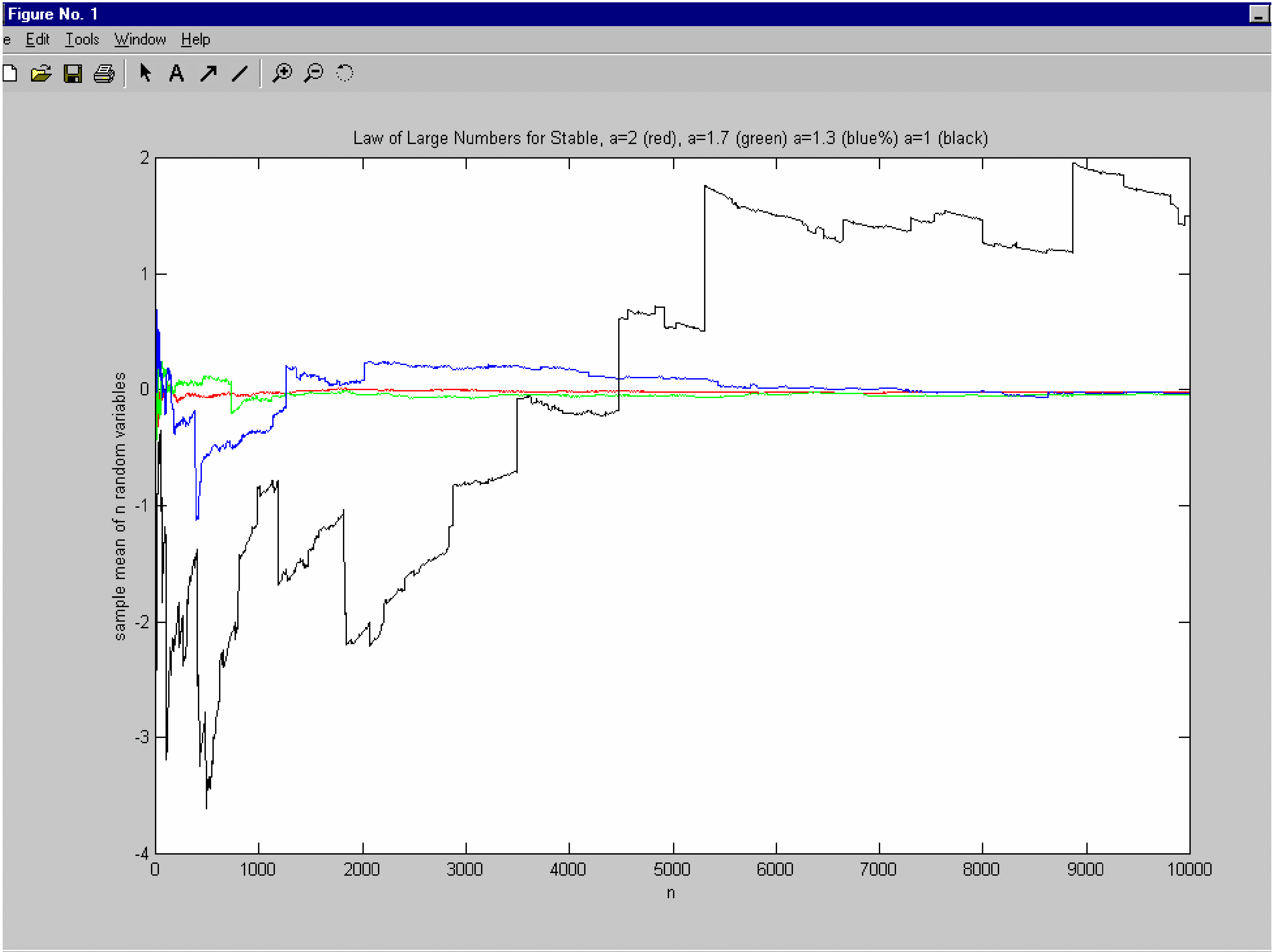


Figure No. 1
e Edit Tools Window Help
Law of Large Numbers for Stable, a=2 (red), a=1.7 (green) a=1.3 (blue%) a=1 (black)
sample mean of n random variables
2
1
0
-1
-2
-3
-4
0
1000
2000
3000
4000
5000
6000
7000
8000
9000
10000
n

Regression in R

- To fit a linear model with two predictors x1 x2,

out <- lm(y ~ x1 + x2)

- To generate the ANOVA table “summary(out)” produces
- Call:
- lm(formula = y ~ x1 + x2)
- Residuals:
- Min 1Q Median 3Q Max
- -1.4445 -0.7684 0.1912 0.5417 1.2968

- Coefficients:
- Estimate Std. Error t value Pr(>|t|)
- (Intercept) 1.22953 0.57489 2.139 0.0473 *
- x1 0.96293 0.03337 28.853 7.00e-16 ***
- x2 0.87006 0.07733 11.251 2.68e-09 ***
- ---
- Signif. codes: 0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

- Residual standard error: 0.8504 on 17 degrees of freedom
- Multiple R-Squared: 0.9846, Adjusted R-squared: 0.9827
- F-statistic: 542.1 on 2 and 17 DF, p-value: 4.007e-16

Estimating the Risk of the James-Stein Estimator

Suppose X is multivariate $MVN(\theta, I_p)$

We wish to use an estimator of the form

$$\delta(\mathrm{X}) = (1 - \frac{\mathrm{a}}{\|\mathrm{X}\|^2})^+ X \text{ for } 0 < a < 2(p-2)$$

