## Given CDF with two cases, generate X

$F(x) = \{ (1/2)e^{3x} \text{ if } x \le 0;\ 1 - (1/2)e^{-5x} \text{ if } x > 0 \}$

**For X ≤ 0 (U ≤ 1/2):**

Set $F(X) = (1/2)e^{3X} = U$
Solve: $X = (1/3)\ln(2U)$

**For X > 0 (U > 1/2):**

Set $F(X) = 1 - (1/2)e^{-5X} = U$
Solve: $X = -(1/5)\ln(2(1-U))$

**Final Answer:**

$X = \{ (1/3)\ln(2U) \text{ if } U \le 1/2;\ -(1/5)\ln(2(1-U)) \text{ if } U > 1/2 \}$

## Arena Templates

Basic Process

| Module | Sheet |
|---|---|
| Create | Attribute |
| Dispose | Entity |
| Process | Queue |
| Batch | Resource |
| Separate | Variable |
| Assign | Schedule |
| Record | Set |

Advanced Process

| Module | Sheet |
|---|---|
| Seize | Adv. Set |
| Delay | Expression |
| Release | Failure |

Blocks

| Module | Sheet |
|---|---|
| Seize | |
| Delay | |
| Release | |
| Queue | |

Advanced Transfer

| Module | Sheet |
|---|---|
| Station | Sequence |
| Route | Conveyor |
| Enter | Transporter |
| Leave | Distance |
| | Segment |

Transporter uses Request/Free, requires Distance Set.
Conveyor uses Access/Exit, requires Segment Set.

## Arena Variables and Function

| | |
|---|---|
| DISC(0.3, 1, 0.8, 2, 1.0, 3) | DISC generates random discrete values based on cumulative probabilities; pair each probability with its corresponding value. |
| TNOW | Current simulated time |
| NR(Res) | Res Servers currently in service |
| NQ(Queue) | Number of customers in Queue |
| Mod.NumberOut | Customers who have left the mode |

## Arena Set Types

Resource, Counter, entity type, entity picture

## Arena Key Modules

| | |
|---|---|
| Assign | Give new value to an attribute |
| Decide | Route customers probabilistically or conditonally |
| Separate / Clone | Split one customer into two or more clones |
| Route | Move entities station to station (advanced transfer) |
| Enter / Leave | Usually paired together for station management |
| Seize - Delay - Release | equals single Process module |
| Queue block | Cannot connect with a process module |

## Example of finding X for Pois

| $x$ | $f(x) = e^{-\lambda}\lambda^x/x!$ | $F(x)$ | Unif(0,1)'s |
|---|---|---|---|
| 0 | 0.8187 | 0.8187 | $[0, 0.8187]$ |
| 1 | 0.1637 | 0.9824 | $(0.8187, 0.9824]$ |
| 2 | 0.0164 | 0.9988 | $(0.9824, 0.9988]$ |
| 3 | 0.0011 | 0.9999 | $(0.9988, 0.9999]$ |
| $\geq 4$ | 0.0001 | 1.0000 | $(0.9999, 1.0000]$ |

Notice that I'm not very concerned with the "$\geq 4$" case. Anyway, suppose that I draw a PRN $U = 0.726$. What value of $X$ do I get?

(a) 0
(b) 0.726
(c) 1
(d) 2
(e) 3

**Solution:** From the table, we see that $U = 0.726$ clearly corresponds to $X = 0$, choice (a). □

## Universal truths

| | |
|---|---|
| −ln(U) | ~Exponential(1) |
| | lnu |

## messy cdf

| | |
|---|---|
| See F(X) | Replace with Uniform(0,1) |
| Multiply by 3 | U(0,3) |
| Subtract 1 | U(−1,2) |
| Mean | (-1+2)/2 = 1/2 |

What is the mean of the random variable 3F(X)−1?

