Symbolic Algebra Applications in Quality and Productivity

Andy Glen
United States Military Academy
Larry Leemis
William & Mary

June 4, 2009
(joint work with John Drew, Matt Duggan, Diane Evans, Jeff Mallozzi, Bruce Schmeiser, Jeff Yang, Billy Kaczynski)

Outline

Introduction

Motivation for APPL (A Probability Programming Language):

- Flexibility over statistical packages
- Research
  - expand the classes of problems addressed analytically
  - mathematically intractable problems in probability
  - creating new probability distributions
- Probability education

APPL is available at no charge to non-commercial users at

`www.APPLsoftware.com`

What is APPL?

APPL (A Probability Programming Language) gives exact solutions to probability problems. Sample functions:

- `Transform` to find the transformation of a random variable;
- `Convolution` to find the sum of independent random variables;
- `Mean` to find the expected value of a random variable;
- `OrderStat` to find the distribution of an order statistic;
- `Minimum` to find the distribution of a minimum;
- `PDF`, `CDF`, `IDF`, `HF` to find the PDF, CDF, IDF, HF;
- `ExponentialRV`, and other popular distributions.

What is APPL?

APPL (A Probability Programming Language) gives exact solutions to probability problems. Sample functions:

- `Transform` to find the transformation of a random variable;
- `Convolution` to find the sum of independent random variables;
- `Mean` to find the expected value of a random variable;
- `OrderStat` to find the distribution of an order statistic;
- `Minimum` to find the distribution of a minimum;
- `PDF`, `CDF`, `IDF`, `HF` to find the PDF, CDF, IDF, HF;
- `ExponentialRV`, and other popular distributions.

What is APPL?

APPL (A Probability Programming Language) gives exact solutions to probability problems. Sample functions:

- `Transform` to find the transformation of a random variable;
- `Convolution` to find the sum of independent random variables;
- `Mean` to find the expected value of a random variable;
- `OrderStat` to find the distribution of an order statistic;
- `Minimum` to find the distribution of a minimum;
- `PDF`, `CDF`, `IDF`, `HF` to find the PDF, CDF, IDF, HF;
- `ExponentialRV`, and other popular distributions.

What is APPL?

APPL (A Probability Programming Language) gives exact solutions to probability problems. Sample functions:

- `Transform` to find the transformation of a random variable;
- `Convolution` to find the sum of independent random variables;
- `Mean` to find the expected value of a random variable;
- `OrderStat` to find the distribution of an order statistic;
- `Minimum` to find the distribution of a minimum;
- `PDF`, `CDF`, `IDF`, `HF` to find the PDF, CDF, IDF, HF;
- `ExponentialRV`, and other popular distributions.

What is APPL?

APPL (A Probability Programming Language) gives exact solutions to probability problems. Sample functions:

- `Transform` to find the transformation of a random variable;
- `Convolution` to find the sum of independent random variables;
- `Mean` to find the expected value of a random variable;
- `OrderStat` to find the distribution of an order statistic;
- `Minimum` to find the distribution of a minimum;
- `PDF`, `CDF`, `IDF`, `HF` to find the PDF, CDF, IDF, HF;
- `ExponentialRV`, and other popular distributions.

What is APPL?

APPL (A Probability Programming Language) gives exact solutions to probability problems. Sample functions:

- `Transform` to find the transformation of a random variable;
- `Convolution` to find the sum of independent random variables;
- `Mean` to find the expected value of a random variable;
- `OrderStat` to find the distribution of an order statistic;
- `Minimum` to find the distribution of a minimum;
- `PDF`, `CDF`, `IDF`, `HF` to find the PDF, CDF, IDF, HF;
- `ExponentialRV`, and other popular distributions.

What is APPL?

Andy Glen
United States Military Academy
Larry Leemis
William & Mary

APPL (A Probability Programming Language) gives exact solutions to probability problems. Sample functions:

- `Transform` to find the transformation of a random variable;
- `Convolution` to find the sum of independent random variables;
- `Mean` to find the expected value of a random variable;
- `OrderStat` to find the distribution of an order statistic;
- `Minimum` to find the distribution of a minimum;
- `PDF`, `CDF`, `IDF`, `HF` to find the PDF, CDF, IDF, HF;
- `ExponentialRV`, and other popular distributions.

What is APPL?

APPL (A Probability Programming Language) gives exact solutions to probability problems. Sample functions:

- `Transform` to find the transformation of a random variable;
- `Convolution` to find the sum of independent random variables;
- `Mean` to find the expected value of a random variable;
- `OrderStat` to find the distribution of an order statistic;
- `Minimum` to find the distribution of a minimum;
- `PDF`, `CDF`, `IDF`, `HF` to find the PDF, CDF, IDF, HF;
- `ExponentialRV`, and other popular distributions.

Example 1: Sums of IID random variables

Let $X_1, X_2, \ldots, X_{10}$ be IID U(0,1) random variables. Find

$$\Pr\left(4 < \sum_{i=1}^{10} X_i < 6\right)$$

Typical approaches

- Central limit theorem
  - population distribution not normal
  - small sample size
  - only one digit of accuracy here
- Simulation
  - requires custom programming
  - solution is given as a point and interval estimator
  - each additional digit of accuracy requires 100× replications

Example 1: Sums of IID random variables

Let $X_1, X_2, \ldots, X_{10}$ be IID U(0,1) random variables. Find

$$\Pr\left(4 < \sum_{i=1}^{10} X_i < 6\right)$$

Typical approaches

- Central limit theorem
  - population distribution not normal
  - small sample size
  - only one digit of accuracy here
- Simulation
  - requires custom programming
  - solution is given as a point and interval estimator
  - each additional digit of accuracy requires 100× replications

