```python
import numpy as np
from scipy.stats import norm

# Define the data
data = np.array([1.2, 2.3, 0.7, 1.6, 1.1, 1.8, 0.9, 2.2])

# Initialize the parameters
mu1 = 0
mu2 = 1
sigma1 = 1
sigma2 = 1
p1 = 0.5
p2 = 0.5

# Run the EM algorithm
for i in range(10):
    # E-step
    likelihood1 = norm.pdf(data, mu1, sigma1)
    likelihood2 = norm.pdf(data, mu2, sigma2)
    weight1 = p1 * likelihood1 / (p1 * likelihood1 + p2 * likelihood2)
    weight2 = p2 * likelihood2 / (p1 * likelihood1 + p2 * likelihood2)

    # M-step
    mu1 = np.sum(weight1 * data) / np.sum(weight1)
    mu2 = np.sum(weight2 * data) / np.sum(weight2)
    sigma1 = np.sqrt(np.sum(weight1 * (data - mu1)**2) / np.sum(weight1))
    sigma2 = np.sqrt(np.sum(weight2 * (data - mu2)**2) / np.sum(weight2))
    p1 = np.mean(weight1)
    p2 = np.mean(weight2)

# Print the final estimates of the parameters
print("mu1:", mu1)
print("mu2:", mu2)
print("sigma1:", sigma1)
print("sigma2:", sigma2)
print("p1:", p1)
print("p2:", p2)
```

```
Python 3.10.11 (tags/v3.10.11:7d4cc5a, Apr  5 2023, 00:38:17) [MSC v.1929 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
============ RESTART: C:/Users/Raghul/Desktop/ML/EM Algorithm(10).py ==========
mu1: 0.9545902456963998
mu2: 1.7595212637782114
sigma1: 0.19986282179149245
sigma2: 0.47713642731204714
p1: 0.3534728534331289
p2: 0.6465271465668712
```

```
Python 3.10.11 (tags/v3.10.11:7d4cc5a, Apr  5 2023, 00:38:17) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
============ RESTART: C:\Users\Raghul\Desktop\ML\EM Algorithm(10).py ============
mu1: 0.9545902456963998
mu2: 1.7595212637782114
sigma1: 0.19986282179149245
sigma2: 0.47713642731204714
p1: 0.3534728534331289
p2: 0.6465271465668712
>>>
```