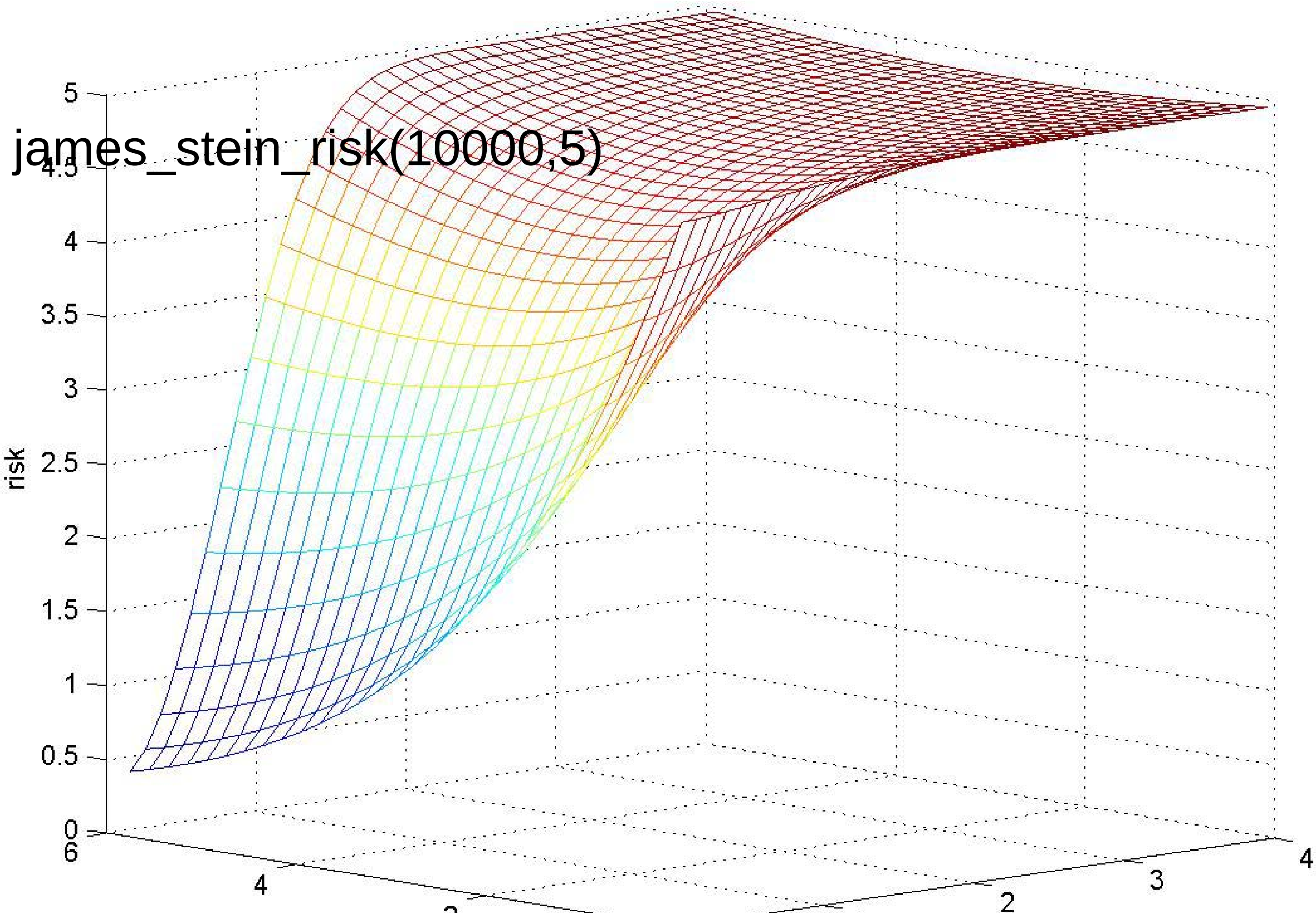
where $\dim(\mathrm{X}) = \mathrm{p}$

Robert&Casella, p.110-112

Matlab Function for James-Stein Risk

```
• function james_stein_risk(nsim,p)
• theta=.1:.1:4; a=.1:.2:2*(p-2); ntheta=length(theta); na=length(a);
• x=randn(nsim,p);
• risk=zeros(ntheta,na);
• for i=1:ntheta
•     for j=1:na
•         z=theta(i)+x;
•         m=max(0,(1-a(j)./sum(z'.^2)));
•         m=repmat(m',1,p);
•         d=z.*m;
•         risk(i,j)=sum(mean((d-theta(i)).^2));
•     end
• end
• mesh(repmat(theta',1,na),repmat(a,ntheta,1),risk)
• xlabel('theta')
• ylabel('a')
• zlabel('risk')
```

James-Stein Squared error risk, p=5



Simulating a time-homogeneous Poisson process

For time - homogeneous Poisson process the times between arrivals are independent exp(λ)

EXAMPLE: Generate Poisson(λ) on interval [0,10].

1. set $t = 0, i = 1$

2. Generate U_i uniform[0,1]

3. Set $t = T_i = t - \frac{1}{\lambda}\ln(U_i)$ (time of i'th arrival)

4. If $T_i < 10$, put $i = i + 1$, go to 2, otherwise stop

A Non-homogeneous Poisson Process

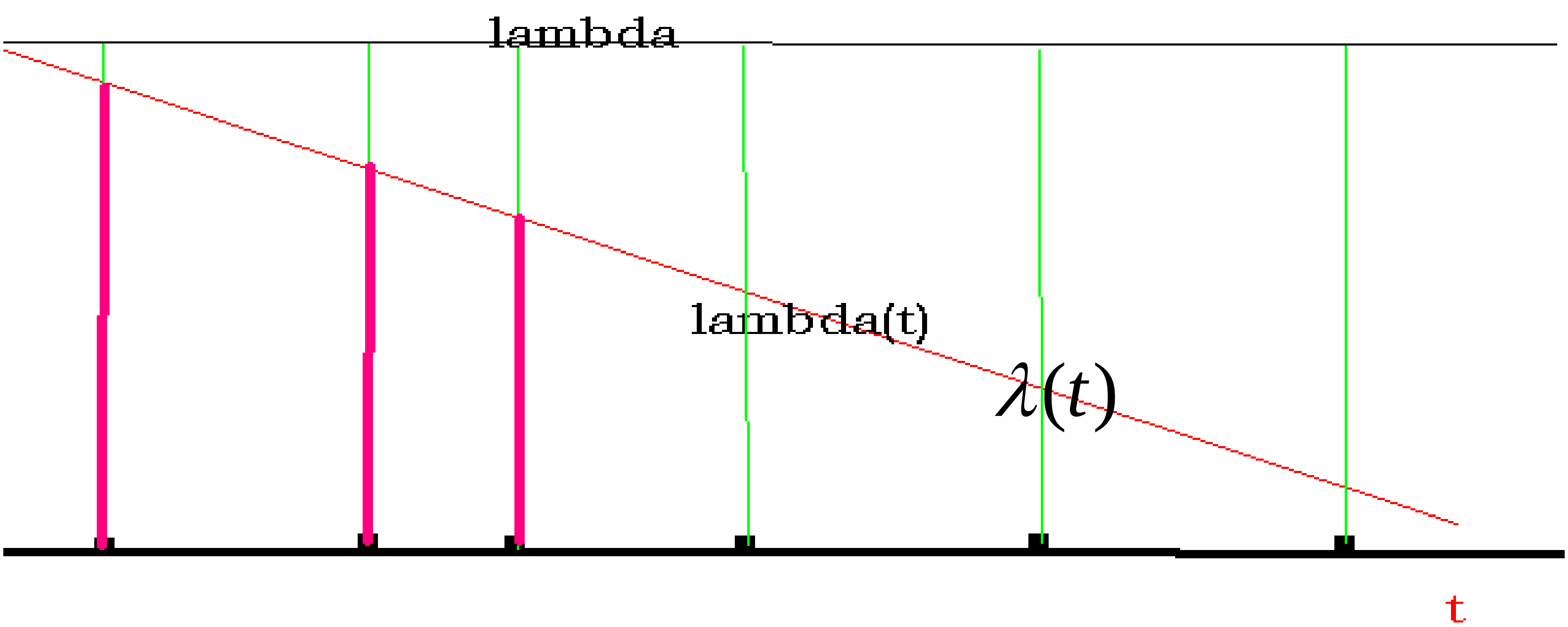
- Suppose arrivals at a ticket office begin at time t=7 and the intensity of arrivals occur at rate

$$\lambda(t) = 100 - 10(t-7), \text{ for } 7 < t < 12.$$

Arrivals occur at a rate higher after 7 a.m than later in the morning and the rate drops off linearly until noon. How do we simulate such a process?

Simulating a non-homogeneous Poisson process

- Suppose we first simulate a Poisson process with intensity $\lambda \geq \lambda(t)$ for all t.



Simulating a non-homogeneous Poisson process

Let T_i be arrival times of homogeneous poisson process with intensity λ.

For point at T_i ``accept'' point with probability $\frac{\lambda(T_i)}{\lambda}$.

Then the "thinned" process of accepted points has intensity $\lambda(t)$.

Algorithm for non-homogeneous Poisson process intensity

$$\lambda(t) = 100 - 10(t-7), \text{ for } 7 < t < 12.$$

1. set $\mathrm{t} = 7, \mathrm{i} = 1$
2. Generate $\mathrm{U_i}$ *uniform*[0,1]
3. Set $t = t - \frac{1}{100}\ln(U_i)$.
4. If $\mathrm{t} > 12$, STOP.
5. Generate $\mathrm{V_i}$ uniform[0,1]
6. If $\mathrm{V_i} > \frac{100 - 10(t-7)}{100}$, go to 2

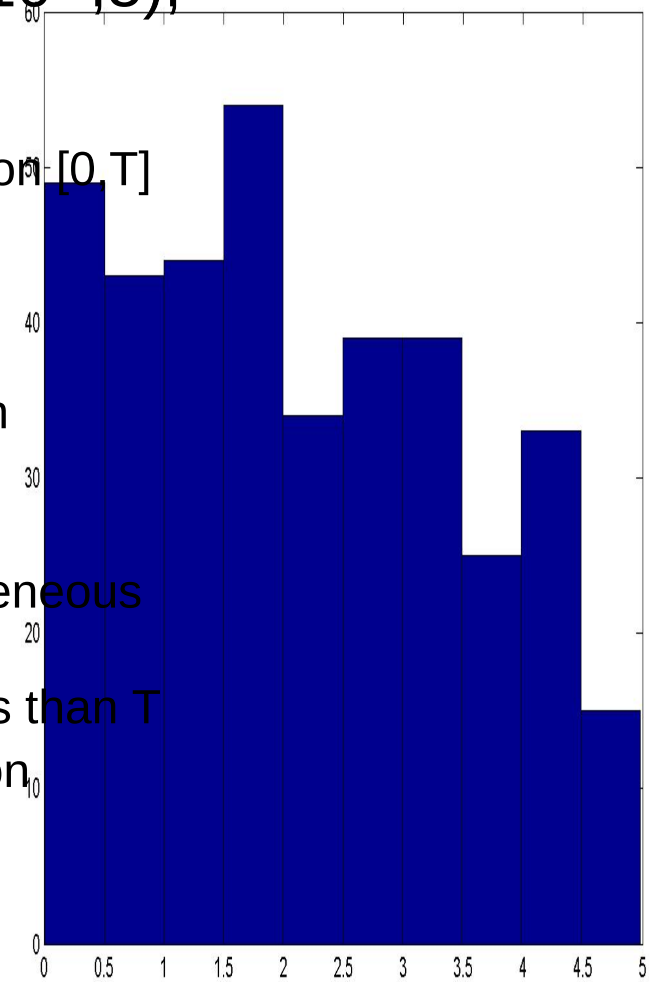
(i.e. reject as an arrival time and proceed to next time)