Example 1: Sums of IID random variables

Let $X_1, X_2, \ldots, X_{10}$ be IID U(0,1) random variables. Find

$$\Pr\left(4 < \sum_{i=1}^{10} X_i < 6\right)$$

Typical approaches

- Central limit theorem
  - population distribution not normal
  - small sample size
  - only one digit of accuracy here
- Simulation
  - requires custom programming
  - solution is given as a point and interval estimator
  - each additional digit of accuracy requires 100× replications

Example 1: Sums of IID random variables

Let $X_1, X_2, \ldots, X_{10}$ be IID U(0,1) random variables. Find

$$\Pr\left(4 < \sum_{i=1}^{10} X_i < 6\right)$$

Typical approaches

- Central limit theorem
  - population distribution not normal
  - small sample size
  - only one digit of accuracy here
- Simulation
  - requires custom programming
  - solution is given as a point and interval estimator
  - each additional digit of accuracy requires 100× replications

Example 1: Sums of IID random variables

Let $X_1, X_2, \ldots, X_{10}$ be IID U(0,1) random variables. Find

$$\Pr\left(4 < \sum_{i=1}^{10} X_i < 6\right)$$

Typical approaches

- Central limit theorem
  - population distribution not normal
  - small sample size
  - only one digit of accuracy here
- Simulation
  - requires custom programming
  - solution is given as a point and interval estimator
  - each additional digit of accuracy requires $100\times$ replications

Andy Glen
United States Military Academy
Larry Leemis
William & Mary

Example 1: Sums of IID random variables

Andy Glen
United States Military Academy
Larry Leemis
William & Mary

Let $X_1, X_2, \ldots, X_{10}$ be IID U(0,1) random variables. Find

$$\Pr\left(4 < \sum_{i=1}^{10} X_i < 6\right)$$

Typical approaches

- Central limit theorem
  - population distribution not normal
  - small sample size
  - only one digit of accuracy here
- Simulation
  - requires custom programming
  - solution is given as a point and interval estimator
  - each additional digit of accuracy requires 100× replications

Example 1: Sums of IID random variables

Let $X_1, X_2, \ldots, X_{10}$ be IID U(0,1) random variables. Find

$$\Pr\left(4 < \sum_{i=1}^{10} X_i < 6\right)$$

Typical approaches

- Central limit theorem
  - population distribution not normal
  - small sample size
  - only one digit of accuracy here
- Simulation
  - requires custom programming
  - solution is given as a point and interval estimator
  - each additional digit of accuracy requires 100× replications

Example 1: Sums of IID random variables

Let $X_1, X_2, \ldots, X_{10}$ be IID U(0,1) random variables. Find

$$\Pr\left(4 < \sum_{i=1}^{10} X_i < 6\right)$$

Typical approaches

- Central limit theorem
  - population distribution not normal
  - small sample size
  - only one digit of accuracy here
- Simulation
  - requires custom programming
  - solution is given as a point and interval estimator
  - each additional digit of accuracy requires $100\times$ replications

Example 1: Sums of IID random variables

Let $X_1, X_2, \ldots, X_{10}$ be IID U(0,1) random variables. Find

$$\Pr\left(4 < \sum_{i=1}^{10} X_i < 6\right)$$

Typical approaches

- Central limit theorem
  - population distribution not normal
  - small sample size
  - only one digit of accuracy here
- Simulation
  - requires custom programming
  - solution is given as a point and interval estimator
  - each additional digit of accuracy requires $100\times$ replications

Example 1: Sums of IID random variables

Andy Glen
United States Military Academy
Larry Leemis
William & Mary

Let $X_1, X_2, \ldots, X_{10}$ be IID U(0,1) random variables. Find

$$\Pr\left(4 < \sum_{i=1}^{10} X_i < 6\right)$$

Typical approaches

- Central limit theorem
  - population distribution not normal
  - small sample size
  - only one digit of accuracy here
- Simulation
  - requires custom programming
  - solution is given as a point and interval estimator
  - each additional digit of accuracy requires $100\times$ replications

Example 1: Sums of IID random variables (continued)

Let $X_1, X_2, \ldots, X_{10}$ be IID U(0,1) random variables. Find

$$\Pr\left(4 < \sum_{i=1}^{10} X_i < 6\right)$$

APPL code:

```
> n := 10;
> X := UniformRV(0, 1);
> Y := ConvolutionIID(X, n);
> CDF(Y, 6) - CDF(Y, 4);
```

$$\frac{655177}{907200}$$

Example 1: Sums of IID random variables (continued)

Let $X_1, X_2, \ldots, X_{10}$ be IID U(0,1) random variables. Find

$$\Pr\left(4 < \sum_{i=1}^{10} X_i < 6\right)$$

APPL code:

```
> n := 10;
> X := UniformRV(0, 1);
> Y := ConvolutionIID(X, n);
> CDF(Y, 6) - CDF(Y, 4);
```

$$\frac{655177}{907200}$$

Example 1: Sums of IID random variables (continued)

Let $X_1, X_2, \ldots, X_{10}$ be IID U(0,1) random variables. Find

$$\Pr\left(4 < \sum_{i=1}^{10} X_i < 6\right)$$

APPL code:

```
> n := 10;
> X := UniformRV(0, 1);
> Y := ConvolutionIID(X, n);
> CDF(Y, 6) - CDF(Y, 4);
```

$$\frac{655177}{907200}$$

Example 1: Sums of IID random variables (continued)

