7. Maximum Likelihood Estimation

In [ ]:
```
import numpy as np
import math
from scipy.optimize import minimize

mu_re = 10
sigma = 20

s = np.random.normal(mu_re, sigma, 1000)

def likelihood(mu, sigma, x):
    return (1 / math.sqrt(2 * math.pi * sigma**2)) * np.exp(-(x - mu)**2 / (2 * sig

def log_likelihood(mu, sigma, data):
    return sum(np.log(likelihood(mu, sigma, x)) for x in data)


neg_log_likelihood = lambda mu: -log_likelihood(mu, sigma, s)

result = minimize(neg_log_likelihood, x0=0.0)

mu_mle = result.x[0]

print(mu_mle)
```

```
9.47181965220062
```

In [ ]:

In [ ]: