

Chapter.9 確率微分方程式の離散時間近似

伊藤克哉

東京大学

2016/10/03

講義ノート スライド コードは

<https://github.com/KatsuyaITO/NSofSDE>

- 1 確率論からの入門
- 2 確率微分方程式の数値解析
- 3 近似解の実装

情報増大系と適合

以下では (Ω, \mathcal{F}, P) という確率空間を考える.

定義 1.1

$(\mathcal{F}_t)_{t \geq 0}$ が**情報増大系**であるとは,

$$\mathcal{F}_t \subset \mathcal{F} : \text{sub } \sigma - \text{alg} \text{ かつ } 0 \leq s \leq t \Rightarrow \mathcal{F}_s \subset \mathcal{F}_t$$

となることである.

また d 次元確率過程 $X = (X_t)_{t \geq 0}$ が情報増大系 $(\mathcal{F}_t)_{t \geq 0}$ に対して**適合 (adapted)** しているとは, $\forall t \geq 0 : X_t : \Omega \rightarrow \mathbb{R}^d$ が \mathcal{F}_t 可測であるということである.

Brown 運動

定義 1.2

確率過程 $B = (B_t)_{t \geq 0}$ が実数値 *Brown 運動* であるとは, 次を満たすことである.

- (i) $B_0 = 0$ a.s.
- (ii) $\forall \omega \in \Omega : B_t(\omega)$ は連続である.
- (iii) $0 < t_0 < t_1 < \cdots < t_n$ という任意の細分に対して, $\{B_{t_j} - B_{t_{j-1}}\}_i$ は互いに独立で $N(0, t_j - t_{j-1})$ に従う.

マルチンゲール

定義 1.3

右連続な確率過程 $X = (X_t)_{t \geq 0}$ が $(\mathcal{F}_t)_{t \geq 0}$ -マルチンゲールであるとは、次を満たすことである.

(i) $\forall t \geq 0: E[|X_t|] < \infty$

(ii) X は $(\mathcal{F}_t)_t$ 適合である.

(iii) $0 \leq \forall s \leq t: E[X_t | \mathcal{F}_s] = X_s$ a.s. である.

伊藤積分

定義 1.4 (伊藤積分)

次のようにして確率過程の族を表す.

$$\mathbb{L}_T^2 := \{f : \text{確率過程} \mid f_t(\omega) \text{ は可測で } \|f\|_{\mathbb{L}_T^2} := E[\int_0^T f_t^2 dt] < \infty\}$$
$$\mathcal{L}_T^2 = \mathcal{L}_T^2(\mathcal{F}_t) := \{f \in \mathbb{L}_T^2 \mid f \text{ は } (\mathcal{F}_t) \text{ 適合}\}$$

伊藤積分の続き

$f = (f_t)$ は次のように表される階段過程であるとする.

$$f_t = \sum_{j=1}^n \tilde{f}_j 1_{[t_{j-1}, t_j)}(t), \quad t \in [0, T]$$

(ただし, \tilde{f}_j は $\mathcal{F}_{t_{j-1}}$ 可測で有界な確率変数)

このとき, 確率過程 f の確率積分を,

$$M_t(f) \equiv \int_0^t f_s dB_s := \sum_{j=1}^n \tilde{f}_j (B_{t \wedge t_j} - B_{t \wedge t_{j-1}})$$

として定める.

また一般の可測な (\mathcal{F}_t) - 適格な確率過程 f に対して, f^n という階段過程の列が存在して, $\|f - f^{(n)}\|_{\mathbb{L}_T^2} \rightarrow 0, n \rightarrow \infty$ とできる.

故に, f の確率積分を

$$\int_0^t f_s dB_s := \lim_{n \rightarrow \infty} M(f^{(n)})$$

によって定める. これは \mathcal{M}_T の元として $f^{(n)}$ のとり方によらず一意的に定まる.