7. Otherwise, $\mathrm{T_i} = t$ (accept time of arrival), $\mathrm{i} = \mathrm{i} + 1$
8. Go to 2.

Matlab code for non-homogeneous Poisson process

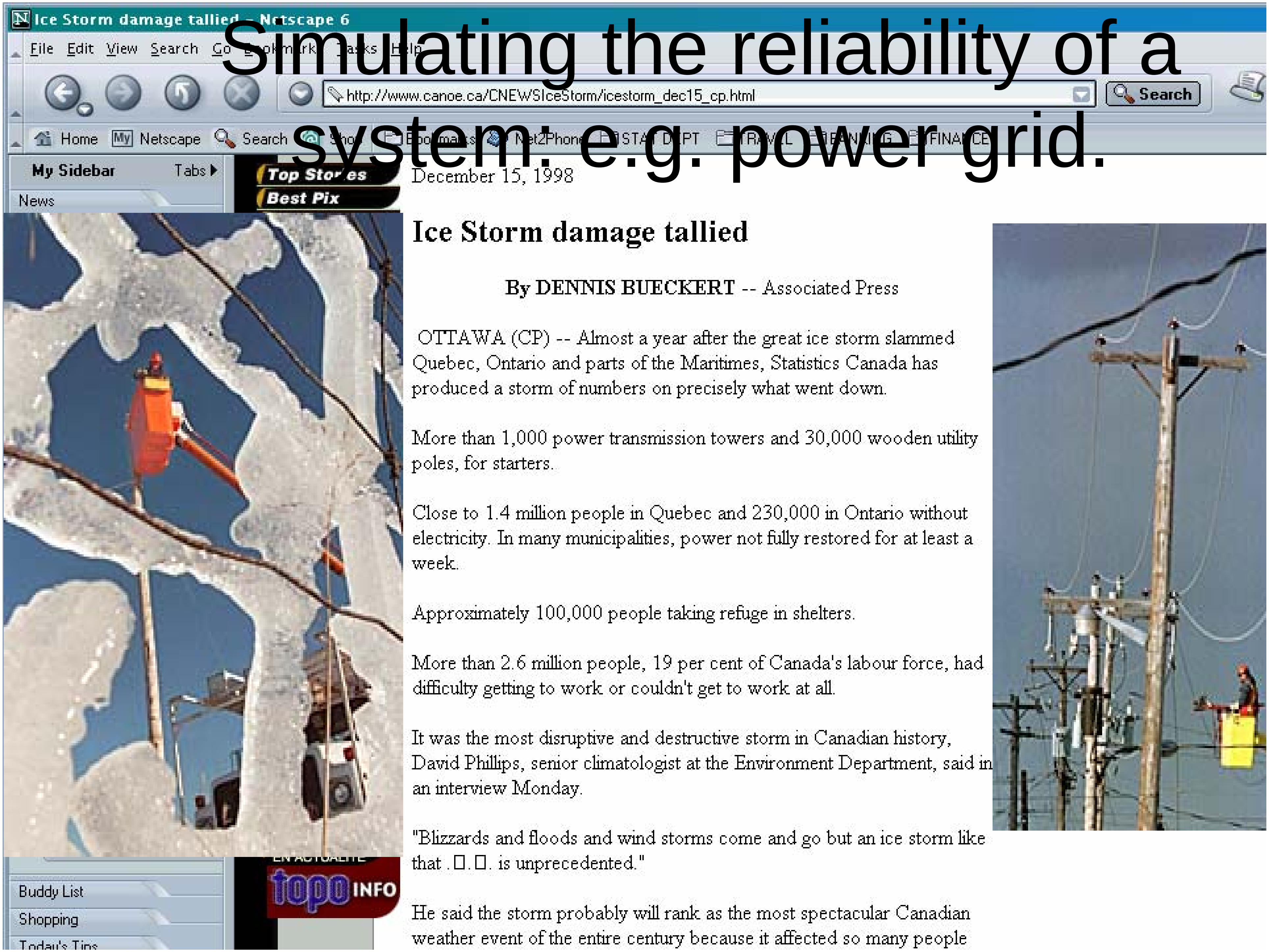
$$\lambda(t) = 100 - 10(t-7), \text{ for } 7 < t < 12.$$

t=7+nonhomopp('100-10*',5);



- function x=nonhomopp(intens,T)
- %generate a nonhomogeneousl poisson process on [0,T] with intensity function intens
- x=0:.1:T;
- l=eval([intens 'x']);
- lam0=max(l); % generate homogeneouos poisson process
- u=rand(1,ceil(1.5*T*lam0));
- x=cumsum(-(1/lam0)*log(u));   %points of homogeneous pp
- x=x(x<T); n=length(x);   % select those points less than T
- l=eval([intens 'x']);   % evaluates intensity function
- x=x(rand(1,n)<l/lam0);   % filter out some points
- hist(x,10)

Simulating the reliability of a system: e.g. power grid.



December 15, 1998

Ice Storm damage tallied

By DENNIS BUECKERT -- Associated Press

OTTAWA (CP) -- Almost a year after the great ice storm slammed Quebec, Ontario and parts of the Maritimes, Statistics Canada has produced a storm of numbers on precisely what went down.

More than 1,000 power transmission towers and 30,000 wooden utility poles, for starters.

Close to 1.4 million people in Quebec and 230,000 in Ontario without electricity. In many municipalities, power not fully restored for at least a week.

Approximately 100,000 people taking refuge in shelters.

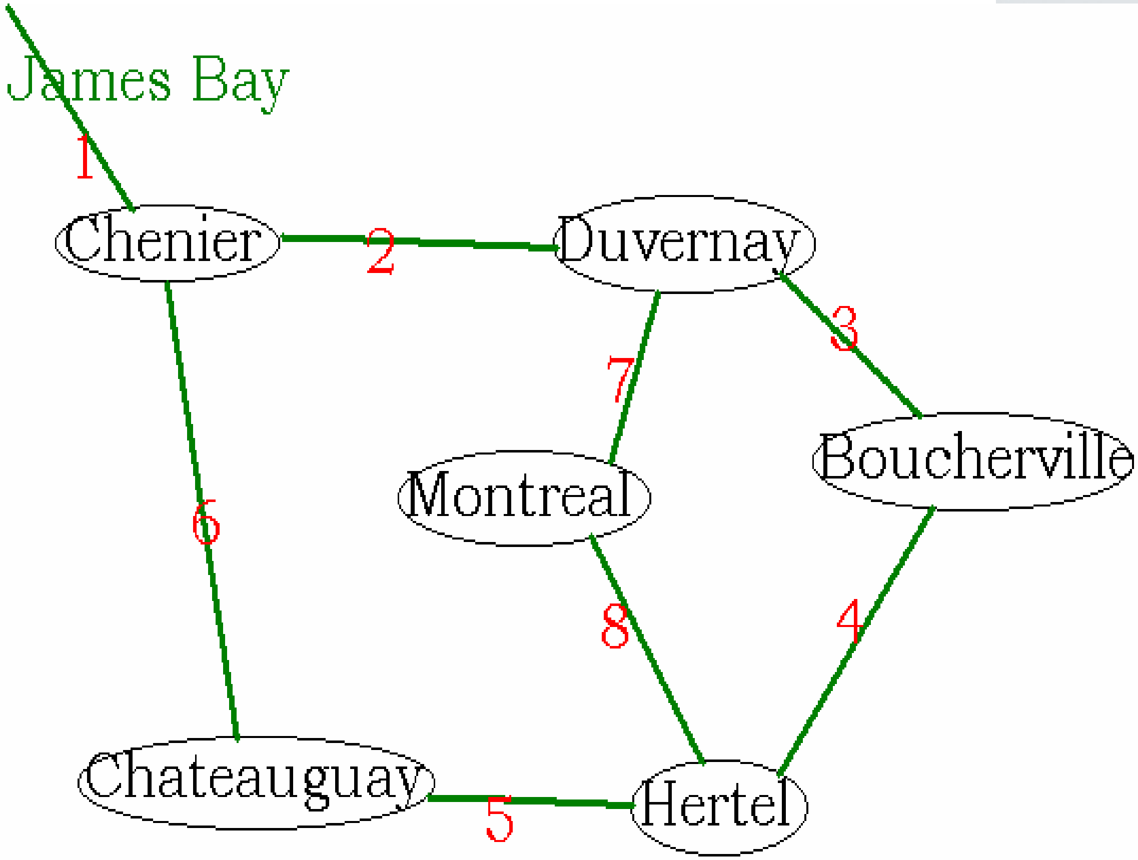
More than 2.6 million people, 19 per cent of Canada's labour force, had difficulty getting to work or couldn't get to work at all.

