X(x))^n \tag{0.5}$$

PDF

The PDF of Y_1 is given by:

$$f_{Y_1}(x) = \frac{d}{dx}(F_{Y_1}(x)) \tag{0.6}$$

Now, let's calculate this derivative:

$$\frac{d}{dx}((F_X(x))^n) = n \times (F_X(x))^{(n-1)} \times \frac{d}{dx}F_X(x) \tag{0.7}$$

Where the PDF of each X_i for $1 \leq i \leq n$ is denoted as $f_X(x)$, and $\frac{d}{dx}F_X(x)$ is its derivative:

$$\frac{d}{dx}F_X(x) = f_X(x) \tag{0.8}$$

Therefore, the PDF of Y_1 is:

$$f_{Y_1}(x) = n \times (F_X(x))^{(n-1)} \times f_X(x) \tag{0.9}$$

Y_2

CDF

The CDF of Y_2 is defined as:

$$\begin{aligned} P(Y_2 \leq x) &= P(\min(X_1, X_2, \ldots, X_n) \leq x) \\ &= 1 - P(Y_2 > x) \\ &= 1 - P(\min(X_1, X_2, \ldots, X_n) > x) \end{aligned} \tag{0.10}$$

Probability of minimum of $X_1, X_2, \ldots, X_n$ being greater than x is equal to the probability of each of $X_1, X_2, \ldots, X_n$ being greater than x. The probability of each X_i being greater than x for $1 \leq i \leq n$ is:

$$1 - \text{probability of } X_i \leq x = 1 - F_X(x) \tag{0.11}$$

Hence, the probability of each of $X_1, X_2, \ldots, X_n$ being greater than x is equal to the product of probabilities, as all these events are independent:

$$P(\min(X_1, X_2, \ldots, X_n) > x) = (1 - F_X(x))^n \tag{0.12}$$

Hence :

$$\begin{aligned} P(Y_2 \leq x) &= P(\min(X_1, X_2, \ldots, X_n) \leq x) \\ &= 1 - (1 - F_X(x))^n \end{aligned} \tag{0.13}$$

This implies the CDF of Y_2:

$$F_{Y_2}(x) = 1 - (1 - F_X(x))^n \tag{0.14}$$

PDF

The PDF of Y_2 is given by:

$$f_{Y_2}(x) = \frac{d}{dx}(F_{Y_2}(x)) \tag{0.15}$$

Now, let's calculate this derivative:

$$\begin{aligned} \frac{d}{dx}(F_{Y_2}(x)) &= \frac{d}{dx}\left(1 - (1 - F_X(x))^n\right) \\ &= -n(1 - F_X(x))^{(n-1)} \cdot \frac{d}{dx}(1 - F_X(x)) \\ &= -n(1 - F_X(x))^{(n-1)} \cdot (-F'_X(x)) \\ &= -n(1 - F_X(x))^{(n-1)} \cdot (-f_X(x)) \\ &= n(1 - F_X(x))^{(n-1)} \cdot f_X(x) \end{aligned} \tag{0.16}$$

Hence,the PDF of Y_2 is:

$$f_{Y_2}(x) = n(1 - F_X(x))^{(n-1)} \cdot f_X(x) \tag{0.17}$$

Question 2

PART

II

Random variable X is defined as a Gaussian mixture model (GMM), which means it follows the distribution:

$$X \sim \sum_{i=1}^{K} p_i \cdot \mathcal{N}(\mu_i, \sigma_i^2)$$

Here, the mixing probabilities $\{p_1, p_2, \ldots, p_K\}$ determine the likelihood of selecting one of the K constituent Gaussians. A Gaussian with a higher mixing probability has a greater chance of being chosen. This results in X being a random variable that can take values from any of these K Gaussians.

On the other hand, random variable Z is defined as:

$$Z = \sum_{i=1}^{K} p_i \cdot X_i$$

where each X_i follows a normal distribution: $X_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$ for each $i \in \{1, 2, \ldots, K\}$. These X_i are independent random variables. Z is formed by a weighted sum of these independent Gaussian variables, reflecting a combination of the different Gaussian components.

In summary, while X is a single random variable that draws values from different Gaussians based on their mixing probabilities, Z is a random variable formed by combining multiple independent Gaussian variables, each weighted by its respective mixing probability.

X

Expectation E(X)

$$\begin{aligned} E(X) &= \int_{-\infty}^{\infty} x f_X(x) dx = \int_{-\infty}^{\infty} x \sum_{i=1}^{K} p_i N(\mu_i, \sigma_i) dx \\ &= \sum_{i=1}^{K} p_i \int_{-\infty}^{\infty} x N(\mu_i, \sigma_i) dx \end{aligned}$$

$$E(X) = \sum_{i=1}^{K} \mu_i p_i$$

$$
\begin{aligned}
\operatorname{Var}(X) &= E[x^2] - (E[x])^2 \\
&= \int_{-\infty}^{\infty} x^2 f_X(x)dx - \left(\int_{-\infty}^{\infty} x f_X(x)dx\right)^2 \\
&= \int_{-\infty}^{\infty} x^2 \sum_{i=1}^{K} p_i N(\mu_i, \sigma_i)dx - \left(\int_{-\infty}^{\infty} x \sum_{i=1}^{K} p_i N(\mu_i, \sigma_i)dx\right)^2 \\
&= \sum_{i=1}^{K} p_i \int_{-\infty}^{\infty} x^2 N(\mu_i, \sigma_i)dx - \left(\sum_{i=1}^{K} \mu_i p_i\right)^2 \\
&= \sum_{i=1}^{K} p_i \int_{-\infty}^{\infty} x^2 \frac{1}{\sqrt{2\pi\sigma_i^2}} e^{-\frac{(x-\mu_i)^2}{2\sigma_i^2}} dx - \left(\sum_{i=1}^{K} \mu_i p_i\right)^2 \\
&= \sum_{i=1}^{K} p_i(\sigma_i^2 + \mu_i^2) - \left(\sum_{i=1}^{K} \mu_i p_i\right)^2
\end{aligned}
$$

Variance(X):

$$
\operatorname{Var}(X) = \sum_{i=1}^{K} p_i(\sigma_i^2 + \mu_i^2) - \left(\sum_{i=1}^{K} \mu_i p_i\right)^2
$$