## Inverse Transformation

| | |
|---|---|
| Given U, find Z | invNorm(U, 0, 1) |
| Given Z, find U | normCdf(-9999, Z, 0, 1) |
| Exponential(λ) | X = -ln(1-U)/λ |
| Uniform(a,b) | X = a + (b-a)·U |
| Weibull(a,b) | X = a * $(-\ln(U))^{1/b}$ |
| Triangular | If U<0.5: √(2U); If U≥0.5: 2-√(2(1-U)) |
| Bernoulli(p) | If U < 1-p → 0; Else → 1 |
| Poisson(λ) | Build CDF, match U |
| Discrete Unif(1,n) | [n·U] |
| Erlang(k,λ) | -(1/λ)ln(∏Ui) |
| Geometric | ln(1-U)/ln(1-p) |

For discrete: Find smallest x where F(x) ≥ U
For continuous: Use inverse CDF formulas
Box-Muller generates TWO Normal(0,1) values from TWO Uniform(0,1) values

## XOR

| | |
|---|---|
| XOR | is only true if different |

## Expected Value, Variance, ...

| | |
|---|---|
| Discrete E[X] | SUM[x * f(x)] |
| Continous E[X] | SUM[x * f(x) dx] |
| Variance of X | $E[X^2] - (E[X])^2$ |
| Standard Deviation of X | SQRT[Var(X)] |

## Expected Value Joint pdf

Joint p.d.f.: $f(x,y) = 2xy^2$; Domain: $0 < x < 1$, $0 < y < 1$

Find: E[2X-1]

$E[2X - 1] = \int_0^1 \int_0^1 (2x - 1) \cdot 2xy^2 \, dy \, dx$

## Random Number Generators

| | |
|---|---|
| Bad generators | Midsquare number generator, Random number tables, von Neumann's mid-square method, Fibonacci generator, Additive congruential generator, RANDU |
| Good generators | Linear Congruential Generators (modern cycle length > $2^{191}$; Mersenne Twister ($2^{19937}$) |
| Randu | 65539Xi mod($2^{31}$) |
| Desert island | 16807Xi mod($2^{31}$-1) mod(2147483647) |
| Desert island technique | Z = [SUM(Ui)-n/2] / [SQRT(n * 1/12] |

## Inverse Transform Method Key Problems

| | |
|---|---|
| If X ~ Normal(0,1), what's the distribution of Φ(X)? | Unif(0,1) |
| If U~Uniform(0,1) and Φ(x) is the CDF of the standard normal, what is the distribution of $2\Phi^{-1}(U)+3$? | $\Phi^{-1}(U)$ turns a Uniform(0,1) into a Normal(0,1). Multiply by 2 → scales the standard deviation by 2 = Normal(0,4). Add 3 → shifts the mean to 3. = Normal(3,4) |
| $-3\ell n(U^2V^2)$ where U, V ~ i.i.d. Unif(0,1) | $= -6\ell n(U) - 6\ell n(V)$ ~ Exp(1/6) + Exp(1/6) ~ $Erlang_2(1/6)$ |

## Joint Probability Mass Function

| | |
|---|---|
| E[XY] | Summe von x * y * f(xy) |

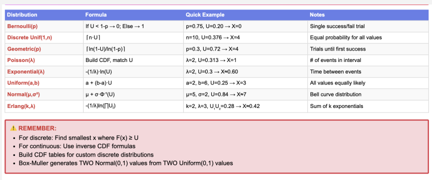
## Joint p.m.f.

| | |
|---|---|
| Are independent if | X and Y are independent if and only if pX,Y(x,y) = pX(x) · pY(y) |
| For example | P(X=1, Y=0) = P(Y=0) * P(X=1) |
| E[XY] | Example: (1)(0)(0.2) + (1)(1)(0.0.0) + ... |
| Cov(X,Y) | E[XY] - E[Y]E[X] |
| Variance X + Y | Var(X) + Var(Y) + 2Cov(X,Y) |
| Variance X - Y | Var(X) + Var(Y) - 2Cov(X,Y) |
| Theorem | Cov(X,Y) = 0 if X, Y indepedent. Converse not true. |
| Correlation | p = Cov(X,Y) / SQRT(Var(X) * Var(Y)) |

## pdf, cdf

| | |
|---|---|
| pdf -> cdf | integrate with x, 0 as limit |
| cdf F-1(U) | solve(F(x)=U, x) |