It was the most disruptive and destructive storm in Canadian history, David Phillips, senior climatologist at the Environment Department, said in an interview Monday.

"Blizzards and floods and wind storms come and go but an ice storm like that . . . is unprecedented."

He said the storm probably will rank as the most spectacular Canadian weather event of the entire century because it affected so many people

Montreal Power Grid

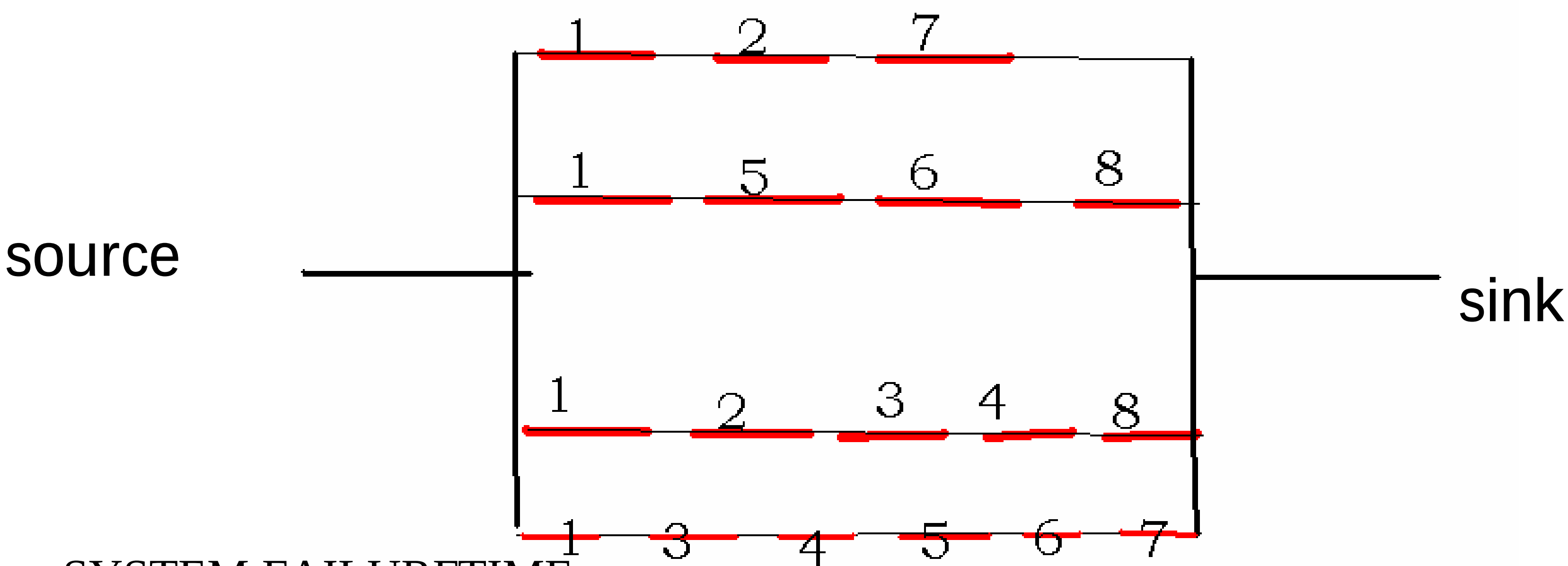


Montreal receives power if

- 1,2,7 or
- 1,6,5,8 or
- 1,2,3,4,8 or
- 1,6,5,4,3,7 are all operating. Suppose times until failure of each of these components under certain circumstance (e.g. ice storm) independent

- System still operating at time t if

$$\min(T_1, T_2, T_7) > t \quad \text{or}$$
$$\min(T_1, T_6, T_5, T_8) > t \quad \text{or}$$
$$\min(T_1, T_2, T_3, T_4, T_8) > t$$
$$\min(T_1, T_6, T_5, T_4, T_3, T_7) > t$$
$$\text{where } T_i = \text{ failure time of link i.}$$



SYSTEM FAILURETIME =

$$\max(\min(T_1, T_2, T_7), \min(T_1, T_6, T_5, T_8), \min(T_1, T_2, T_3, T_4, T_8), \min(T_1, T_6, T_5, T_4, T_3, T_7))$$

Simulation of Montreal power grid

- function L=lifepowr(T)
- % input vector T of length 8, lifetimes of components of montreal
- % power grid. Outputs lifetime L of the system.
- L1=min(T([1 2 7]));
- L2=min(T([1 5 6 8]));
- L3=min(T([1 2 3 4 8]))
- L4=min(T([1 3 4 5 6 7]));
- L=max([L1,L2,L3,L4]);

Run Montreal power grid simulation.

- Assume exponential lifetimes, mean=20 years. (Matlab code)
- L=[];
- for i=1:10000
- L=[L lifepowr(exprnd(20,1,8))];
- end
- hist(L,50)

(or try L=[L lifepowr(unifrnd(0,40,1,8))];   )

Results:

min(L)