そしてこれを $[0, \infty)$ に拡張して確率積分を定義することもできる. また同様にして次の確率過程の族を定める.

$$\mathcal{L}^2(\mathcal{F}_t) := \{f = (f_t)_{t \geq 0} : \text{確率過程} \mid \forall T > 0 : (f_t)_{t \in [0, T]} \in \mathcal{L}_T^2\}$$

伊藤積分の性質

定理 1.5

確率積分 $M_t(f) = \int_0^t f_s dB_s$ は次をみたす.

(1) $M_t(f)$ は \mathcal{F}_t マルチンゲールである.

(2) $E(M_t(f)) = 0$

(3) $E(M_t(f)^2) = \int_0^T E(f(t, -)^2) dt$

(4) $I(af + bg) = aI(f) + bI(g)$ a.s.

確率微分方程式

定義 1.6

確率過程 $X = (X_t)_{t \geq 0}$ が次を満たすとき, **確率微分方程式 (stochastic differential equation)**

$$dX_t = a(t, X_t)dt + b(t, X_t)dW_t \quad X_{t_0} = X_0 \quad (1.1)$$

の解であるという.

X は (Ω, \mathcal{F}, P) 上で定義された \mathcal{F}_t 適合かつ可測な \mathbb{R} 値連続確率過程で,

(i)

$$a(t, X_t) \in L^1_{loc}([0, \infty)), \quad b(t, X_t) \in L^2(\mathcal{F}_t) \text{ a.s.}$$

(ii) 確率積分方程式

$$X_t = X_0 + \int_0^t a(s, X_s)ds + \int_0^t b(s, X_s)dB_s$$

を満たす.

解の一意性

定義 1.7

$[t_0, T]$ 上の確率微分方程式が **pathwise unique** な解を持つとは、
任意の 2 組の解 X_t, \tilde{X}_t が、

$$P\left(\sup_{t_0 \leq t \leq T} |X_t - \tilde{X}_t| > 0\right) = 0$$

を満たすということである。

定理 1.8 (解の一意性)

 $[t_0, T]$ 上の確率微分方程式

$$dX_t = a(t, X_t)dt + b(t, X_t)dW_t, \quad X_{t_0} = X_0$$

は次の 4 条件を満たすとき

$$\sup_{t_0 \leq t \leq T} E(|X_t|^2) < \infty$$

を満たすような *pathwise unique* な強解 X_t を $[t_0, T]$ 上持つ

(A1) (可測性) $a(t, x), b(t, x)$ は $[t_0, T] \times \mathbb{R}$ で L^2 可測.

(A2)(Lipschitz 条件) 次を満たす定数 $K > 0$ が存在する.

$$|a(t, x) - a(t, y)| \leq K|x - y|$$

$$|b(t, x) - b(t, y)| \leq K|x - y|$$

(A3) 次を満たす定数 $K > 0$ が存在する.

$$|a(t, x)|^2 \leq K^2(1 + |x|^2)$$

$$|b(t, x)|^2 \leq K^2(1 + |x|^2)$$

(A4) X_{t_0} は \mathcal{F}_{t_0} 可測で $E(|X_{t_0}|^2) < \infty$ を満たす.

定理 1.9 (伊藤の公式)

$U: [0, T] \times \mathbb{R} \rightarrow \mathbb{R}$ は C^2 級であるとする.
 X_t は次の確率微分方程式の解であるとする.

$$dX_t = e(t, \omega)dt + f(t, \omega)dW_t(\omega)$$

このとき, $Y_t = U(t, X_t)$ は

$$\begin{aligned} Y_t - Y_s = & \int_s^t \left\{ \frac{\partial U}{\partial t}(u, X_u) + e_u \frac{\partial U}{\partial X}(u, X_u) + \frac{1}{2} f_u^2 \frac{\partial^2 U}{\partial X^2}(u, X_u) \right\} du \\ & + \int_s^t f_u \frac{\partial U}{\partial X}(u, X_u) dW_u \end{aligned}$$