Let $X_1, X_2, \ldots, X_{10}$ be IID U(0,1) random variables. Find

$$\Pr\left(4 < \sum_{i=1}^{10} X_i < 6\right)$$

APPL code:

```
> n := 10;
> X := UniformRV(0, 1);
> Y := ConvolutionIID(X, n);
> CDF(Y, 6) - CDF(Y, 4);
```

$$\frac{655177}{907200}$$

Example 1: Sums of IID random variables (continued)

Let $X_1, X_2, \ldots, X_{10}$ be IID U(0,1) random variables. Find

$$\Pr\left(4 < \sum_{i=1}^{10} X_i < 6\right)$$

APPL code:

```
> n := 10;
> X := UniformRV(0, 1);
> Y := ConvolutionIID(X, n);
> CDF(Y, 6) - CDF(Y, 4);
```

$$\frac{655177}{907200}$$

Example 1: Sums of IID random variables (continued)

Let $X_1, X_2, \ldots, X_{10}$ be IID U(0,1) random variables. Find

$$\Pr\left(4 < \sum_{i=1}^{10} X_i < 6\right)$$

APPL code:

```
> n := 10;
> X := UniformRV(0, 1);
> Y := ConvolutionIID(X, n);
> CDF(Y, 6) - CDF(Y, 4);
```

$$\frac{655177}{907200}$$

Example 2: Product of two independent random variables

Let X and Y be independent random variables:

$$X \sim Triangular(1, 2, 4)$$

$$Y \sim Triangular(1, 2, 3)$$

Find the distribution of $V = XY$.

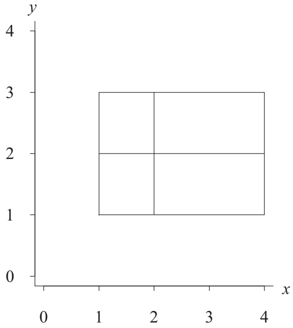
Example 2: Product of two independent random variables

Let X and Y be independent random variables:

$$X \sim Triangular(1, 2, 4)$$

$$Y \sim Triangular(1, 2, 3)$$

Find the distribution of $V = XY$.



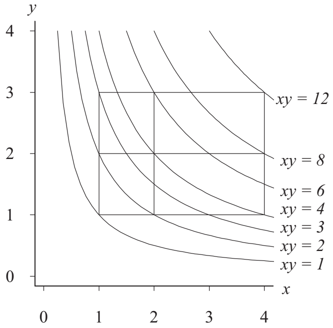
Example 2: Product of two independent random variables

Let X and Y be independent random variables:

$$X \sim Triangular(1, 2, 4)$$

$$Y \sim Triangular(1, 2, 3)$$

Find the distribution of $V = XY$.



Example 2: Product of two independent random variables (continued)

Andy Glen
United States Military Academy
Larry Leemis
William & Mary

The APPL code is

```
> X := TriangularRV(1, 2, 4);
> Y := TriangularRV(1, 2, 3);
> V := Product(X, Y);
```

$$f_V(v) = \begin{cases} -\frac{4}{3}v + \frac{2}{3}\ln v + \frac{2v}{3}\ln v + \frac{4}{3} & 1 < v \le 2 \\ -8 + \frac{14}{3}\ln 2 + \frac{7v}{3}\ln 2 + \frac{10}{3}v - 4\ln v - \frac{5v}{3}\ln v & 2 < v \le 3 \\ -4 + \frac{14}{3}\ln 2 + \frac{7v}{3}\ln 2 + 2v - 2\ln v - \\ \quad v\ln v - 2\ln 3 - \frac{2v}{3}\ln 3 & 3 < v \le 4 \\ \frac{44}{3} - 14\ln 2 - \frac{7v}{3}\ln 2 - \frac{8}{3}v - 2\ln 3 + \\ \quad \frac{22}{3}\ln v - \frac{2v}{3}\ln 3 + \frac{4v}{3}\ln v & 4 < v \le 6 \\ \frac{8}{3} - 8\ln 2 - \frac{4v}{3}\ln 2 - \frac{2}{3}v + \frac{4}{3}\ln v + \\ \quad \frac{v}{3}\ln v + 4\ln 3 + \frac{v}{3}\ln 3 & 6 < v \le 8 \\ -8 + 8\ln 2 + \frac{2v}{3}\ln 2 + \frac{2}{3}v + 4\ln 3 - \\ \quad 4\ln v + \frac{v}{3}\ln 3 - \frac{v}{3}\ln v & 8 < v < 12 \end{cases}$$

Example 2: Product of two independent random variables (continued)

The APPL code is

```
> X := TriangularRV(1, 2, 4);
> Y := TriangularRV(1, 2, 3);
> V := Product(X, Y);
```

$$f_V(v) = \begin{cases} -\frac{4}{3}v + \frac{2}{3}\ln v + \frac{2v}{3}\ln v + \frac{4}{3} & 1 < v \le 2 \\ -8 + \frac{14}{3}\ln 2 + \frac{7v}{3}\ln 2 + \frac{10}{3}v - 4\ln v - \frac{5v}{3}\ln v & 2 < v \le 3 \\ -4 + \frac{14}{3}\ln 2 + \frac{7v}{3}\ln 2 + 2v - 2\ln v - \\ \quad v\ln v - 2\ln 3 - \frac{2v}{3}\ln 3 & 3 < v \le 4 \\ \frac{44}{3} - 14\ln 2 - \frac{7v}{3}\ln 2 - \frac{8}{3}v - 2\ln 3 + \\ \quad \frac{22}{3}\ln v - \frac{2v}{3}\ln 3 + \frac{4v}{3}\ln v & 4 < v \le 6 \\ \frac{8}{3} - 8\ln 2 - \frac{4v}{3}\ln 2 - \frac{2}{3}v + \frac{4}{3}\ln v + \\ \quad \frac{v}{3}\ln v + 4\ln 3 + \frac{v}{3}\ln 3 & 6 < v \le 8 \\ -8 + 8\ln 2 + \frac{2v}{3}\ln 2 + \frac{2}{3}v + 4\ln 3 - \\ \quad 4\ln v + \frac{v}{3}\ln 3 - \frac{v}{3}\ln v & 8 < v < 12 \end{cases}$$

Example 2: Product of two independent random variables (continued)

The APPL code is

```
> X := TriangularRV(1, 2, 4);
> Y := TriangularRV(1, 2, 3);
> V := Product(X, Y);
```

$$f_V(v) = \begin{cases} -\frac{4}{3}v + \frac{2}{3}\ln v + \frac{2v}{3}\ln v + \frac{4}{3} & 1 < v \le 2 \\ -8 + \frac{14}{3}\ln 2 + \frac{7v}{3}\ln 2 + \frac{10}{3}v - 4\ln v - \frac{5v}{3}\ln v & 2 < v \le 3 \\ -4 + \frac{14}{3}\ln 2 + \frac{7v}{3}\ln 2 + 2v - 2\ln v - \\ \qquad v\ln v - 2\ln 3 - \frac{2v}{3}\ln 3 & 3 < v \le 4 \\ \frac{44}{3} - 14\ln 2 - \frac{7v}{3}\ln 2 - \frac{8}{3}v - 2\ln 3 + \\ \qquad \frac{22}{3}\ln v - \frac{2v}{3}\ln 3 + \frac{4v}{3}\ln v & 4 < v \le 6 \\ \frac{8}{3} - 8\ln 2 - \frac{4v}{3}\ln 2 - \frac{2}{3}v + \frac{4}{3}\ln v + \\ \qquad \frac{v}{3}\ln v + 4\ln 3 + \frac{v}{3}\ln 3 & 6 < v \le 8 \\ -8 + 8\ln 2 + \frac{2v}{3}\ln 2 + \frac{2}{3}v + 4\ln 3 - \\ \qquad 4\ln v + \frac{v}{3}\ln 3 - \frac{v}{3}\ln v & 8 < v < 12 \end{cases}$$

Example 2: Product of two independent random variables (continued)

The APPL code is

```
> X := TriangularRV(1, 2, 4);
> Y := TriangularRV(1, 2, 3);
> V := Product(X, Y);
```

$$f_V(v) = \begin{cases} -\frac{4}{3}v + \frac{2}{3}\ln v + \frac{2v}{3}\ln v + \frac{4}{3} & 1 < v \le 2 \\ -8 + \frac{14}{3}\ln 2 + \frac{7v}{3}\ln 2 + \frac{10}{3}v - 4\ln v - \frac{5v}{3}\ln v & 2 < v \le 3 \\ -4 + \frac{14}{3}\ln 2 + \frac{7v}{3}\ln 2 + 2v - 2\ln v - \\ \quad v\ln v - 2\ln 3 - \frac{2v}{3}\ln 3 & 3 < v \le 4 \\ \frac{44}{3} - 14\ln 2 - \frac{7v}{3}\ln 2 - \frac{8}{3}v - 2\ln 3 + \\ \quad \frac{22}{3}\ln v - \frac{2v}{3}\ln 3 + \frac{4v}{3}\ln v & 4 < v \le 6 \\ \frac{8}{3} - 8\ln 2 - \frac{4v}{3}\ln 2 - \frac{2}{3}v + \frac{4}{3}\ln v + \\ \quad \frac{v}{3}\ln v + 4\ln 3 + \frac{v}{3}\ln 3 & 6 < v \le 8 \\ -8 + 8\ln 2 + \frac{2v}{3}\ln 2 + \frac{2}{3}v + 4\ln 3 - \\ \quad 4\ln v + \frac{v}{3}\ln 3 - \frac{v}{3}\ln v & 8 < v < 12 \end{cases}$$

Example 2: Product of two independent random variables (continued)

The APPL code is

```
> X := TriangularRV(1, 2, 4);
> Y := TriangularRV(1, 2, 3);
> V := Product(X, Y);
```

$$f_V(v) = \begin{cases} -\frac{4}{3}v + \frac{2}{3}\ln v + \frac{2v}{3}\ln v + \frac{4}{3} & 1 < v \le 2 \\ -8 + \frac{14}{3}\ln 2 + \frac{7v}{3}\ln 2 + \frac{10}{3}v - 4\ln v - \frac{5v}{3}\ln v & 2 < v \le 3 \\ -4 + \frac{14}{3}\ln 2 + \frac{7v}{3}\ln 2 + 2v - 2\ln v - \\ \quad v\ln v - 2\ln 3 - \frac{2v}{3}\ln 3 & 3 < v \le 4 \\ \frac{44}{3} - 14\ln 2 - \frac{7v}{3}\ln 2 - \frac{8}{3}v - 2\ln 3 + \\ \quad \frac{22}{3}\ln v - \frac{2v}{3}\ln 3 + \frac{4v}{3}\ln v & 4 < v \le 6 \\ \frac{8}{3} - 8\ln 2 - \frac{4v}{3}\ln 2 - \frac{2}{3}v + \frac{4}{3}\ln v + \\ \quad \frac{v}{3}\ln v + 4\ln 3 + \frac{v}{3}\ln 3 & 6 < v \le 8 \\ -8 + 8\ln 2 + \frac{2v}{3}\ln 2 + \frac{2}{3}v + 4\ln 3 - \\ \quad 4\ln v + \frac{v}{3}\ln 3 - \frac{v}{3}\ln v & 8 < v < 12 \end{cases}$$

Example 3: Order statistics

A bag contains 15 billiard balls, numbered 1 to 15. If 7 balls are drawn from the bag at random, find the probability that the median number drawn is 5 when (*a*) sampling is performed without replacement; (*b*) sampling is performed with replacement.

(*a*) Sampling without replacement

```
> X := UniformDiscreteRV(1, 15);
> Y := OrderStat(X, 7, 4, "wo");
> PDF(Y, 5);
```

$$\frac{32}{429}$$

Example 3: Order statistics

A bag contains 15 billiard balls, numbered 1 to 15. If 7 balls are drawn from the bag at random, find the probability that the median number drawn is 5 when (*a*) sampling is performed without replacement; (*b*) sampling is performed with replacement.

(*a*) Sampling without replacement

```
> X := UniformDiscreteRV(1, 15);
> Y := OrderStat(X, 7, 4, "wo");
> PDF(Y, 5);
```

$$\frac{32}{429}$$

Example 3: Order statistics

A bag contains 15 billiard balls, numbered 1 to 15. If 7 balls are drawn from the bag at random, find the probability that the median number drawn is 5 when (a) sampling is performed without replacement; (b) sampling is performed with replacement.

(a) Sampling without replacement

```
> X := UniformDiscreteRV(1, 15);
> Y := OrderStat(X, 7, 4, "wo");
> PDF(Y, 5);
```

$$\frac{32}{429}$$

Example 3: Order statistics

A bag contains 15 billiard balls, numbered 1 to 15. If 7 balls are drawn from the bag at random, find the probability that the median number drawn is 5 when (*a*) sampling is performed without replacement; (*b*) sampling is performed with replacement.

(*a*) Sampling without replacement

```
> X := UniformDiscreteRV(1, 15);
> Y := OrderStat(X, 7, 4, "wo");
> PDF(Y, 5);
```

$$\frac{32}{429}$$

Example 3: Order statistics

A bag contains 15 billiard balls, numbered 1 to 15. If 7 balls are drawn from the bag at random, find the probability that the median number drawn is 5 when (*a*) sampling is performed without replacement; (*b*) sampling is performed with replacement.

(*a*) Sampling without replacement

```
> X := UniformDiscreteRV(1, 15);
> Y := OrderStat(X, 7, 4, "wo");
> PDF(Y, 5);
```

$$\frac{32}{429}$$

Example 3: Order statistics (continued)

A bag contains 15 billiard balls, numbered 1 to 15. If 7 balls are drawn from the bag at random, find the probability that the median number drawn is 5 when (a) sampling is performed without replacement; (b) sampling is performed with replacement.

(b) Sampling with replacement

```
> X := UniformDiscreteRV(1, 15);
> Y := OrderStat(X, 7, 4);
> PDF(Y, 5);
```

$$\frac{2949971}{34171875}$$

Example 3: Order statistics (continued)

A bag contains 15 billiard balls, numbered 1 to 15. If 7 balls are drawn from the bag at random, find the probability that the median number drawn is 5 when (*a*) sampling is performed without replacement; (*b*) sampling is performed with replacement.

(*b*) Sampling with replacement

```
> X := UniformDiscreteRV(1, 15);
> Y := OrderStat(X, 7, 4);
> PDF(Y, 5);
```

$$\frac{2949971}{34171875}$$

Example 3: Order statistics (continued)

A bag contains 15 billiard balls, numbered 1 to 15. If 7 balls are drawn from the bag at random, find the probability that the median number drawn is 5 when (*a*) sampling is performed without replacement; (*b*) sampling is performed with replacement.

(*b*) Sampling with replacement

```
> X := UniformDiscreteRV(1, 15);
> Y := OrderStat(X, 7, 4);
> PDF(Y, 5);
```

$$\frac{2949971}{34171875}$$

Example 3: Order statistics (continued)

A bag contains 15 billiard balls, numbered 1 to 15. If 7 balls are drawn from the bag at random, find the probability that the median number drawn is 5 when (*a*) sampling is performed without replacement; (*b*) sampling is performed with replacement.

(*b*) Sampling with replacement

```
> X := UniformDiscreteRV(1, 15);
> Y := OrderStat(X, 7, 4);
> PDF(Y, 5);
```

$$\frac{2949971}{34171875}$$

Example 3: Order statistics (continued)

A bag contains 15 billiard balls, numbered 1 to 15. If 7 balls are drawn from the bag at random, find the probability that the median number drawn is 5 when (*a*) sampling is performed without replacement; (*b*) sampling is performed with replacement.

(*b*) Sampling with replacement

```
> X := UniformDiscreteRV(1, 15);
> Y := OrderStat(X, 7, 4);
> PDF(Y, 5);
```

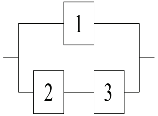
$$\frac{2949971}{34171875}$$

Example 4: Systems reliability

Let T_1, T_2, T_3 be independent and identically distributed Weibull(2, 3) random variables with survivor function

$$S_i(t) = e^{-(2t)^3} \qquad t > 0; i = 1, 2, 3$$

what is the mean system lifetime?



APPL solution

```
> T1 := WeibullRV(2, 3);
> T2 := WeibullRV(2, 3);
> T3 := WeibullRV(2, 3);
> T  := Maximum(T1, Minimum(T2, T3));
> Mean(T);
```

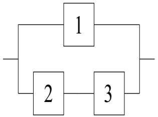
$$\frac{\pi\left(\sqrt{3}2^{2/3} + 2\sqrt{3} - 2\sqrt[6]{3}\right)}{18\Gamma\left(2/3\right)} \cong 0.4913$$

Example 4: Systems reliability

Let T_1, T_2, T_3 be independent and identically distributed Weibull(2, 3) random variables with survivor function

$$S_i(t) = e^{-(2t)^3} \qquad t > 0;\, i = 1, 2, 3$$

what is the mean system lifetime?