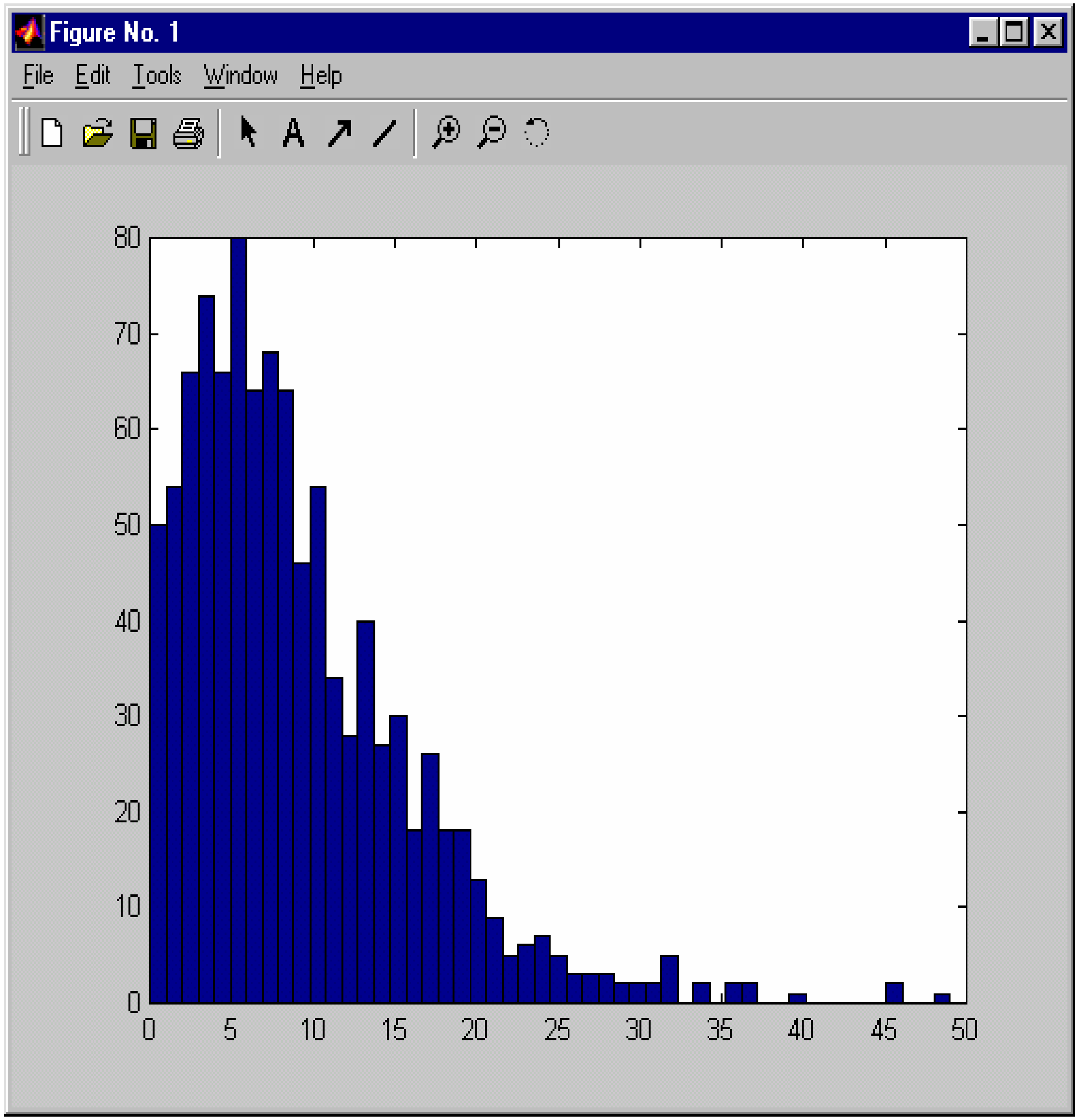
ans =

0.0185

» mean(L)

ans =

9.1255



Number of simulations for given accuracy.

95% CI for parameter is $mean \pm 2S/\sqrt{n}$

where S^2 is sample variance obtained from pilot simulation .

To ensure estimator within δ,

(with confidence around 95%)

set $\delta = 2S/\sqrt{n}$

Solve for n. $n \geq \left(\dfrac{2S}{\delta} \right)^2$

95% confidence interval for mean system life

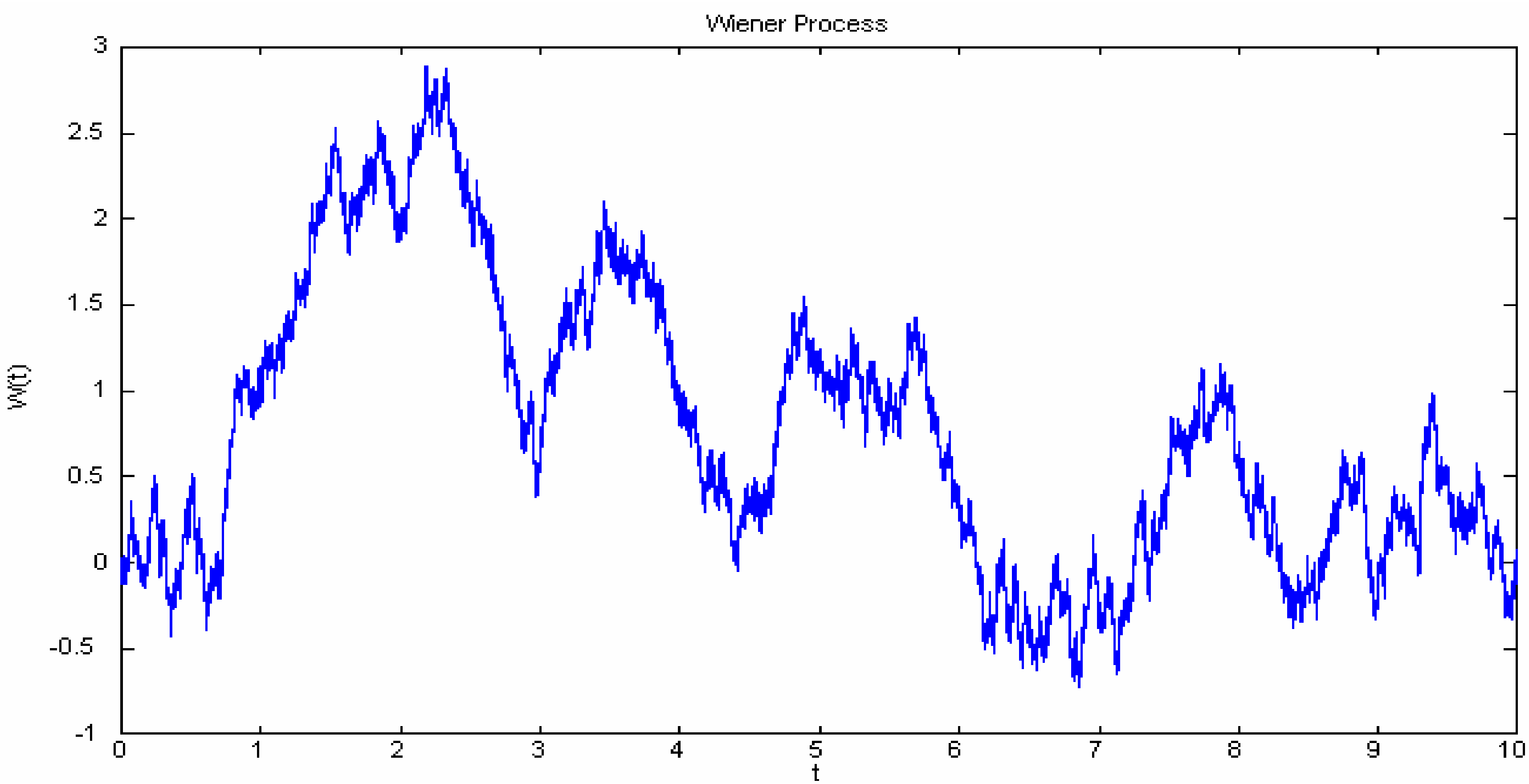
- M=mean(L)
- S=sqrt(var(L))
- M+[-1 1]*2*S/sqrt(length(L))
- to achieve accuracy to 2 decimals, need delta=.01
- sample size=(2*S/.01)^2
- TOO LARGE-- we need VARIANCE REDUCTION (better simulations)

The Brownian Motion (Wiener) Process

and diffusion. First, we define a stochastic process W_t called the *standard Brownian motion* or *Wiener process* having the following properties;

1. *For each* $h > 0$, *the increment* $W(t+h) - W(t)$ *has a* $N(0, h)$ *distribution and is independent of all preceding increments* $W(u) - W(v), t > u > v > 0.$
2. $W(0) = 0.$

Brownian Motion Path



Stochastic integral and approximating sums

$$\int_0^T h(t)dW_t \approx \sum_{i=0}^{n-1} h(t_i)(W(t_{i+1}) - W(t_i)), 0 = t_0 < t_1 < \ldots < t_n = T.$$

- Denote by *dW* a small increment $W(t_{i+1}) - W(t_i)$

- Then *dW* distributed *N(0,dt)*.

- *dW* has standard deviation $(dt)^{1/2}$

Taylor’s expansion

•Define a new process
(e.g. derivative price)

$$V_t = g(W_t, t)$$

•By Taylor’s Theorem

$$dV_t = \frac{\partial}{\partial W} g(W_t, t) dW + \frac{\partial^2}{\partial W^2} g(W_t, t) \frac{dW^2}{2} + \frac{\partial}{\partial t} g(W_t, t) dt + (\text{stuff}) \times (dW)^3 +$$

…..