を *almost surely* に満たす

```
class Process:
    def show(self):
        plt.plot(self.show_x, self.show_y)
    def at(self, t):
        if t in self.dic:
            return self.dic[t]
        else:
            return self.show_y[
                np.searchsorted(self.show_x, t)]
```



```
class Wiener(Process):
    def __init__(self, start, end, n):
        t = list(np.linspace(start, end, n+1))
        w = [0]
        sigma = math.sqrt(t[1]-t[0])
        for tn in t[:-1]:
            w.append(w[-1]+
                    np.random.normal(0, sigma))
        self.start = start
        self.end = end
        self.t = t
        self.w = w
        self.dic = dict(zip(t, w))
        self.show_x = t
        self.show_y = w
```

```
class Solution(Process):  
    def __init__(self, t, f, w):  
        self.x = [f(tn, w) for tn in t]  
        self.f = f  
        self.w = w  
        self.t = t  
        self.dic = dict(zip(t, self.x))  
        self.show_x = t  
        self.show_y = self.x
```

```
class Euler_Maruyama(Process):
    def __init__(self, a, b, x0, w, delta):
        t = list(np.arange(w.start, w.end+delta, delta))
        y = [x0]
        l = len(t)

        for i in range(l-1):
            y.append(y[-1]
                    +a(t[i], y[-1])*(t[i+1]-t[i])
                    +b(t[i], y[-1])*(w.at(t[i+1])-w.at(t[i])))

        self.t = t
        self.w = w
        self.a = a
        self.b = b
        self.x0 = x0
        self.delta = delta
```

PC-Exercise-9.2.1

区間 $[0, 1]$ に於いて等間隔 $\Delta = 2^{-2}$ のオイラー近似を作成し, $dX_t = 1.5X_t dt + 1.0X_t dW_t$, $X_0 = 1.0$ という確率微分方程式を近似せよ.

```
a921 = lambda t,x: 1.5*x
```

```
b921 = lambda t,x: 1.0*x
```

```
delta921 = 2**(-2)
```

```
x0_921 = 1.0
```

```
t0_921 = 0
```

```
t1_921 = 1
```

```
sol921 = lambda t,w: math.exp(t+w.at(t))
```

```
W = Wiener(t0_921,t1_921,2**9)
```

```
Y = Euler_Maruyama(a921,b921,x0_921,W,delta921)
```

```
X = Solution(W.t,sol921,W)
```

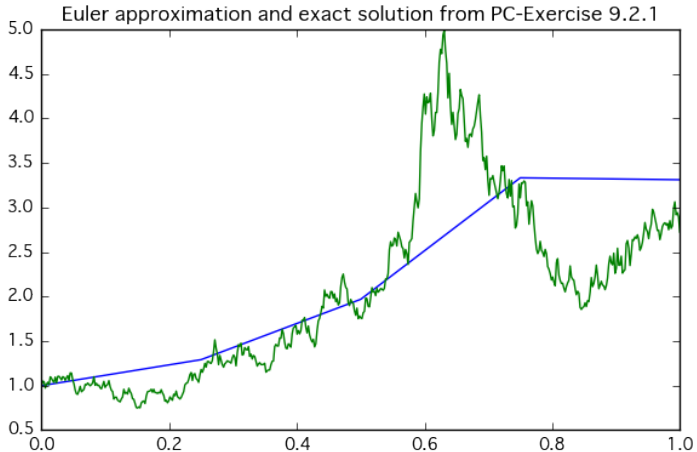
```
plt.title("Euler approximation and exact solution from
```

```
Y.show()
```

```
X.show()
```

```
plt.show()
```

```
plt.close()
```



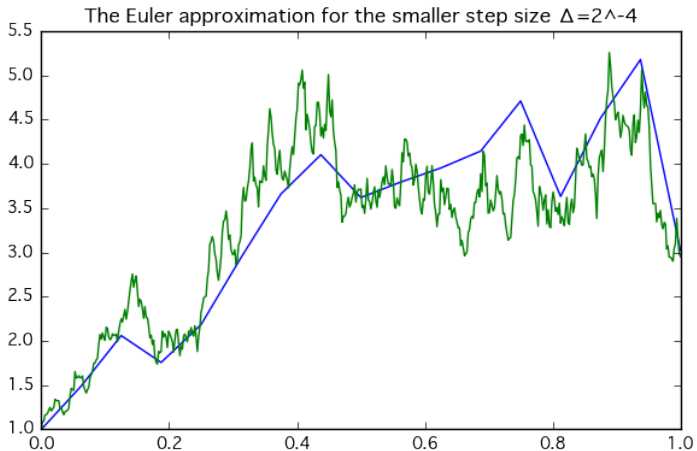
PC-Exercise 9.2.2

PC-Exercise 9.2.1 を間隔を $\Delta = 2^{-4}$ にして繰り返せ.

```
delta922 = 2**(-4)
W = Wiener(t0_921,t1_921,2**9)
Y = Euler_Maruyama(a921,b921,x0_921,W,delta922)
X = Solution(W.t,sol921,W)
```

```
plt.title("The Euler approximation for the step size  $\Delta t$ ")
```

```
Y.show()
X.show()
plt.show()
plt.close()
```

PC-Exercise 9.3.1

$$dX_t = 1.5X_t dt + 1.0X_t dW_t, \quad X_0 = 1.0$$

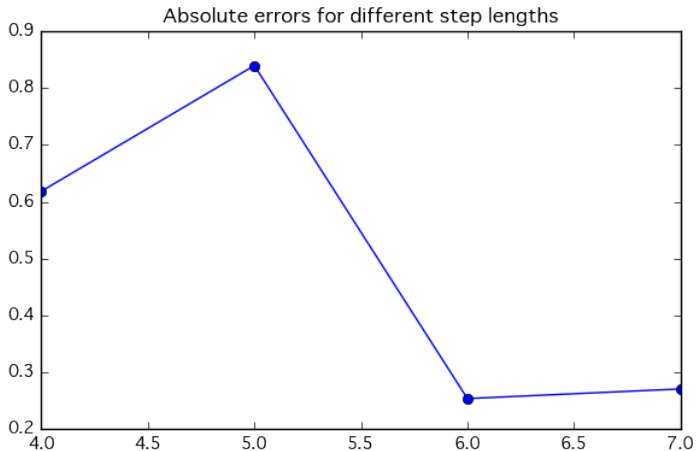
という $[0, 1]$ 上の伊藤過程に対して $\Delta = 2^{-4}$ の等間隔のオイラー近似を $N = 25$ 回繰り返し, 統計的誤差 $\hat{\epsilon}$ を計算せよ. これを $\Delta = 2^{-5}, 2^{-6}, 2^{-7}$ についても繰り返し Δ と $\hat{\epsilon}$ の関係を示せ.

```
N931 = 100
Deltas = [2**(-4), 2**(-5), 2**(-6), 2**(-7)]
LogDeltas = [4, 5, 6, 7]
epsilons = []
for delta in Deltas:
    eps = []

    for i in range(N931):
        W = Wiener(t0_921, t1_921, 2**9)
        Y = Euler_Maruyama(a921, b921, x0_921, W, delta)
        X = Solution(W.t, sol921, W)
        eps.append(math.fabs(X.at(1) - Y.at(1)))

    epsilons.append(np.mean(eps))

plt.title("Absolute errors for different step lengths")
plt.plot(LogDeltas, epsilons, "-o")
plt.show()
```



PC-Exercise 9.3.3

$$dX_t = 1.5X_t dt + 0.1X_t dW_t, X_0 = 1.0$$

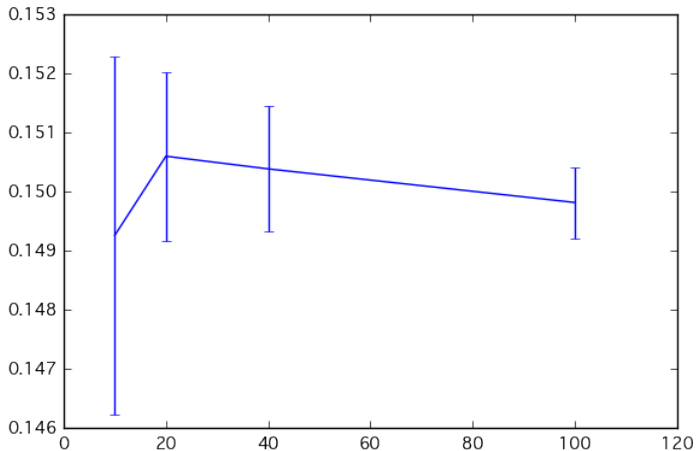
という $[0, 1]$ 上の伊藤過程に対して $\Delta = 2^{-4}$ の等間隔のオイラー近似を $N = 25$ 回繰り返し, 更にこれを $M = 10$ 組繰り返しことによって, 絶対誤差 ϵ の 90% を $M = 20, 40$ and 100 についても繰り返し Δ と ϵ の信頼区間との関係を示せ.

```

Ms = [10,20,40,100]
N = 100
b933 = lambda t,x: 0.1*x
sol933 = lambda t,w: math.exp(1.495*t+0.1*w.at(t))

all_eps = []
all_deltaeps = []
for M in Ms:
    print(M)
    epsilons = []
    for i in range(M):
        eps = []
        for j in range(N):
            W = Wiener(t0_921,t1_921,2**9)
            Y = Euler_Maruyama(a921,b933,x0_921,W,delt
            eps.append(math.fabs(sol933(1,W) - Y.at(1)
        epsilons.append(np.mean(eps))
    all_eps.append(np.mean(epsilons))

```



PC-Exercise 9.3.4

$$dX_t = 1.5X_t dt + 0.1X_t dW_t, \quad X_0 = 1.0$$

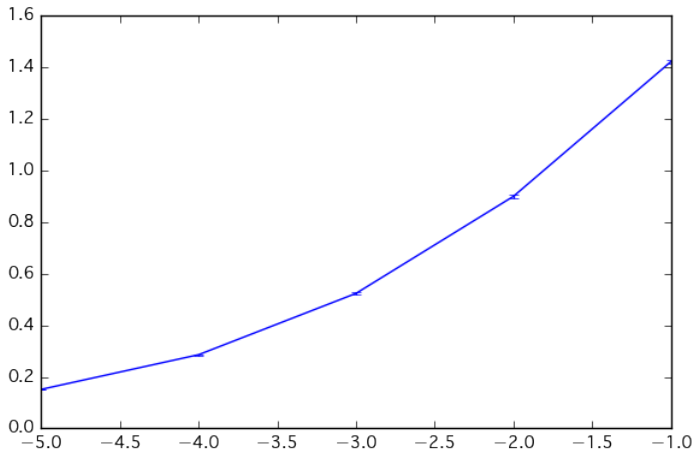
という $[0, 1]$ 上の伊藤過程に対して $\Delta = 2^{-2}$ の等間隔のオイラー近似を $N = 25$ 回繰り返し, 更にこれを $M = 100$ 組繰り返しことによって, 絶対誤差 ϵ の 90 % 信頼区間を図示せよ. これを, $\Delta = 2^{-5}, 2^{-6}$ and 2^{-7} についても繰り返し Δ と ϵ の関係を示せ.


```

Deltas = [2**(-1), 2**(-2), 2**(-3), 2**(-4), 2**(-5)]
LogDeltas = [-1, -2, -3, -4, -5]
N = 100
M = 20

all_eps = []
all_deltaeps = []
for delta in Deltas:
    print(delta)
    epsilons = []
    for i in range(M):
        eps = []
        for j in range(N):
            W = Wiener(t0_921, t1_921, 2**10)
            Y = Euler_Maruyama(a921, b933, x0_921, W, delta)
            eps.append(math.fabs(sol933(1, W) - Y.at(1)))
        epsilons.append(np.mean(eps))
    all_eps.append(np.mean(epsilons))