APPL solution

```
> T1 := WeibullRV(2, 3);
> T2 := WeibullRV(2, 3);
> T3 := WeibullRV(2, 3);
> T  := Maximum(T1, Minimum(T2, T3));
> Mean(T);
```

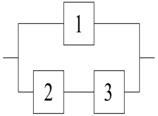
$$\frac{\pi\left(\sqrt{3}2^{2/3} + 2\sqrt{3} - 2\sqrt[6]{3}\right)}{18\Gamma(2/3)} \cong 0.4913$$

Example 4: Systems reliability

Let T_1, T_2, T_3 be independent and identically distributed Weibull(2, 3) random variables with survivor function

$$S_i(t) = e^{-(2t)^3} \qquad t > 0; i = 1, 2, 3$$

what is the mean system lifetime?



APPL solution

```
> T1 := WeibullRV(2, 3);
> T2 := WeibullRV(2, 3);
> T3 := WeibullRV(2, 3);
> T  := Maximum(T1, Minimum(T2, T3));
> Mean(T);
```

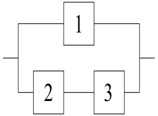
$$\frac{\pi\left(\sqrt{3}2^{2/3} + 2\sqrt{3} - 2\sqrt[6]{3}\right)}{18\Gamma\left(2/3\right)} \cong 0.4913$$

Example 4: Systems reliability

Let T_1, T_2, T_3 be independent and identically distributed Weibull(2, 3) random variables with survivor function

$$S_i(t) = e^{-(2t)^3} \qquad t > 0; i = 1, 2, 3$$

what is the mean system lifetime?



APPL solution

```
> T1 := WeibullRV(2, 3);
> T2 := WeibullRV(2, 3);
> T3 := WeibullRV(2, 3);
> T  := Maximum(T1, Minimum(T2, T3));
> Mean(T);
```

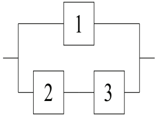
$$\frac{\pi\left(\sqrt{3}2^{2/3} + 2\sqrt{3} - 2\sqrt[6]{3}\right)}{18\Gamma\left(2/3\right)} \cong 0.4913$$

Example 4: Systems reliability

Let T_1, T_2, T_3 be independent and identically distributed Weibull(2, 3) random variables with survivor function

$$S_i(t) = e^{-(2t)^3} \qquad t > 0; i = 1, 2, 3$$

what is the mean system lifetime?



APPL solution

```
> T1 := WeibullRV(2, 3);
> T2 := WeibullRV(2, 3);
> T3 := WeibullRV(2, 3);
> T  := Maximum(T1, Minimum(T2, T3));
> Mean(T);
```

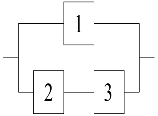
$$\frac{\pi\left(\sqrt{3}2^{2/3} + 2\sqrt{3} - 2\sqrt[6]{3}\right)}{18\Gamma(2/3)} \cong 0.4913$$

Example 4: Systems reliability

Let T_1, T_2, T_3 be independent and identically distributed Weibull(2, 3) random variables with survivor function

$$S_i(t) = e^{-(2t)^3} \qquad t > 0;\, i = 1, 2, 3$$

what is the mean system lifetime?



APPL solution

```
> T1 := WeibullRV(2, 3);
> T2 := WeibullRV(2, 3);
> T3 := WeibullRV(2, 3);
> T  := Maximum(T1, Minimum(T2, T3));
> Mean(T);
```

$$\frac{\pi\left(\sqrt{3}2^{2/3} + 2\sqrt{3} - 2\sqrt[6]{3}\right)}{18\Gamma\left(2/3\right)} \cong 0.4913$$

Applications

1. Bootstrapping
2. Kolmogorov–Smirnov test statistic
3. Stochastic activity networks
4. Lower bound on system reliability
5. Benford's law
6. Control chart constants

Application 1: Bootstrapping

Is there a difference between the medians of the rat survival times (days) of the two populations? Strategy: estimate the standard error of the difference between the medians.

Group	Data	n	Median
Treatment	16, 23, 38, 94, 99, 141, 197	7	94
Control	10, 27, 30, 40, 46, 51, 52, 104, 146	9	46

Generate B bootstrap samples, each of which consists of $n = 7$ samples drawn with replacement from 16, 23, 38, 94, 99, 141, and 197. Calculate sample standard deviation of the medians.

Application 1: Bootstrapping

Is there a difference between the medians of the rat survival times (days) of the two populations? Strategy: estimate the standard error of the difference between the medians.

Group	Data	n	Median
Treatment	16, 23, 38, 94, 99, 141, 197	7	94
Control	10, 27, 30, 40, 46, 51, 52, 104, 146	9	46

Generate B bootstrap samples, each of which consists of $n = 7$ samples drawn with replacement from 16, 23, 38, 94, 99, 141, and 197. Calculate sample standard deviation of the medians.