Meaning of differentials

$$dX_t = h(t)dW_t$$

then this only has real meaning through its integrated version

$$X_t = X_0 + \int_0^t h(t)dW_t.$$

Ito’s lemma

What about the terms involving $(dW)^2$? What meaning should we assign to a term like $\int h(t)(dW)^2$? Consider the approximating function $\sum h(t_i)(W(t_{i+1}) - W(t_i))^2$. Notice that, at least in the case that the function h is non-random we are adding up independent random variables $h(t_i)(W(t_{i+1}) - W(t_i))^2$ each with expected value $h(t_i)(t_{i+1} - t_i)$ and when we add up these quantities the limit is $\int h(t)dt$ by the law of large numbers. Roughly speaking, as differentials, we should interpret $(dW)^2$ as dt because that is the way it acts in an integral. Subsequent terms such as $(dW)^3$ or $(dt)(dW)^2$ are all $o(dt)$, i.e. they all approach 0 faster than does dt as $dt \to 0$. So finally substituting for $(dW)^2$ in ref: ito2 and ignoring all terms that are $o(dt)$, we obtain a simple version of *Ito's lemma*

$$dg(W_t, t) = \frac{\partial}{\partial W} g(W_t, t)dW + \{\frac{1}{2}\frac{\partial^2}{\partial W^2} g(W_t, t) + \frac{\partial}{\partial t} g(W_t, t)\}dt.$$

Example of Ito's lemma

This rule results, for example, when we put $g(W_t, t) = W_t^2$ in

$$d(W_t^2) = 2W_t dW_t + dt$$

or on integrating both sides and rearranging,

$$\int_a^b W_t dW_t = \frac{1}{2}(W_b^2 - W_a^2) - \frac{1}{2}\int_a^b dt.$$

Martingale property of stochastic integrals

There is one more property of the stochastic integral that makes it a valuable tool in the construction of models in finance, and that is that a stochastic integral with respect to a Brownian motion process is *always a martingale*. To see this, note that in an approximating sum

$$\int_0^T h(t)dW_t \approx \sum_{i=0}^{n-1} h(t_i)(W(t_{i+1}) - W(t_i))$$

each of the summands has conditional expectation 0 given the past, i.e.

$$E[h(t_i)(W(t_{i+1}) - W(t_i))|H_{t_i}] = h(t_i)E[(W(t_{i+1}) - W(t_i))|H_{t_i}] = 0$$

since the Brownian increments have mean 0 given the past and since $h(t)$ is measurable with respect to H_t.

The Black-Scholes model for Stock Prices (discrete time)

- Assume that the price of a stock on day m is

$$S_t = S_0 \exp\{\sum_{j=1}^{m} Z_j\}$$

where S_0 is the stock price at time 0 and the random

variables Z_i are independent normal $N(\mu, \sigma^2 / N)$

where $\mu = \frac{r}{N} - \frac{\sigma^2}{2N}$,

r is the annual interest rate (e.g. 0.05)

and N is the number of (trading) days in a year (e.g. 252).

The model (for pricing financial derivatives)

- Notice that under this model, S_m has a lognormal distribution
- with expected value $S_0 e^{rm/N}$
- annual volatility= σ
- Expected value of future stock price is the same as that of bank deposit of equal amount. (this assumption is forced on us by no-arbitrage conditions whenever we are pricing a financial derivative)

Financial derivatives

- Financial instruments that derive their value from an associated asset (e.g. stock, index)
- Used to speculate on a rise (call option) or fall (put option) in asset price.
- used hedge a portfolio already held-e.g. a promise to deliver IBM stock at point in future. Insurance against disadvantageous moves in asset prices, currency exchange rates, interest rates, credit changes, etc.
- Financial equivalent of insurance company: they allow for transfer of RISK

What is a Call Option?

- A call option is a right, but not an obligation, to purchase an underlying stock for a fixed price K (exercise price) at a fixed time T (maturity).
  Payoff at time T years

= $\max(S_T - K, 0)$

- Where stock price at maturity = S_T
- If interest rates are constant r compounded continuously, payoff discounted to present (t=0) is

$$e^{-rT} \max(S_T - K, 0)$$

Call Option Price is..

- the expected present value (i.e. discounted to present) of future returns is

$$E\{e^{-rT}\max(S_T - K, 0)\}$$

where expectation is under the Black-Scholes model above.

- Example: A stock price worth $100. I have a call option with strike K=$100 maturing in 53 business days. If stock sells then for $120, the present value of the payoff assuming 5% interest rate and N=252

$$= e^{-.05(53/252)}\max(120 - 100, 0)$$

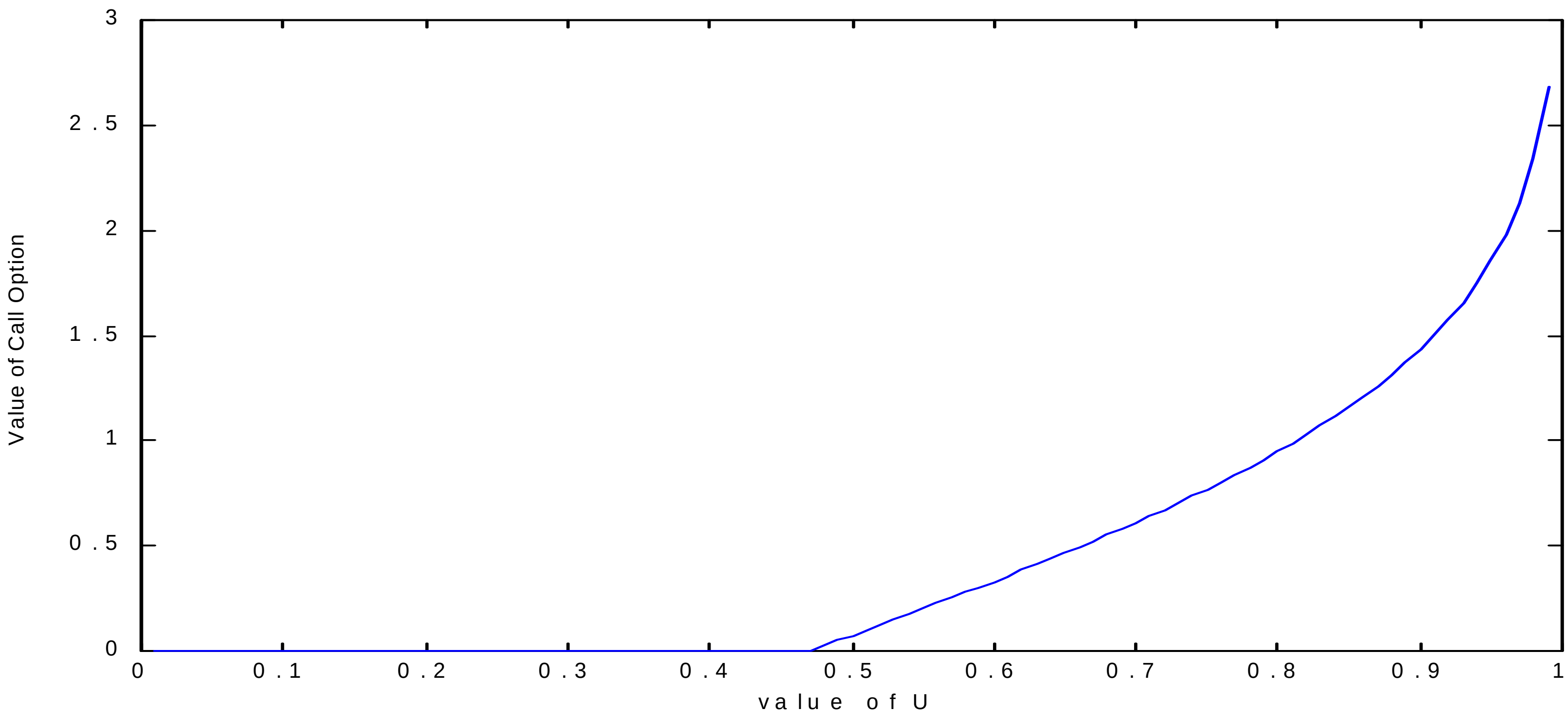
Simulated value of a call option

for a European call option with value function at maturity

$$V_o(x) = \max(x - K, 0), \text{ where } K = \text{exercise price and}$$

$$K = S_0 = 10, r = 0.05, \sigma = 0.2, T = 0.25$$

(area under graph below)



Crude simulation for European call option

Find by simulation the value of a European call option

$= $ expected payoff from option discounted to present

$= \mathrm{E}\{\mathrm{e}^{\text{-rT}} \max(S_0 e^X - K, 0)\}$ where X is a

Normal random variable with mean $rT - \sigma^2 T / 2$ and

variance $\sigma^2 T$ and S_0 is the current stock price.