```



PC-Exercise 9.4.1

$dX_t = 1.5X_t dt + 0.1X_t dW_t$, $X_0 = 1.0$ という $[0, 1]$ 上の伊藤過程に対して $\Delta = 2^{-4}$ の等間隔のオイラー近似を $N = 100$ 回繰り返す, 更にこれを $M = 20, 40, 100$ 組繰り返すことによって, 平均の誤差 μ の 90 % 信頼区間を図示せよ.

```
Ms = [10,20,40,100]
```

```
N = 100
```

```
delta = 2**(-4)
```

```
mus = []
```

```
delta_mu = []
```

```
for M in Ms:
```

```
    print(M)
```

```
    mu_j = []
```

```
    for i in range(M):
```

```
        Y_t_j = []
```

```
        for j in range(N):
```

```
            W = Wiener(t0_921,t1_921,2**9)
```

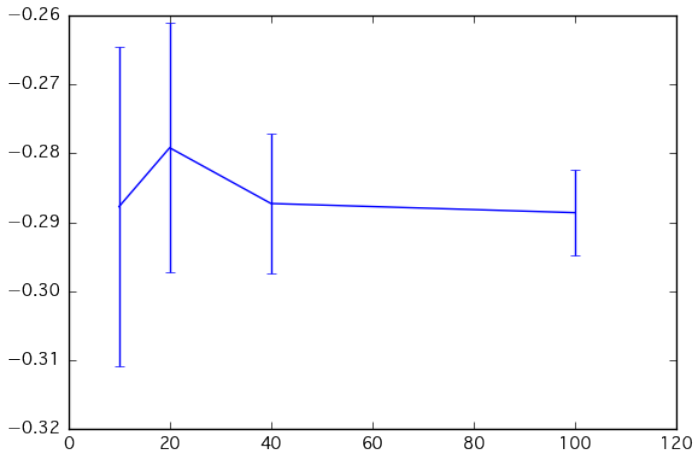
```
            Y = Euler_Maruyama(a921,b933,x0_921,W,delta)
```

```
            Y_t_j.append(Y.at(1))
```

```
        mu_j.append(np.mean(Y_t_j)-math.exp(1.5))
```

```
    mus.append(np.mean(mu_j))
```

```
    delta_mu.append(stats.t.ppf(1-(1-0.9)/2, M-1))
```



$dX_t = 1.5X_t dt + 0.1X_t dW_t$, $X_0 = 1.0$ という $[0, 1]$ 上の伊藤過程に対して, $M = 20, N = 100$ として, $\Delta = 2^{-3}, \dots, 2^{-5}$ に対して μ の信頼区間を示せ.

```

Deltas = [2**(-1), 2**(-2), 2**(-3), 2**(-4), 2**(-5)]
LogDeltas = [-1, -2, -3, -4, -5]
N = 100
M = 20

```

```

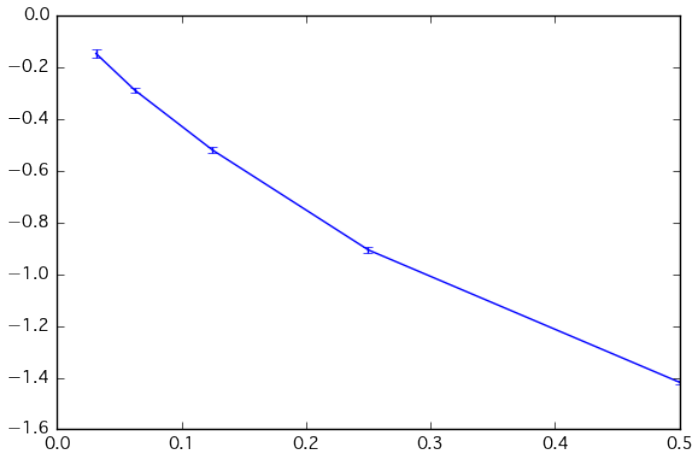
mus = []
delta_mu = []