MGF (X)

$$
\begin{aligned}
\phi_X(t) &= \int_{-\infty}^{\infty} e^{tx} f_X(x)dx = \int_{-\infty}^{\infty} e^{tx} \sum_{i=1}^{K} p_i N(\mu_i, \sigma_i)dx \\
&= \sum_{i=1}^{K} p_i \int_{-\infty}^{\infty} e^{tx} N(\mu_i, \sigma_i)dx \\
&= \sum_{i=1}^{K} p_i \cdot MGF(X_i)
\end{aligned}
$$

Now, let's derive the MGF for the Gaussian component X_i:

$$\begin{aligned}
\phi_{X_i}(t) &= E(e^{tX_i}) \\
&= \int_{-\infty}^{\infty} e^{tx} \frac{1}{\sqrt{2\pi\sigma_i^2}} e^{-\frac{(x-\mu_i)^2}{2\sigma_i^2}}\, dx \\
&= \frac{1}{\sqrt{2\pi\sigma_i^2}} \int_{-\infty}^{\infty} e^{tx-\frac{(x-\mu_i)^2}{2\sigma_i^2}}\, dx \\
&= \frac{1}{\sqrt{2\pi\sigma_i^2}} \int_{-\infty}^{\infty} e^{-\frac{1}{2\sigma_i^2}(x^2-2\mu_i x+\mu_i^2-2\sigma_i^2 tx)}\, dx \\
&= \frac{1}{\sqrt{2\pi\sigma_i^2}} \int_{-\infty}^{\infty} e^{-\frac{1}{2\sigma_i^2}\left(x^2-2(\mu_i+\sigma_i^2 t)x+\mu_i^2+(\mu+\sigma_i^2 t)^2-(\mu+\sigma_i^2 t)^2\right)}\, dx \\
&= \frac{1}{\sqrt{2\pi\sigma_i^2}} \int_{-\infty}^{\infty} e^{-\frac{1}{2\sigma_i^2}\left(x^2-2(\mu_i+\sigma_i^2 t)x+(\mu+\sigma_i^2 t)^2-\sigma_i^4 t^2-2(\mu\sigma_i^2 t)\right)}\, dx \\
&= \frac{1}{\sqrt{2\pi\sigma_i^2}} \int_{-\infty}^{\infty} e^{-\frac{1}{2\sigma_i^2}\left(x-(\mu_i+\sigma_i^2 t)\right)^2} e^{(\frac{\sigma_i^2 t^2}{2}+\mu_i t)}\, dx \\
&= \frac{1}{\sqrt{2\pi\sigma_i^2}} e^{(\frac{\sigma_i^2 t^2}{2}+\mu_i t)} \int_{-\infty}^{\infty} e^{-\frac{1}{2\sigma_i^2}\left(x-(\mu_i+\sigma_i^2 t)\right)^2}\, dx \\
&= e^{(\frac{\sigma_i^2 t^2}{2}+\mu_i t)} \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi\sigma_i^2}} e^{-\frac{1}{2\sigma_i^2}\left(x-(\mu_i+\sigma_i^2 t)\right)^2}\, dx \\
&= e^{\frac{\mu_i t}{2}+\sigma_i^2 t^2}
\end{aligned}$$

So, the MGF of X_i is $e^{\mu_i t+\frac{\sigma_i^2 t^2}{2}}$. Therefore, the MGF of X is:

$$\phi_X(t) = \sum_{i=1}^{K} p_i e^{\mu_i t+\frac{\sigma_i^2 t^2}{2}}$$

Z

Expectation (E(Z)):

The expectation of the random variable Z is given by:

$$E(Z) = E\left(\sum_{i=1}^{K} p_i X_i\right)$$

Using the linearity of expectation, we can express $E(Z)$ as:

$$E(Z) = \sum_{i=1}^{K} p_i E(X_i)$$

Since X_i follows a normal distribution with mean μ_i for each $i \in \{1, 2, ..., K\}$, we have $E(X_i) = \mu_i$. Therefore, we can write:

$$E(Z) = \sum_{i=1}^{K} p_i \mu_i$$

So, the expectation of Z is:

$$E(Z) = \sum_{i=1}^{K} p_i \mu_i$$

Variance (Var(Z)):

The variance of the random variable Z is given by:

$$\mathrm{Var}(Z) = \mathrm{Var}\left(\sum_{i=1}^{K} p_i X_i\right)$$

Using the property that the variances of independent random variables add up, we can express $\mathrm{Var}(Z)$ as:

$$\mathrm{Var}(Z) = \sum_{i=1}^{K} p_i^2 \mathrm{Var}(X_i)$$

Since X_i follows a normal distribution with variance σ_i^2 for each $i \in \{1, 2, ..., K\}$, we have $\mathrm{Var}(X_i) = \sigma_i^2$. Therefore, we can write:

$$\mathrm{Var}(Z) = \sum_{i=1}^{K} p_i^2 \sigma_i^2$$

So, the variance of Z is:

$$\mathrm{Var}(Z) = \sum_{i=1}^{K} p_i^2 \sigma_i^2$$

Moment Generating Function (MGF) of Z:

The MGF of Z can be calculated using the MGFs of the individual X_is. The MGF of X_i is given by:

$$\phi_{X_i}(t) = e^{\mu_i t + \frac{\sigma_i^2 t^2}{2}}$$

Now, let's derive the MGF of Z:

$$\text{MGF of } Z: \quad \phi_Z(t) = E(e^{tZ})$$

Using the fact that Z is a weighted sum of X_is and they are independent, we can express $\phi_Z(t)$ as a product of the MGFs of X_is:

$$\phi_Z(t) = \phi_{\sum_{i=1}^{K} p_i X_i}(t)$$

$$\phi_Z(t) = \prod_{i=1}^{K} \phi_{p_i X_i}(t)$$

$$\phi_Z(t) = \prod_{i=1}^{K} \phi_{X_i}(tp_i)$$

Substituting the expression for $\phi_{X_i}(t)$ into the above equation:

$$\phi_Z(t) = \prod_{i=1}^{K} e^{(\mu_i tp_i) + \frac{\sigma_i^2 (tp_i)^2}{2}}$$

Simplifying the product:

$$\phi_Z(t) = e^{\sum_{i=1}^{K} (\mu_i tp_i) + \frac{1}{2} \sum_{i=1}^{K} (\sigma_i^2 (tp_i)^2)}$$

So, the moment generating function (MGF) of Z is:

$$\phi_Z(t) = e^{\sum_{i=1}^{K} (\mu_i tp_i) + \frac{1}{2} \sum_{i=1}^{K} (\sigma_i^2 (tp_i)^2)}$$

These are the complete derivations for $E(Z)$, $\mathrm{Var}(Z)$, and the MGF of Z.

PDF of Z:

$$f(z; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(z-\mu)^2}{2\sigma^2}}$$

Where: z is the random variable. μ is the mean of the distribution, calculated as:

$$\mu = \sum_{i=1}^{K} p_i \mu_i$$

σ is the standard deviation of the distribution, calculated as:

$$\sigma = \sqrt{\sum_{i=1}^{K} p_i^2 \sigma_i^2}$$

Hence,PDF OF Z,

$$f = \frac{1}{\left(\sqrt{\sum_{i=1}^{K} p_i^2 \sigma_i^2}\right)\sqrt{2\pi}} e^{-\frac{\left(z - \left(\sum_{i=1}^{K} p_i \mu_i\right)\right)^2}{2\left(\sum_{i=1}^{K} p_i^2 \sigma_i^2\right)}} \tag{0.18}$$

Question 3

PART

III

Consider the following integral for any $b > 0$,

For $\tau > 0$

$$P(x - \mu \geq \tau) = P(x - \mu + b \geq \tau + b) \leq P((x - \mu + b)^2 \geq (\tau + b)^2)$$

using Markov's inequality we get,

$$P(x - \mu \geq \tau) \leq \frac{E[(x - \mu + b)^2]}{(\tau + b)^2}$$

$$\begin{aligned} E[(x - \mu + b)^2] &= E[(x - \mu)^2 + b^2 + 2b(x - \mu)] \\ &= E[(x - \mu)^2] + 2bE[(x - \mu)] + E[b^2] \\ &= \sigma^2 + b^2 \end{aligned}$$

Substituting the value in inequality we get,

$$P(X - \mu \geq \tau) \leq \frac{\sigma^2 + b^2}{(\tau + b)^2}$$

The RHS of the inequality reaches minimum for $b = \frac{\sigma^2}{\tau}$. Subtituting b in the equation, we get,

$$\boxed{P(X - \mu \geq \tau) \leq \frac{\sigma^2}{(\tau^2 + \sigma^2)}}$$

Notice that $P(\mu - X \geq \tau)$ will also yield the same result i.e.

$$P(\mu - X \geq \tau) \leq \frac{\sigma^2}{(\tau^2 + \sigma^2)}$$

For $\tau < 0$

$$P(X - \mu \geq \tau) = 1 - P(X - \mu \leq \tau) = 1 - P(\mu - X \geq -\tau)$$

Also,

$$P(\mu - X \geq -\tau) = P(\mu - X + b \geq -\tau + b) \leq P((\mu - X + b)^2 \geq (-\tau + b)^2)$$

$$\begin{aligned} E[(\mu - X + b)^2] &= E[(\mu - X)^2 + b^2 + 2b(\mu - X)] \\ &= E[(x - \mu)^2] - 2bE[(x - \mu)] + E[b^2] \\ &= \sigma^2 + b^2 \end{aligned}$$

Substituting the value in inequality we get,

$$P(\mu - X \leq -\tau) \leq \frac{\sigma^2 + b^2}{(\tau - b)^2}$$

The RHS of the inequality reaches minimum for $b = -\frac{\sigma^2}{\tau}$. Subtituting b in the equation, we get,

$$P(X - \mu \leq \tau) \geq \frac{\sigma^2}{(\tau^2 + \sigma^2)}$$

or,

$$\boxed{P(X - \mu \geq \tau) \geq 1 - \frac{\sigma^2}{(\tau^2 + \sigma^2)}}$$

Question 4

PART

IV

$$\phi_{\mathbf{X}}(t) = \int_{-\infty}^{\infty} e^{tx} f_X(x)dx$$

For $t > 0$

$$\phi_{\mathbf{X}}(t) \geq \int_{x}^{\infty} e^{tx} f_X(x)dx$$

Since the value of e^{tx} increases as x varies from x to ∞

$$\begin{aligned}\phi_{\mathbf{X}}(t) &\geq \int_{x}^{\infty} e^{tx} f_X(x)dx \\ &\geq e^{tx} \int_{x}^{\infty} f_X(x)dx \;=\; e^{tx} \cdot P(X \geq x)\end{aligned}$$

Hence $P(X \geq x) \leq e^{-tx}\phi_{\mathbf{X}}(t)$

For $t < 0$

$$\phi_{\mathbf{X}}(t) \geq \int_{-\infty}^{x} e^{tx} f_X(x)dx$$

Since the value of e^{tx} increases as x varies from x to $-\infty$

$$\begin{aligned}\phi_{\mathbf{X}}(t) &\geq \int_{-\infty}^{x} e^{tx} f_X(x)dx \\ &\geq e^{tx} \int_{-\infty}^{x} f_X(x)dx \;=\; e^{tx} \cdot P(X \leq x)\end{aligned}$$

Hence $P(X \leq x) \leq e^{-tx}\phi_{\mathbf{X}}(t)$

Given: $X = X_1 + \ldots + X_n$

For $P(X > (1+\delta)\mu)$, we replace the value of x with $(1+\delta)\mu)$ in the equation derived from the first part.

$$P(X > (1+\delta)\mu) \leq \frac{\phi_{\mathbf{X}}(t)}{e^{t(1+\delta)\mu}}$$

Since MGF of sum of independent random variables is equal to the product of MGF of each random variable,

$$\begin{aligned}P(X > (1+\delta)\mu) &\leq \frac{\phi_{X_1}(t)\ldots\phi_{X_n}(t)}{e^{t(1+\delta)\mu}} \\ &\leq \frac{(1-p_1+p_1e^t)\ldots(1-p_n+p_ne^t)}{e^{t(1+\delta)\mu}}\end{aligned}$$

Using $1+x \leq e^x$ we can say that $1 + p_i(e^t - 1) \leq e^{p_i(e^t-1)}$

$$
\begin{aligned}
P(X > (1+\delta)\mu) &\leq \frac{e^{p_1(e^t-1)} \dots e^{p_n(e^t-1)}}{e^{t(1+\delta)\mu}} \\
&\leq \frac{e^{(e^t-1)(p_1 \dots + p_n)}}{e^{t(1+\delta)\mu}} \\
P(X > (1+\delta)\mu &\leq \frac{e^{(e^t-1)\mu}}{e^{t(1+\delta)\mu}}
\end{aligned}
$$

Note that the above equation is valid $\forall t > 0, \forall \delta \geq 0$. Inorder to find a tighter bound, we can find the minimum value of expression in RHS wrt to variable t.