S-Plus solution

```
> set.seed(1)
> tr <- c(16, 23, 38, 94, 99, 141, 197)
> medn <- function(x) quantile(x, 0.50)
> bootstrap(tr, medn, B = 50)
```

This returns the estimated standard error of the median of the treatment group as

$$41.18$$

Application 1: Bootstrapping (continued)

S-Plus solution

```
> set.seed(1)
> tr <- c(16, 23, 38, 94, 99, 141, 197)
> medn <- function(x) quantile(x, 0.50)
> bootstrap(tr, medn, B = 50)
```

This returns the estimated standard error of the median of the treatment group as

$$41.18$$

Application 1: Bootstrapping (continued)

S-Plus solution

```
> set.seed(1)
> tr <- c(16, 23, 38, 94, 99, 141, 197)
> medn <- function(x) quantile(x, 0.50)
> bootstrap(tr, medn, B = 50)
```

This returns the estimated standard error of the median of the treatment group as

$$41.18$$

Application 1: Bootstrapping (continued)

S-Plus solution

```
> set.seed(1)
> tr <- c(16, 23, 38, 94, 99, 141, 197)
> medn <- function(x) quantile(x, 0.50)
> bootstrap(tr, medn, B = 50)
```

This returns the estimated standard error of the median of the treatment group as

$$41.18$$

Application 1: Bootstrapping (continued)

S-Plus solution

```
> set.seed(1)
> tr <- c(16, 23, 38, 94, 99, 141, 197)
> medn <- function(x) quantile(x, 0.50)
> bootstrap(tr, medn, B = 50)
```

This returns the estimated standard error of the median of the treatment group as

$$41.18$$

Application 1: Bootstrapping (continued)

S-Plus solution

```
> set.seed(1)
> tr <- c(16, 23, 38, 94, 99, 141, 197)
> medn <- function(x) quantile(x, 0.50)
> bootstrap(tr, medn, B = 50)
```

This returns the estimated standard error of the median of the treatment group as

$$41.18$$

Application 1: Bootstrapping (continued)

Bootstrap estimates of the standard error of the median:

	$B = 50$	$B = 100$	$B = 250$	$B = 1000$	$B = +\infty$
Treatment	41.18	37.63	36.88	38.98	37.83
Control	20.30	12.68	9.538	13.82	13.08

Application 1: Bootstrapping (continued)

Calculate the $B = +\infty$ column via the APPL statements:

```
> treatment := [16, 23, 38, 94, 99, 141, 197];
> X := BootstrapRV(treatment);
> Y := OrderStat(X, 7, 4);
> sqrt(Variance(Y));
```

$$f(y) = \begin{cases} 8359/823543 & y = 16 \\ 80809/823543 & y = 23 \\ 196519/823543 & y = 38 \\ 252169/823543 & y = 94 \\ 196519/823543 & y = 99 \\ 80809/823543 & y = 141 \\ 8359/823543 & y = 197 \end{cases}$$

Standard error: $\dfrac{2}{823543}\sqrt{242712738519382} \cong 37.8347$

Application 1: Bootstrapping (continued)

Calculate the $B = +\infty$ column via the APPL statements:

```
> treatment := [16, 23, 38, 94, 99, 141, 197];
> X := BootstrapRV(treatment);
> Y := OrderStat(X, 7, 4);
> sqrt(Variance(Y));
```

$$f(y) = \begin{cases} 8359/823543 & y = 16 \\ 80809/823543 & y = 23 \\ 196519/823543 & y = 38 \\ 252169/823543 & y = 94 \\ 196519/823543 & y = 99 \\ 80809/823543 & y = 141 \\ 8359/823543 & y = 197 \end{cases}$$

Standard error: $\dfrac{2}{823543}\sqrt{242712738519382} \cong 37.8347$

Application 1: Bootstrapping (continued)

Calculate the $B = +\infty$ column via the APPL statements:

```
> treatment := [16, 23, 38, 94, 99, 141, 197];
> X := BootstrapRV(treatment);
> Y := OrderStat(X, 7, 4);
> sqrt(Variance(Y));
```

$$f(y) = \begin{cases} 8359/823543 & y = 16 \\ 80809/823543 & y = 23 \\ 196519/823543 & y = 38 \\ 252169/823543 & y = 94 \\ 196519/823543 & y = 99 \\ 80809/823543 & y = 141 \\ 8359/823543 & y = 197 \end{cases}$$

Standard error: $\frac{2}{823543}\sqrt{24271273851938 2} \cong 37.8347$

Application 1: Bootstrapping (continued)

Calculate the $B = +\infty$ column via the APPL statements:

```
> treatment := [16, 23, 38, 94, 99, 141, 197];
> X := BootstrapRV(treatment);
> Y := OrderStat(X, 7, 4);
> sqrt(Variance(Y));
```

$$f(y) = \begin{cases} 8359/823543 & y = 16 \\ 80809/823543 & y = 23 \\ 196519/823543 & y = 38 \\ 252169/823543 & y = 94 \\ 196519/823543 & y = 99 \\ 80809/823543 & y = 141 \\ 8359/823543 & y = 197 \end{cases}$$

Standard error: $\frac{2}{823543}\sqrt{242712738519382} \cong 37.8347$

Application 1: Bootstrapping (continued)

Calculate the $B = +\infty$ column via the APPL statements:

```
> treatment := [16, 23, 38, 94, 99, 141, 197];
> X := BootstrapRV(treatment);
> Y := OrderStat(X, 7, 4);
> sqrt(Variance(Y));
```

$$f(y) = \begin{cases} 8359/823543 & y = 16 \\ 80809/823543 & y = 23 \\ 196519/823543 & y = 38 \\ 252169/823543 & y = 94 \\ 196519/823543 & y = 99 \\ 80809/823543 & y = 141 \\ 8359/823543 & y = 197 \end{cases}$$

Standard error: $\frac{2}{823543}\sqrt{242712738519382} \cong 37.8347$

Application 1: Bootstrapping (continued)