```

```

for delta in Deltas:
    print(M)
    mu_j = []
    for i in range(M):
        Y_t_j = []
        for j in range(N):
            W = Wiener(t0_921, t1_921, 2**9)
            Y = Euler_Maruyama(a921, b933, x0_921, W, delta)
            Y_t_j.append(Y.at(1))
        mu_j.append(np.mean(Y_t_j) - math.exp(1.5))

```



PC-Exercise9.8.2

$$dX_t = 5X_t dt + dW_t, \quad X_0 = 1.0$$

という $[0, 1]$ 上の伊藤過程に対して $\Delta = 2^{-4}$ の等間隔のオイラー近似をせよ.

```
a982 = lambda t,x: 5*x
```

```
b982 = lambda t,x: 1
```

```
delta982 = 2**(-4)
```

```
x0_921 = 1.0
```

```
t0_921 = 0
```

```
t1_921 = 1
```

```
sol921 = lambda t,w: math.exp(t+w.at(t))
```

```
W = Wiener(t0_921,t1_921,2**10)
```

```
Y = Euler_Maruyama(a982,b982,x0_921,W,delta921)
```

```
X = Euler_Maruyama(a982,b982,x0_921,W,2**(-10))
```

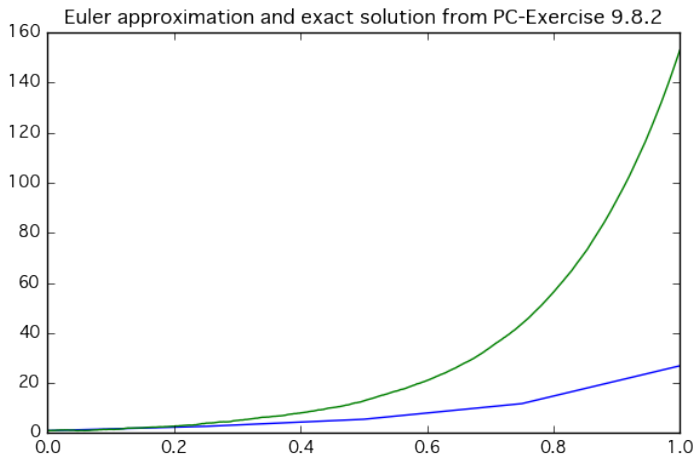
```
plt.title("Euler approximation and exact solution from
```

```
Y.show()
```

```
X.show()
```

```
plt.show()
```

```
plt.close()
```



PC-Exercise9.8.3

$$dX_t = -5X_t dt + dW_t, \quad X_0 = 1.0$$

という $[0, 1]$ 上の伊藤過程に対して $\Delta = 2^{-4}$ の等間隔のオイラー近似をせよ.

```
a982 = lambda t,x: -5*x
```

```
b982 = lambda t,x: 1
```

```
delta982 = 2**(-4)
```

```
x0_921 = 1.0
```

```
t0_921 = 0
```

```
t1_921 = 1
```

```
sol921 = lambda t,w: math.exp(t+w.at(t))
```

```
W = Wiener(t0_921,t1_921,2**10)
```

```
Y = Euler_Maruyama(a982,b982,x0_921,W,delta921)
```

```
X = Euler_Maruyama(a982,b982,x0_921,W,2**(-10))
```

```
plt.title("Euler approximation and exact solution from
```

```
Y.show()
```

```
X.show()
```

```
plt.show()
```

```
plt.close()
```