$$
f(t) = \frac{e^{(e^t-1)\mu}}{e^{t(1+\delta)\mu}}
$$

Taking log and differentiating both sides wrt t we find the the minimum exists at,

$$
t = ln\,(1+\delta)
$$

Hence the optimal value is,

$$
P(X > (1+\delta)\mu) \leq \frac{e^{\delta\mu}}{e^{(1+\delta)\mu ln(1+\delta)}}
$$

PART

V

Question 5

a

My MATLAB code calculates the empirically determined distribution of the average of N independent random variables, each taking on values 1, 2, 3, 4, 5 with probabilities 0.05, 0.4, 0.15, 0.4, and 0.1 respectively. I loop through different values of N, generate random samples for each N, calculate the average, and plot histograms with 50 bins to visualize the distribution. The histograms demonstrate how the average of these random variables converges to its expected value as N increases, showcasing the Central Limit Theorem in action.

```
clear;
close all;
clc;

% Define the discrete distribution parameters
values = [1, 2, 3, 4, 5];
probabilities = [0.05, 0.4,0.15, 0.3, 0.1];

for N = [5, 10, 20, 50, 100, 200, 500, 1000, 5000, 10000]
    nsamp = 6000;
    X = zeros(nsamp, N);

    for i = 1:N
        X(:, i) = randsample(values, nsamp, true,
            probabilities);
    end

    % Calculate the sample means
    sample_means = mean(X, 2);

    % Plot the histogram of the sample means
    numbins = 50;
    histogram(sample_means, numbins, 'Normalization', '
        probability');
    title(sprintf('PDF of the average of %d iid random
        variables', N));

    fname = sprintf('histogram_%d.png', N);
    saveas(gcf, fname);

    pause(10);
end
```

Listing 1. MATLAB Code

This code uses minipage to create a grid of figures with captions. If you still encounter issues, please check if there are any conflicting packages or settings in your LaTeX document that might be causing the problem.