## Complete Distribution Reference Table



| Distribution | Formula | Quick Example | Notes |
|---|---|---|---|
| Bernoulli(p) | If U < 1-p → 0; Else → 1 | p=0.75, U=0.20 → X=0 | Single success/fail trial |
| Discrete Unif(1,n) | ⌈n·U⌉ | n=10, U=0.376 → X=4 | Equal probability for all values |
| Geometric(p) | ⌈ln(1-U)/ln(1-p)⌉ | p=0.3, U=0.72 → X=4 | Trials until first success |
| Poisson(λ) | Build CDF, match U | λ=2, U=0.313 → X=1 | # of events in interval |
| Exponential(λ) | -(1/λ)·ln(U) | λ=2, U=0.3 → X≈0.60 | Time between events |
| Uniform(a,b) | a + (b-a)·U | a=2, b=6, U=0.25 → X=3 | All values equally likely |
| Normal(μ,σ²) | μ + σ·Φ⁻¹(U) | μ=5, σ=2, U=0.84 → X≈7 | Bell curve distribution |
| Erlang(k,λ) | $-(1/\lambda)\ln(\prod U_i)$ | k=2, λ=3, $U_1U_2$=0.28 → X≈0.42 | Sum of k exponentials |

⚠ **REMEMBER:**

- For discrete: Find smallest x where F(x) ≥ U
- For continuous: Use inverse CDF formulas
- Build CDF tables for custom discrete distributions
- Box-Muller generates TWO Normal(0,1) values from TWO Uniform(0,1) values

## Box Muller Method

| | |
|---|---|
| Z1 | $\sqrt{(-2\cdot\ln(U_1))} \cdot \cos(2\pi\cdot U_2)$ |
| Z2 | $\sqrt{(-2\cdot\ln(U_1))} \cdot \sin(2\pi\cdot U_2)$ |
| Z1/Z2 | $\cot(2\pi\cdot U_2)$ ~Cauchy |
| Z2/Z1 | $\tan(2\pi\cdot U_2)$ ~Cauchy |

Radian-Modus einschalten!

## Chi-Square Distribution

If $Z_1$, $Z_2$, $Z_3$ are i.i.d. Nor(0,1), find c such that $P(Z_1^2 + Z_2^2 + Z_3^2 < c) = 0.99$

Calculator: chiSqInv(0.99, 3) → 11.34

$X_1$, $X_2$, $X_3$, $X_4$ are i.i.d. Nor(0,1). Find $Pr(X_1 + X_2 - X_3 - X_4 > 10)$

Calculator normCdf(10,unendlich,0,n/2)

## Find distribution of U1 and U2

| |
|---|
| Find distribution of $-4(U_1 + U_2) - 2$ |
| $U_1 + U_2 \sim$ Tria(0, 1, 2) |
| Apply the transformation $-4(U_1 + U_2) = 4$(Tria(0, 1, 2) ) |
| The minimum becomes: -4(2) = -8; The mode becomes: -4(1) = -4; The maximum becomes: -4(0) = 0 |
| 4·Tria(0, 1, 2) = Tria(-8, -4, 0) |
| Subtract 2 |
| Tria(-10, -6, -2) |

## Acceptance-Rejection

| | |
|---|---|
| Goal | Generate random samples from a hard-to-sample distribution (f(x)). |
| Idea | Sample from an easier distribution (h(x)), then accept or reject each sample based on how well it fits f(x). |
| Example | If a random variable X has the beta distribution, then its p.d.f. is of the form f(x) = Γ(α+β) Γ(α)Γ(β) xα−1(1−x)β−1, 0 <x<1, for parameters α and β >0, and where Γ(·) is the gamma function. How might you generate such a random variate? Pick the best answer. |

## Goodness of fit

| | |
|---|---|
| Find critical value | Inverse X2; Area = 1-alpha; df = n-1; invx2(0.9,3) |
| | If critical value is bigger than accept H0 |

## Hypothesis Testing Errors

| | |
|---|---|
| Type I Error (False positive) | $\alpha = P(\text{Reject } H_0 \mid H_0 \text{ is true})$ , Fire alarm when no fire |
| Type II Error (False negative) | $\beta = P(\text{Fail to reject } H_0 \mid H_0 \text{ is false})$; No fire alarm when there IS a fire |