Option value= Expectation with respect to uniform

Suppose X has cumulative distribution function F so

$P[X \le x] = F(x)$.

We may generate X using inverse transform

$X = F^{-1}(U)$ where U is uniform on [0,1].

Then the option price

$$\mathrm{E}\{\mathrm{e}^{\text{-rT}} \max(S_0 e^{F^{-1}(U)} - K, 0)\}$$

$$= \int_0^1 f(u)du$$

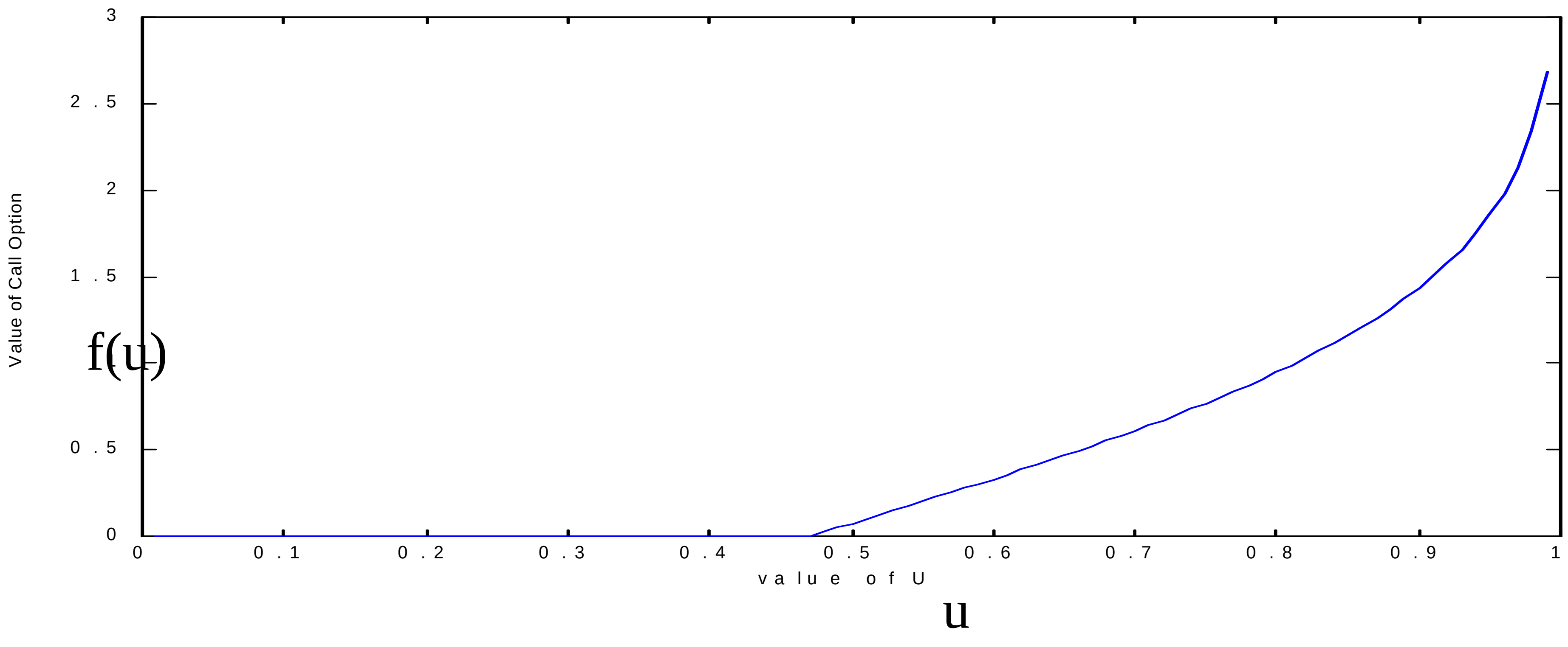
where $f(u) = \mathrm{e}^{\text{-rT}} \max(S_0 e^{F^{-1}(u)} - K, 0)$

Call option value is area under this graph

For European call option,

$$f(u) = e^{-rT}\ \max(S_0 e^{F^{-1}(u)} - K, 0)\ \text{ where } K = \text{exercise price.}$$

For graph, I chose $K = S_0 = 10, r = 0.05, \sigma = 0.2, T = 0.25$



Matlab Function for simulated value call option

- ```
  function v= fn(u)
  ```
- ```
  % discounted payoff for  call option with
  %exercise price K, r=annual interest rate,
  %sigma=annual vol, S0=current stock price,
  %u=vector of uniform (0,1) input to generate
  %normal variate by inverse transform. %T=maturity
  (years)
  ```
- ```
  %(For Black-Scholes price, integrate over(0,1)).
  ```
- ```
  S0=10 ;K=10;r=.05; sigma=.2 ;T=.25 ;
  ```
- ```
  ST=S0*exp(norminv(u,T*(r-sigma^2/2),sigma*sqrt(T)));
  ```
- ```
  % ST is the stock price at maturity. Discount
  ```
- ```
  v=exp(-r*T)*max((ST-K),0);
  ```
  ```

Black-Scholes price is integral over (0,1)

- We assumed
  - current stock price=exercise price=10
  - Annual vol (sigma)=0.2.
  - Interest rate =5%.
  - T=1/4 years.
- Monte-Carlo integral obtained by
  - u=rand(1,500000); mean(fn(u));

Accuracy of Crude Monte-Carlo option price:

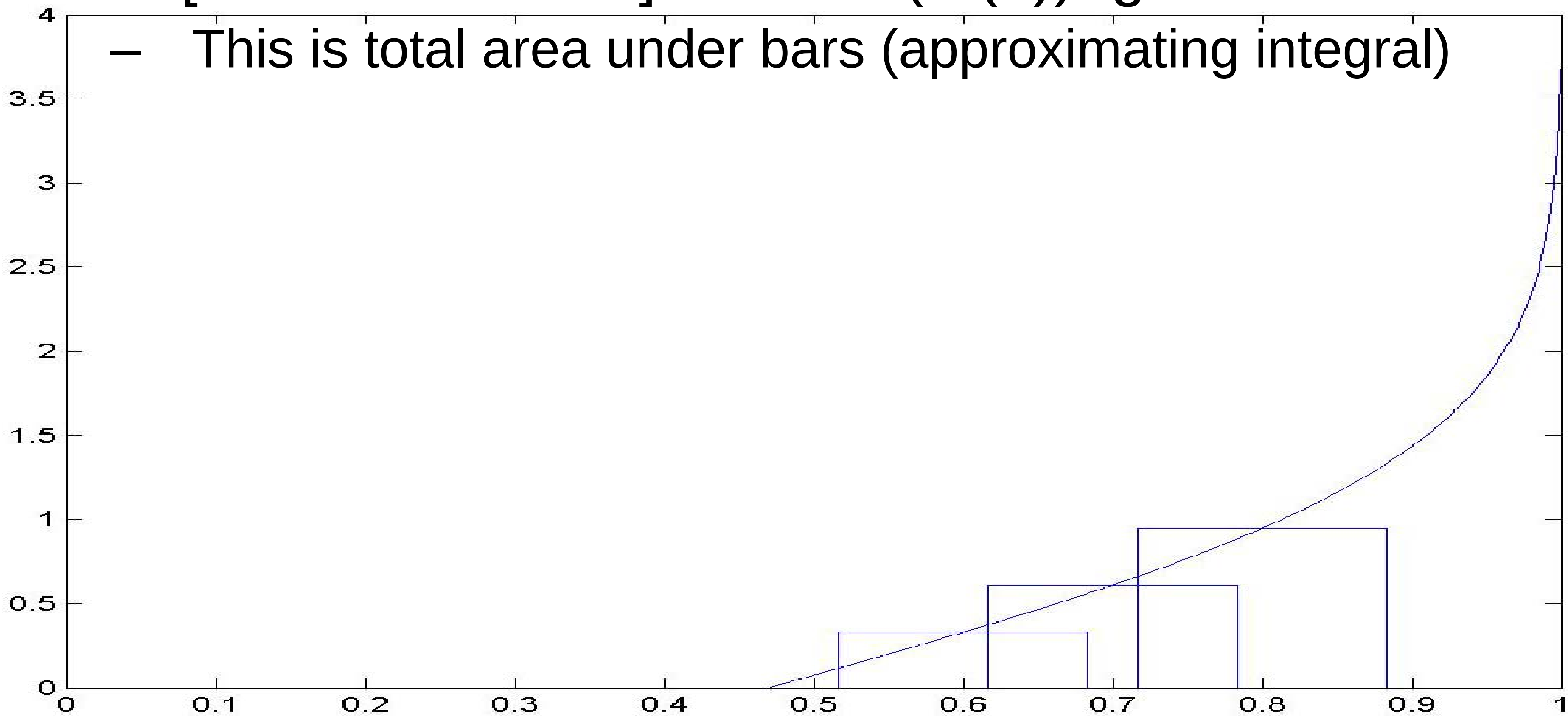
- Recall that $$Var(\frac{1}{n}\sum_{i=1}^{n} f(U_i)) = \frac{1}{n}Var(f(U_1))$$
- The standard error or standard deviation of the estimated Monte-Carlo integral is estimated by
- se=sqrt(var(fn(u))/length(u));
- True val=.4615, standard error approx .002

The true value by Black-Scholes formula (Matlab)

- [CALL,PUT]=BLSPRICE(S0,K,r,T,sigma,0) last argument = dividend rate assumed 0 in this problem. CALL= price of call option, PUT=price of put option both assuming Black-Scholes model.
- [CALL,PUT]=BLSPRICE(10,10,.05,.25,.2,0)

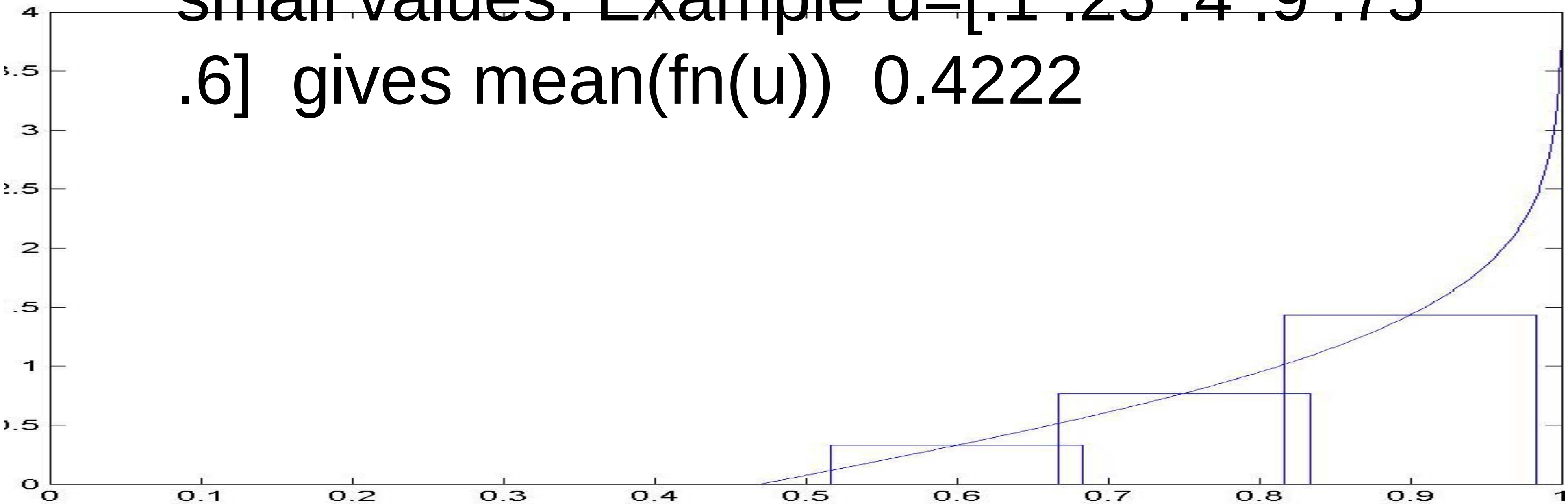
Example of crude Monte Carlo

- u=[.1 .25 .4 .6 .7 .8]   mean(fn(u)) gives   0.3146
  - This is total area under bars (approximating integral)



Antithetic Random Numbers

- **If we use a given uniform random number u, also use 1-u.** Achieves a balance between large values of f(u) and small values. Example u=[.1 .25 .4 .9 .75 .6] gives mean(fn(u)) 0.4222



Why are antithetic random numbers better? Consider independent uniform

Consider just two uniform random numbers

U_1, U_2 both uniform on [0,1] and independen t.

Estimator is $\frac{1}{2}(f(U_1) + f(U_2))$.

Expected value is $E[\frac{1}{2}(f(U_1) + f(U_2))] = \frac{1}{2}(Ef(U_1) + Ef(U_2)) = \int_0^1 f(u)du$.

Estimator has Variance

$$\text{var}[\frac{1}{2}(f(U_1) + f(U_2))] = \frac{1}{4}(\text{var}(f(U_1)) + \text{var}(f(U_2)) + 2\,\text{cov}(f(U_1), f(U_2)))$$

$$= \frac{1}{2}(\text{var}(f(U_1)) \text{ since } \text{var}(f(U_1)) = \text{var}(f(U_2))$$

and by independen ce, $\text{cov}(f(U_1), f(U_2)) = 0$

Expected value when they are antithetic

- The expected value is the same when antithetic:

U_1, U_2 both uniform on [0,1] and $U_1 = 1 - U_2$

Estimator is $\frac{1}{2}(f(U_1) + f(1-U_1))$.

Expected value is

$$E[\frac{1}{2}(f(U_1) + f(1-U_1))] = \frac{1}{2}(Ef(U_1) + Ef(1-U_1))$$

$$= \int_0^1 f(u)du.$$

Variance of Antithetic estimator (n=2)

Estimator has Variance

$$\mathrm{var}[\frac{1}{2}(f(U_1)+f(1-U_1))]=\frac{1}{4}(\mathrm{var}(f(U_1))+\mathrm{var}(f(1-U_1))+2\,\mathrm{cov}(f(U_1),f(1-U_1)))$$

$$=\frac{1}{2}(\mathrm{var}(f(U_1)+\mathrm{cov}(f(U_1),f(1-U_1)))\ \text{ since } \mathrm{var}(f(U_1))=\mathrm{var}(f(U_2))$$

This is better than crude if $\mathrm{cov}(f(U_1),f(1-U_1))<0.$

In other words when $f(U_1)$ is large, $f(1-U_1)$ tends to be smaller.

TRUE if f(u) is monotone function of u.

The antithetic estimator has smaller variance provided f is monotone

Summary of antithetic

- Instead of using n uniform[0,1], use n/2 (u, say) together with n/2 values of 1-u. This is better than n crude IF the function f(u) is monotonic (increasing or decreasing).
  - How much better?? For crude, u=rand(1,500000); mean(fn(u)) (0.4620)
  - var(fn(u))/500000 = 8.7e-007
- ANTITHETIC*********************
  - u=rand(1,500000); e=(fn(u)+fn(1-u))/2; mean(e) = 0.4630
  - var(e)/length(e) (2.23e-7—equivalent to about 2 million crude)

Efficiency Gain

- The same number of calls to the function fn using antithetic gives ratio of variances 2.1723/1.1244 relative to crude. Nearly twice as efficient (half the sample size necessary)
- In this example, if we insist on a certain precision (variance of estimator), antithetic method will get it with half as many calls to the function. (script8)

Antithetic: efficiency higher if function more linear

Find $\int_0^1 (1-x^2)^{3/4}\,dx.$

The true value is approx 0.71888.

- Using crude
  - u=rand(1,100000); f= (1-u.^2).^(3/4);
  - Mean(f)
  - Var(f)/100000

%mean(f) Var(f)/100000= 0.7193 7.1783e-007

- Using Antithetic
  - u=rand(1,50000); f= .5*((1-u.^2).^(3/4)+ (1-(1-u).^2).^(3/4));
  - mean(f)
  - Var(f)/50000

%mean(f) Var(f)/50000 = 0.7184 1.3704e-007

- Efficiency gain=7.18/1.37= 5.24