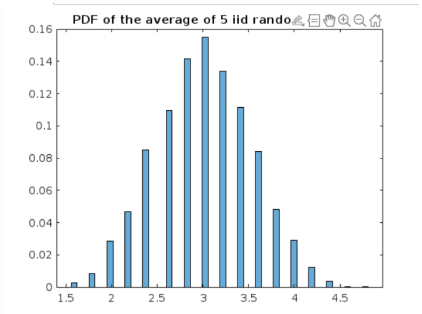

Figure 1. 5 random variables

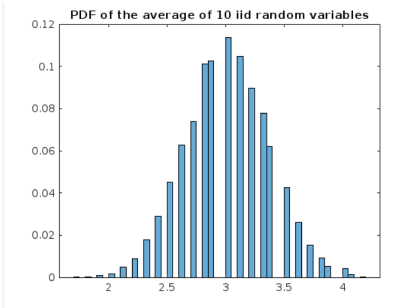

Figure 2. 10 random variables

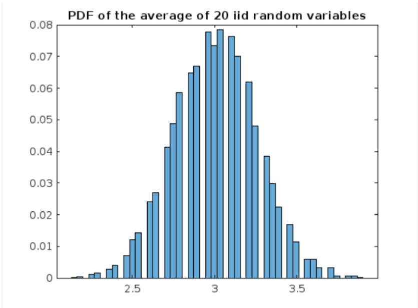

Figure 3. 20 random variables

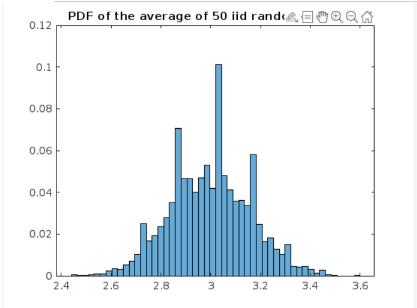

Figure 4. 50 random variables

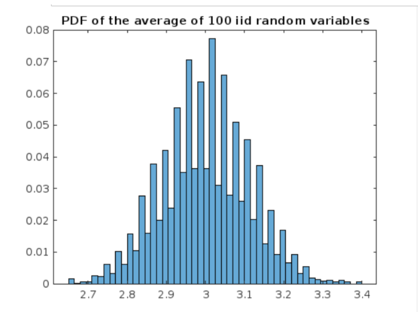

Figure 5. 100 random variables

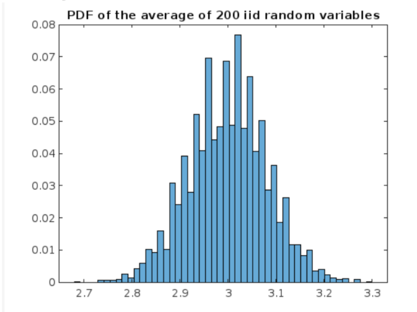

Figure 6. 200 random variables

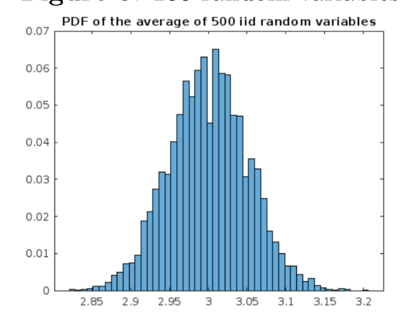

Figure 7. 500 random variables

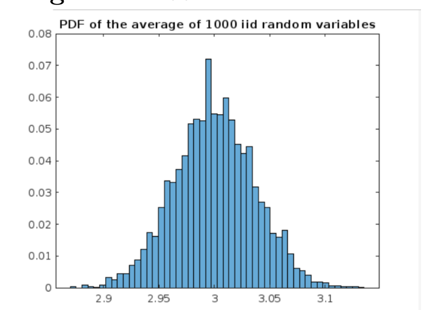

Figure 8. 1000 random variables

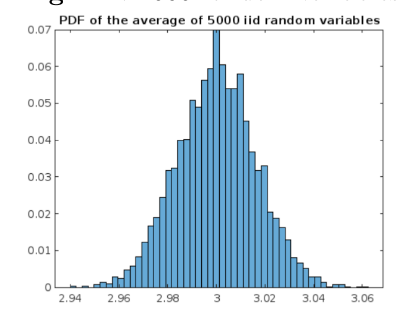

Figure 9. 5000 random variables

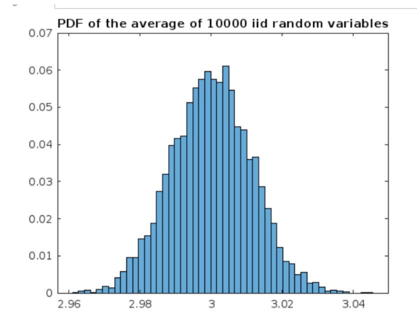

Figure 10. 10000 random variables

Figure 11. Empirically determined averages of random variables

b

In the MATLAB code, I simulated random variables with specific probabilities and calculated their averages. I then determined the empirical cumulative distribution function (CDF) for these averages and plotted it alongside the CDF of a Gaussian distribution with matching mean and standard deviation. This allowed me to visualize how the sample averages converge to a Gaussian distribution as the sample size increases, illustrating the Central Limit Theorem.

```
clear;
close all;
clc;

% Define the discrete distribution parameters
values = [1, 2, 3, 4, 5];
probabilities = [0.05, 0.4, 0.15, 0.3, 0.1];

for N = [5, 10, 20, 50, 100, 200, 500, 1000, 5000, 10000]
    nsamp = 6000;
    X = zeros(nsamp, N);

    for i = 1:N
        X(:, i) = randsample(values, nsamp, true,
            probabilities);
    end

    % Calculate the sample means
    sample_means = mean(X, 2);

    % Compute the empirical CDF
    [ecdf_vals, ecdf_x] = ecdf(sample_means);

    % Calculate the mean and standard deviation of the sample
      means
    mu = mean(sample_means);
    sigma = std(sample_means);

    % Compute the Gaussian CDF
    norm_cdf = normcdf(ecdf_x, mu, sigma);

    % Plot the empirical CDF and Gaussian CDF
    figure;
    plot(ecdf_x, ecdf_vals, 'b');
    hold on;
    plot(ecdf_x, norm_cdf, 'r');
    title(sprintf('Empirical vs. Gaussian CDF for %d iid
      random variables', N));
    legend('Empirical CDF', 'Gaussian CDF');
    hold off;
```

```

    pause(10);
end
```

Listing 2. MATLAB Code

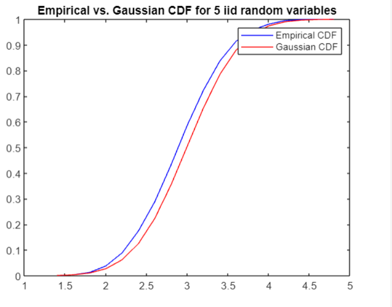

Figure 12. 5 random variables

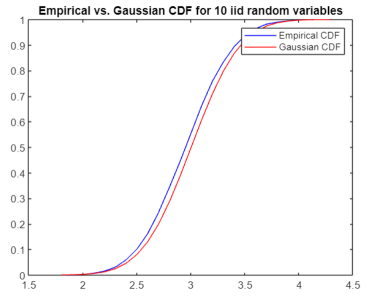

Figure 13. 10 random variables

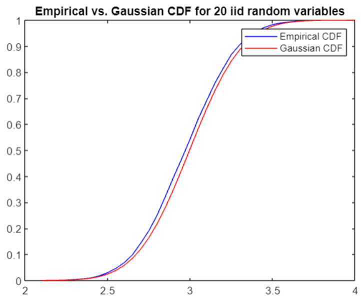

Figure 14. 20 random variables

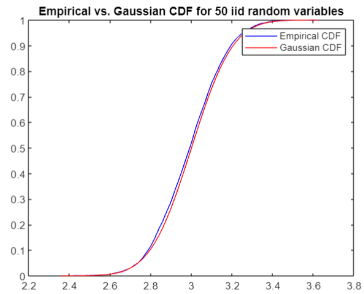

Figure 15. 50 random variables

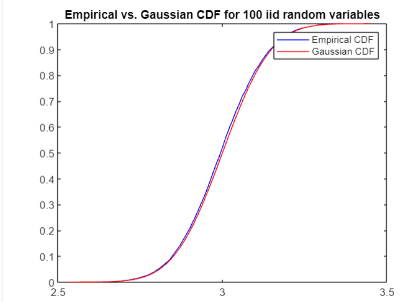

Figure 16. 100 random variables

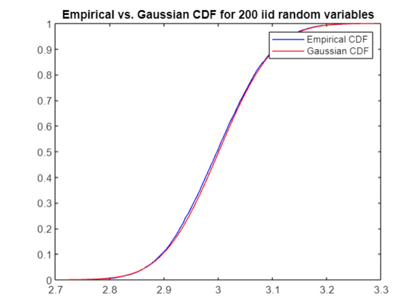

Figure 17. 200 random variables

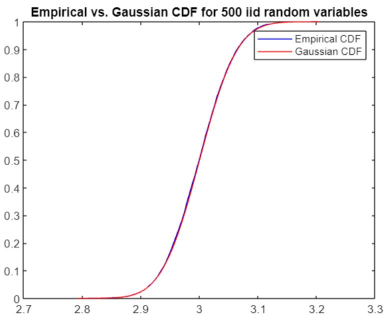

Figure 18. 500 random variables

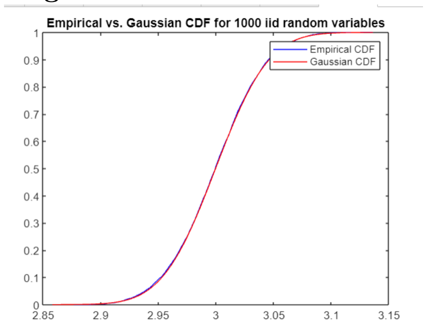

Figure 19. 1000 random variables

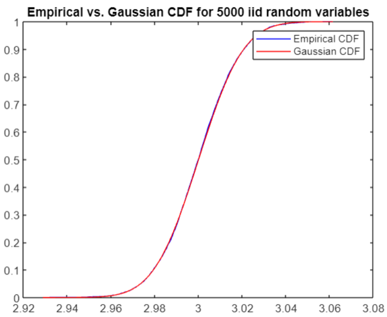

Figure 20. 5000 random variables

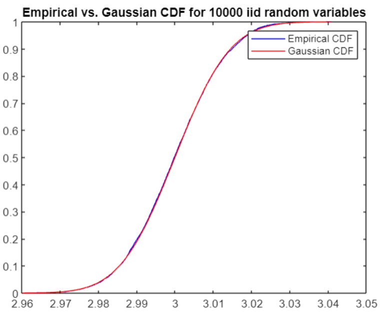

Figure 21. 10000 random variables

Figure 22. Empirically vs Gaussian CDF

c

Clearing previous data, closing existing plots, and initializing variables are the initial steps. The code then loops through different sample sizes ($N_{v}alues$) and generates random samples for each N. The samples are constructed based on given probabilities. Next, the empirical cumulative distribution function (ECDF) is calculated using the ecdf function.

For each N, the mean (mu) and standard deviation (sigma) of the samples are computed. The code then calculates the Maximum Absolute Difference (MAD) between the ECDF and Gaussian Cumulative Distribution Function (CDF). This MAD quantifies how much the empirical distribution differs from a Gaussian distribution with the same mean and standard deviation.

The MAD values for different N are stored in the '*mad_values*' array. Finally, the code plots a graph of MAD as a function of N to visualize how the difference between the empirical and Gaussian CDFs changes with increasing sample size. This helps us understand how well the Central Limit Theorem holds for the given random variables as N varies.

```
clear;
close all;
clc;

% Define the discrete distribution parameters
values = [1, 2, 3, 4, 5];
probabilities = [0.05, 0.4, 0.15, 0.3, 0.1];

% Define x-values for calculating MAD
x_values = linspace(0, 6, 1000);  % Adjust the range and granularity as needed

% Initialize arrays to store MAD values and N values
N_values = [5, 10, 20, 50, 100, 200, 500, 1000, 5000, 10000];
mad_values = zeros(size(N_values));

for n_idx = 1:length(N_values)
    N = N_values(n_idx);
    nsamp = 6000;

    % Generate random variables with specified probabilities
    X = zeros(nsamp, N);
    for i = 1:N
        random_indices = randsample(length(values), nsamp, true, probabilities);
        X(:, i) = values(random_indices);
    end

    % Calculate the sample means
```

```
    sample_means = mean(X, 2);

    % Compute the empirical CDF
    [ecdf_vals, ecdf_x] = ecdf(sample_means);

    % Calculate the Gaussian CDF
    mu = mean(sample_means);
    sigma = std(sample_means);
    norm_cdf = normcdf(ecdf_x, mu, sigma);

    % Interpolate the Gaussian CDF to match the x-values of the empirical CDF
    %interpolated_norm_cdf = interp1(ecdf_x, norm_cdf, x_values);

    % Calculate the MAD for this N
    mad_values(n_idx) = max(abs(ecdf_vals - norm_cdf));
end

% Plot MAD as a function of N
figure;
plot(N_values, mad_values, 'o-');
xlabel('N (Number of Random Variables)');
ylabel('Maximum Absolute Difference (MAD)');
title('Maximum Absolute Difference (MAD) vs. N');
grid on;
```

Listing 3. MATLAB Code

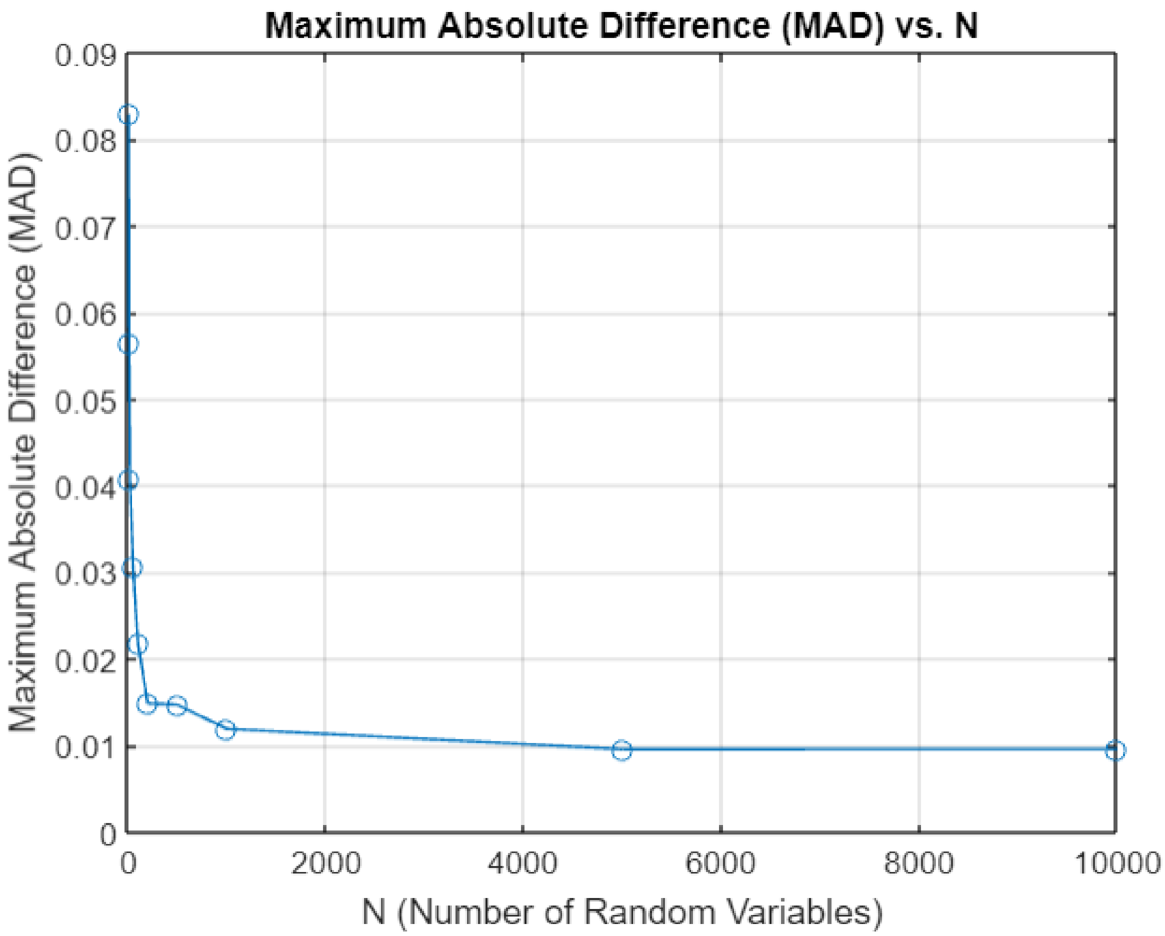

Figure 23. MAD as a function of N

PART

VI

Question 6

Analysis Results

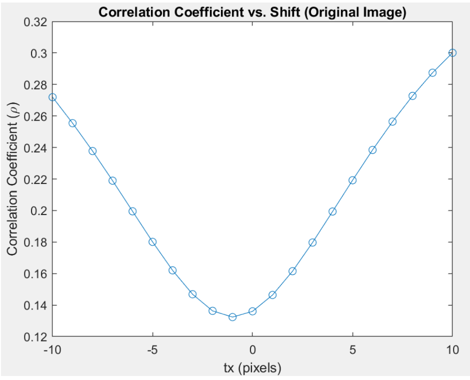

Figure 1. Correlation Coefficient (ρ) vs. Shift (Original Image)

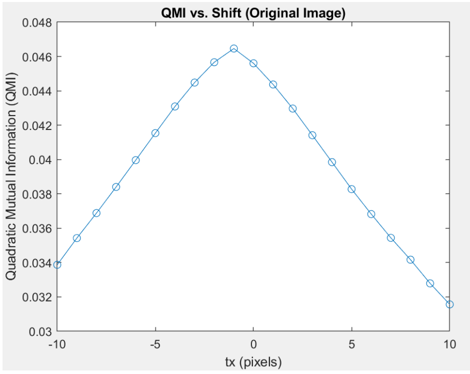

Figure 2. QMI vs. Shift (Original Image)

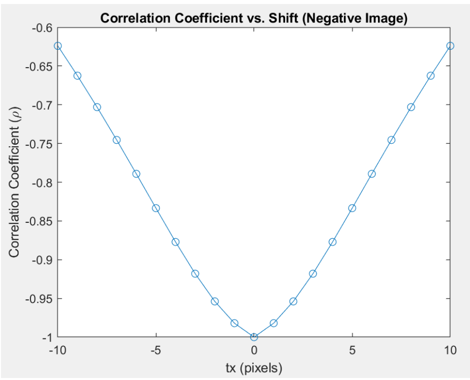

Figure 3. Correlation Coefficient (ρ) vs. Shift (Negative Image)

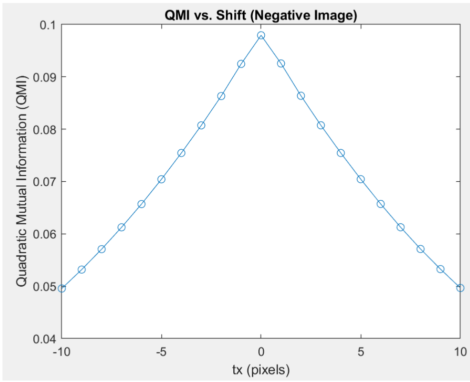

Figure 4. QMI vs. Shift (Negative Image)

Figure 5. Dependence Measures for Image Alignment

For Quadratic Mutual Information (QMI):

- QMI values start relatively high at 0.0496 when the images perfectly match (tx = 0).
- As we shift the images, QMI drops significantly, reaching its lowest value of approximately 0.0496 at the maximum shift.
- The decrease in QMI indicates that the dependence between pixel intensities in the negative image decreases as the images are shifted.

For Correlation Coefficient (ρ):

- The correlation coefficient starts low at approximately -0.624 when the images are perfectly aligned (tx = 0).

- As we shift the images, the correlation coefficient becomes even more negative, reaching its lowest possible value of -1 at the maximum shift.
- This strong negative correlation indicates an opposite relationship between pixel intensities in the negative image, and the negative correlation becomes even stronger as the images are shifted.

Summary:

- Image alignment significantly affects the relationship between the two images.
- The negative image shows a strong negative correlation when aligned but becomes even more negative when shifted.
- The original image exhibits weaker relationships when shifted, with both correlation and QMI decreasing slightly.

MATLAB Code

```
clear
clc

image1_path = 'T1.jpg';
image2_path = 'T2.jpg';

image1_info = imfinfo(image1_path);
image2_info = imfinfo(image2_path);

im1 = double(imread(image1_path));
im2 = double(imread(image2_path));

shift_range = -10:10;

num_shifts = length(shift_range);
correlation_coefficients = zeros(1, num_shifts);
qmi_values = zeros(1, num_shifts);

correlation_coefficients_negative = zeros(1, num_shifts);
qmi_values_negative = zeros(1, num_shifts);

bin_width = 10;

for i = 1:num_shifts
    % Shift the second image along the X direction
    tx = shift_range(i);
    shifted_im2 = imtranslate(im2, [tx, 0]);

    % Calculate the joint histogram
    joint_hist = zeros(round(256 / bin_width), round(256 /
       bin_width));
```

```
    for x = 1:image1_info.Height
        for y = 1:image1_info.Width
            i1 = floor(im1(x, y) / bin_width) + 1;
            i2 = floor(shifted_im2(x, y) / bin_width) + 1;
            joint_hist(i1, i2) = joint_hist(i1, i2) + 1;
        end
    end

    % Normalize the joint histogram
    joint_hist = joint_hist / sum(joint_hist(:));

    % Calculate the marginal histograms
    marginal_hist1 = sum(joint_hist, 2);
    marginal_hist2 = sum(joint_hist, 1);

    % Calculate the correlation coefficient
    covar = sum(sum((im1 - mean(im1(:))) .* (shifted_im2 - mean(shifted_im2(:)))));
    std1 = sqrt(sum(sum((im1 - mean(im1(:))).^2)));
    std2 = sqrt(sum(sum((shifted_im2 - mean(shifted_im2(:))).^2)));
    correlation_coefficients(i) = covar / (std1 * std2);

    % Calculate the QMI
    qmi = 0;
    for i1 = 1:size(joint_hist, 1)
        for i2 = 1:size(joint_hist, 2)
            pI1I2 = joint_hist(i1, i2);
            pI1 = marginal_hist1(i1);
            pI2 = marginal_hist2(i2);
            qmi = qmi + (pI1I2 - pI1 * pI2)^2;
        end
    end
    qmi_values(i) = qmi;

    % Calculate the correlation coefficient for negative image
    negative_im2 = 255 - im1;
    shifted_negative_im2 = imtranslate(negative_im2, [tx, 0]);

    % Calculate the joint histogram for negative image
    joint_hist_negative = zeros(round(256 / bin_width), round(256 / bin_width));
    for x = 1:image1_info.Height
        for y = 1:image1_info.Width
            i1 = floor(im1(x, y) / bin_width) + 1;
            i2 = floor(shifted_negative_im2(x, y) / bin_width) + 1;
            joint_hist_negative(i1, i2) = joint_hist_negative(i1, i2) + 1;
```

```
        end
    end

    % Normalize the joint histogram for negative image
    joint_hist_negative = joint_hist_negative / sum(
       joint_hist_negative(:));

    marginal_hist1 = sum(joint_hist_negative, 2);
    marginal_hist2 = sum(joint_hist_negative, 1);

    % Calculate the correlation coefficient for negative image
    covar_negative = sum(sum((im1 - mean(im1(:))) .* (
       shifted_negative_im2 - mean(shifted_negative_im2(:)))))
       ;
    std1_negative = sqrt(sum(sum((im1 - mean(im1(:))).^2)));
    std2_negative = sqrt(sum(sum((shifted_negative_im2 - mean(
       shifted_negative_im2(:))).^2)));
    correlation_coefficients_negative(i) = covar_negative / (
       std1_negative * std2_negative);

    % Calculate the QMI for negative image
    qmi_negative = 0;
    for i1 = 1:size(joint_hist_negative, 1)
        for i2 = 1:size(joint_hist_negative, 2)
            pI1I2 = joint_hist_negative(i1, i2);
            pI1 = marginal_hist1(i1);
            pI2 = marginal_hist2(i2);
            qmi_negative = qmi_negative + (pI1I2 - pI1 * pI2)
               ^2;
        end
    end
    qmi_values_negative(i) = qmi_negative;
end

% Plot correlation coefficients versus tx for the original
   image
figure;
plot(shift_range, correlation_coefficients, '-o');
xlabel('tx (pixels)');
ylabel('Correlation Coefficient (\rho)');
title('Correlation Coefficient vs. Shift (Original Image)');

% Plot QMI values versus tx for the original image
figure;
plot(shift_range, qmi_values, '-o');
xlabel('tx (pixels)');
ylabel('Quadratic Mutual Information (QMI)');
title('QMI vs. Shift (Original Image)');

