1 (a)

$$t = w_0 + w_1 x_1 + \epsilon$$

$$\epsilon : N(0, \sigma^2 = 0.2^2) \Rightarrow \bar{\epsilon} = 0; \ \sigma_\epsilon^2 = 0.2^2; \ E(\epsilon^2) = 0.2^2$$

$$x_1 : \text{Uniform}(-1, 1) \Rightarrow \bar{x}_1 = 0; \ \sigma_{x_1}^2 = \frac{1}{3}; \ E(x_1^2) = \frac{1}{3}$$

$$E(t) = E(w_0) + E(w_1 x_1) + E(\epsilon)$$

$$'' = w_0 + w_1 \underbrace{E(x_1)}_{0} + \underbrace{E(\epsilon)}_{0}$$

$$\boxed{E(t) = w_0}$$

$$\sigma_t^2 = E(t^2) - E(t)^2$$

$$t^2 = w_1^2 x_1^2 + 2 w_0 w_1 x_1 + 2 \epsilon w_1 x_1 + w_0^2 + 2 \epsilon w_0 + \epsilon^2$$

$$E(t^2) = w_1^2 E(x_1^2) + 2 w_1 \underbrace{E(\epsilon x_1)}_{0 \text{ uncorr.}} + w_0^2 + E(\epsilon^2)$$

$$'' = w_1^2 \frac{1}{3} + w_0^2 + 0.2^2$$

$$\boxed{\sigma_t^2 = w_1^2 \frac{1}{3} + 0.2^2}$$

$$\therefore \ \boxed{\begin{array}{l} \bar{t} = w_0 \\ \sigma_t^2 = \underbrace{\frac{w_1^2}{3}}_{w_1^2 \sigma_x^2} + \sigma_\epsilon^2 \end{array}}$$

exact soln

```python
# SOLUTION HW 2
import numpy as np
import matplotlib.pyplot as plt
#------------------------
# DATA GENERATION
NROW = 2
NCOL =2
NPTS = 40


# Exact coeff
w_exact = np.zeros((NROW))
w_exact =[-0.3,0.5]


# Noise signal
e        =np.zeros((NPTS))
e_std    = 0.2
e_var    = e_std**2
e_mean   = 0.0
np.random.seed (1234)
e        = np.random.normal(e_mean,e_std,NPTS)

# input basis vectors
x         = np.zeros((NROW,NPTS))
x1        = np.random.uniform(-1.0,1.0,NPTS)

for i in range(0,NPTS,1):
        x[0][i]    = 1.0
        x[1][i]    = x1[i]

# output vector
t        = np.zeros((NPTS))
for i in range(0,NPTS,1):
        t[i] = e[i]
        for k in range (0,NCOL,1):
                t[i]      = t[i] + w_exact[k]*x[k][i]
####################################################
# part (a)
####################################################
t_mean_approx= np.sum(t  )/NPTS
t_var_approx = np.sum(t*t)/NPTS - t_mean_approx**2
print(' HW2 Part (a)')
print('approx mean(exact): ', t_mean_approx,'(',w_exact[0],')')
print('approx var (exact): ', t_var_approx,'(',w_exact[1]**2/3 + e_var,')')

####################################################
# part (b,c)
# MLE solution
####################################################
phi_trans = np.zeros((NROW,NPTS))
for i in range(0,NPTS,1):
        phi_trans[0][i] = x[0][i]
        phi_trans[1][i] = x[1][i]

A = np.matmul(phi_trans,np.transpose(phi_trans))
B = np.matmul(phi_trans,t)
w_approx = np.linalg.solve(A,B)

print(' HW2 Part (b)')
print  ('MLE approx (exact)', w_approx,'(',w_exact,')')
print  ('condition number A',"%.4f" % np.linalg.cond(A))
```



```python
####################################################

x1_mean_approx= np.sum(x1  )/NPTS
x1_var_approx = np.sum(x1*x1)/NPTS - x1_mean_approx**2
print('approx mean(var) x1: ', x1_mean_approx,'(',x1_var_approx,')')

e_mean_approx= np.sum(e  )/NPTS
e_var_approx = np.sum(e*e)/NPTS - e_mean_approx**2
print('approx mean(var) e: ', e_mean_approx,'(',e_var_approx,')')

ez= np.sum ( (e-e_mean_approx)*(x1-x1_mean_approx))/NPTS
print('approx mean(var) ez: ', ez,'(',0,')')


exit()
```

Department of Electrical and Computer Engineering
University of Massachusetts Lowell
EECE 5440 Computational Data Modeling I

Assignment #2

1. Consider the system

$$t = w_0 + w_1 x_1 + \varepsilon$$

where $w_0 = -0.3$ and $w_1 = 0.5$. The random variable ε is drawn from the normal distribution with mean 0 and variance 0.2^2. The random variable x_1 is uniformly distributed between (-1,1). Generate the observation t for 40 trials.

a. Using the trial values for t find estimates for the mean and variance of t. Compare the approximate to the exact result.

b. Using the observations for the j^{th} trial t_j and input $\underline{x}^{(j)} = [1, x_1^{(j)}]$ for $j = (0, 39)$. Find $\underline{w}$ using the MLE approach .

c. Compare the computed and exact result for $\underline{w} = [w_0, w_1]^T$.