

```

import numpy as np
import matplotlib.pyplot as plt
import os

# Seeding for reproducibility and setting number of samples.
np.random.seed(0)
N = 100

# Making directory for the plots to go to.
os.mkdir('./GaussianPlots')

# Part A: Drawing samples from the multivariate Gaussian distribution.
mu = np.array([0, 0])
Sigma = np.array([[1, 0],
                  [0, 1]])
samples = np.random.multivariate_normal(mu, Sigma, N)

plt.figure()
plt.scatter(samples[:,0], samples[:,1], alpha=0.7)
plt.title("Mean = [0,0], Cov = I")
plt.xlabel("x1")
plt.ylabel("x2")
plt.axis('equal')
plt.savefig('./GaussianPlots/Gaussian1.pdf')
plt.show()

# Part B: Changing the mean.
mu_shifted = np.array([1, 1])
samples_shifted = np.random.multivariate_normal(mu_shifted, Sigma, N)

plt.figure()
plt.scatter(samples_shifted[:,0], samples_shifted[:,1], alpha=0.7, c='orange')
plt.title("Mean = [1,1], Cov = I")
plt.xlabel("x1")
plt.ylabel("x2")
plt.axis('equal')
plt.savefig('./GaussianPlots/Gaussian2.pdf')
plt.show()

# Part C: Doubling the variance in each direction.
Sigma_doubled = 2 * np.eye(2)

```

```

samples_doubled = np.random.multivariate_normal(mu, Sigma_doubled, N)

plt.figure()
plt.scatter(samples_doubled[:,0], samples_doubled[:,1], alpha=0.7, c='green')
plt.title("Mean = [0,0], Cov = 2 * I")
plt.xlabel("x1")
plt.ylabel("x2")
plt.axis('equal')
plt.savefig('./GaussianPlots/Gaussian3.pdf')
plt.show()

# Part D: Changing the covariance matrix.
Sigma_corr1 = np.array([[1, 0.5],
                        [0.5, 1]])
samples_corr1 = np.random.multivariate_normal(mu, Sigma_corr1, N)

plt.figure()
plt.scatter(samples_corr1[:,0], samples_corr1[:,1], alpha=0.7, c='red')
plt.title("Mean = [0,0], Cov = [[1,0.5],[0.5,1]]")
plt.xlabel("x1")
plt.ylabel("x2")
plt.axis('equal')
plt.savefig('./GaussianPlots/Gaussian4.pdf')
plt.show()

# Part E: Changing the covariance matrix again.
Sigma_corr2 = np.array([[1, -0.5],
                        [-0.5, 1]])
samples_corr2 = np.random.multivariate_normal(mu, Sigma_corr2, N)

plt.figure()
plt.scatter(samples_corr2[:,0], samples_corr2[:,1], alpha=0.7, c='magenta')
plt.title("Mean = [0,0], Cov = [[1,-0.5],[-0.5,1]]")
plt.xlabel("x1")
plt.ylabel("x2")
plt.axis('equal')
plt.savefig('./GaussianPlots/Gaussian5.pdf')
plt.show()

```