```

```
% Plot correlation coefficients versus tx for the negative
    image
figure;
plot(shift_range, correlation_coefficients_negative, '-o');
xlabel('tx (pixels)');
ylabel('Correlation Coefficient (\rho)');
title('Correlation Coefficient vs. Shift (Negative Image)');

% Plot QMI values versus tx for the negative image
figure;
plot(shift_range, qmi_values_negative, '-o');
xlabel('tx (pixels)');
ylabel('Quadratic Mutual Information (QMI)');
title('QMI vs. Shift (Negative Image)');
```

Question 7

PART

VII

Let $\mathbf{X}$ be a random vector with n dimensions $X_1, X_2, \ldots, X_n$, representing the count of occurrences in n categories in a multinomial distribution.

The moment generating function (MGF) X_i is defined as:

$$M_{\mathbf{X}}(\mathbf{t}) = E[\exp\left(\mathbf{t}^T\mathbf{X}\right)]$$

Which means,

$$M_{\mathbf{X}}(\mathbf{t}) = E[\exp\left(\sum t_i X_i\right)]$$

Let k_1,…,k_n be non-negative integers and $k = k1 + \ldots + k_n$, then:

$$\begin{aligned}
\frac{\partial^k}{\partial t_1^{k_1} \ldots \partial t_n^{k_n}} M_{\mathbf{X}}(\mathbf{t}) &= \frac{\partial^k}{\partial t_1^{k_1} \ldots \partial t_n^{k_n}} E(e^{t_1X_1+\ldots+t_nX_n}) \\
&= E(\frac{\partial^k}{\partial t_1^{k_1} \ldots \partial t_n^{k_n}}(e^{t_1X_1+\ldots+t_nX_n})) \\
&= E(X_1^{k_1} \ldots X_n^{k_n}(e^{t_1X_1+\ldots+t_nX_n}))
\end{aligned} \tag{0.19}$$

Setting $\mathbf{t} = \mathbf{0} = (0, \ldots, 0)^T$ we get,

$$\boxed{\frac{\partial^k}{\partial t_1^{k_1} \ldots \partial t_n^{k_n}} M_{\mathbf{X}}(\mathbf{t})|_{t=0} = E(X_1^{k_1} \ldots X_n^{k_n})} \tag{0.20}$$

Also we know $M_{\mathbf{X}}(\mathbf{t})$ of a multinomial distribution is,

$$M_{\mathbf{X}}(\mathbf{t}) = (p_1 \cdot e^{t_1} + p_2 \cdot e^{t_2} \ldots + p_n \cdot e^{t_n})^n$$

Our goal is to find the covariance matrix of $\mathbf{X}$, which is a $n \times n$ matrix. Since a covariance matrix C is symmetric, we need to derive expressions for both the diagonal elements C_{ii} and the off-diagonal elements C_{ij} where $i \neq j$.

Deriving Diagonal Elements C_{ii}

The diagonal elements of the covariance matrix represent the variances of individual components X_i:

$$\text{Var}(X_i) = E[X_i^2] - (E[X_i])^2$$

From the above derivation we can calculate $E[X_i^2]$ and $E[X_i]$ as follows,

$$
\begin{aligned}
C_{ii} &= \frac{\partial^2}{\partial t_i^2} M_{\mathbf{X}}(\mathbf{t}) - \left(\frac{\partial}{\partial t_i} M_{\mathbf{X}}(\mathbf{t})\right)^2 |_{t=0} \\
&= np_i(1 + (n-1)p_i) - (np_i)^2 \\
&= np_i(1 - p_i)
\end{aligned}
$$

Deriving Off-Diagonal Elements C_{ij}

The off-diagonal elements of the covariance matrix represent the covariances between different components X_i and X_j where $i \neq j$:

$$
\begin{aligned}
\mathrm{Cov}(X_i, X_j) &= E[(X_i - \mu_i)(X_j - \mu_j)] && (0.21) \\
&= E[X_i X_j] - E[X_i]E[X_j] && (0.22)
\end{aligned}
$$

Where μ_i and μ_j are the means of X_i and X_j, respectively.

To find $\mathrm{Cov}(X_i, X_j)$, we use the above derivation,

$$
\begin{aligned}
C_{ij} &= \frac{\partial^2}{\partial t_i \partial t_j} M_{\mathbf{X}}(\mathbf{t}) - \left(\frac{\partial}{\partial t_i} M_{\mathbf{X}}(\mathbf{t})\right)\left(\frac{\partial}{\partial t_j} M_{\mathbf{X}}(\mathbf{t})\right) |_{t=0} \\
&= n(n-1)p_i p_j - np_i np_j \\
&= -np_i p_j
\end{aligned}
$$

Assembling the Covariance Matrix

Hence the covariance matrix becomes as follows,

$$
C_{ij} = \begin{cases} np_i(1 - p_i) & i = j \\ -np_i p_j & i \neq j \end{cases}
$$