Calculate the $B = +\infty$ column via the APPL statements:

```
> treatment := [16, 23, 38, 94, 99, 141, 197];
> X := BootstrapRV(treatment);
> Y := OrderStat(X, 7, 4);
> sqrt(Variance(Y));
```

$$f(y) = \begin{cases} 8359/823543 & y = 16 \\ 80809/823543 & y = 23 \\ 196519/823543 & y = 38 \\ 252169/823543 & y = 94 \\ 196519/823543 & y = 99 \\ 80809/823543 & y = 141 \\ 8359/823543 & y = 197 \end{cases}$$

Standard error: $\frac{2}{823543}\sqrt{242712738519382} \cong 37.8347$

Application 1: Bootstrapping (continued)

Calculate the $B = +\infty$ column via the APPL statements:

```
> treatment := [16, 23, 38, 94, 99, 141, 197];
> X := BootstrapRV(treatment);
> Y := OrderStat(X, 7, 4);
> sqrt(Variance(Y));
```

$$f(y) = \begin{cases} 8359/823543 & y = 16 \\ 80809/823543 & y = 23 \\ 196519/823543 & y = 38 \\ 252169/823543 & y = 94 \\ 196519/823543 & y = 99 \\ 80809/823543 & y = 141 \\ 8359/823543 & y = 197 \end{cases}$$

Standard error: $\dfrac{2}{823543}\sqrt{242712738519382} \cong 37.8347$

Application 1: Bootstrapping (continued)

So is there a statistically significant difference between the medians?

Standard error treatment:

$$\frac{2}{823543}\sqrt{242712738519382} \cong 37.8347$$

Standard error control:

$$\frac{1}{387420489}\sqrt{25662937134123797402} \cong 13.0759$$

The seemingly large difference between the two sample medians, $94 - 46 = 48$ days, is only

$$48/\sqrt{37.83^2 + 13.08^2} \cong 1.19$$

standard-deviation units away from zero.

Conclusion: no statistically significant difference between the medians.

Application 1: Bootstrapping (continued)

So is there a statistically significant difference between the medians? Standard error treatment:

$$\frac{2}{823543}\sqrt{242712738519382} \cong 37.8347$$

Standard error control:

$$\frac{1}{387420489}\sqrt{2566293713412379 7402} \cong 13.0759$$

The seemingly large difference between the two sample medians, $94 - 46 = 48$ days, is only

$$48/\sqrt{37.83^2 + 13.08^2} \cong 1.19$$

standard-deviation units away from zero.

Conclusion: no statistically significant difference between the medians.

Application 1: Bootstrapping (continued)

So is there a statistically significant difference between the medians? Standard error treatment:

$$\frac{2}{823543}\sqrt{242712738519382} \cong 37.8347$$

Standard error control:

$$\frac{1}{387420489}\sqrt{25662937134123797402} \cong 13.0759$$

The seemingly large difference between the two sample medians, $94 - 46 = 48$ days, is only

$$48/\sqrt{37.83^2 + 13.08^2} \cong 1.19$$

standard-deviation units away from zero.

Conclusion: no statistically significant difference between the medians.

Application 1: Bootstrapping (continued)

So is there a statistically significant difference between the medians? Standard error treatment:

$$\frac{2}{823543}\sqrt{242712738519382} \cong 37.8347$$

Standard error control:

$$\frac{1}{387420489}\sqrt{25662937134123797402} \cong 13.0759$$

The seemingly large difference between the two sample medians, $94 - 46 = 48$ days, is only

$$48/\sqrt{37.83^2 + 13.08^2} \cong 1.19$$

standard-deviation units away from zero.

Conclusion: no statistically significant difference between the medians.

Application 1: Bootstrapping (continued)

So is there a statistically significant difference between the medians? Standard error treatment:

$$\frac{2}{823543}\sqrt{242712738519382} \cong 37.8347$$

Standard error control:

$$\frac{1}{387420489}\sqrt{25662937134123797402} \cong 13.0759$$

The seemingly large difference between the two sample medians, $94 - 46 = 48$ days, is only

$$48/\sqrt{37.83^2 + 13.08^2} \cong 1.19$$

standard-deviation units away from zero.

Conclusion: no statistically significant difference between the medians.

Application 2: Kolmogorov–Smirnov test statistic

Defining formula:

$$D_n = \sup_x |F(x) - F_n(x)|$$

Computational formula:

$$D_n = \max_{i=1,2,\ldots,n} \left\{ \left| \frac{i-1}{n} - x_{(i)} \right|, \left| \frac{i}{n} - x_{(i)} \right| \right\}$$

The CDF of D_n (all parameters known case, Birnbaum, 1952):

$$P\left(D_n < \frac{1}{2n} + v\right) = n! \int_{\frac{1}{2n}-v}^{\frac{1}{2n}+v} \int_{\frac{3}{2n}-v}^{\frac{3}{2n}+v} \cdots \int_{\frac{2n-1}{2n}-v}^{\frac{2n-1}{2n}+v} g(u_1, u_2, \ldots, u_n)\, du_n \ldots du_2\, du_1$$

for $0 \le v \le \frac{2n-1}{2n}$, where

$$g(u_1, u_2, \ldots, u_n) = 1$$

for $0 \le u_1 \le u_2 \le \cdots \le u_n$.

Application 2: Kolmogorov–Smirnov test statistic (continued)

CASE I: $n = 1$

$$F_{D_1}(t) = \Pr(D_1 \le t) = \begin{cases} 0 & t \le \frac{1}{2} \\ 2t - 1 & \frac{1}{2} < t < 1 \\ 1 & t \ge 1 \end{cases}$$

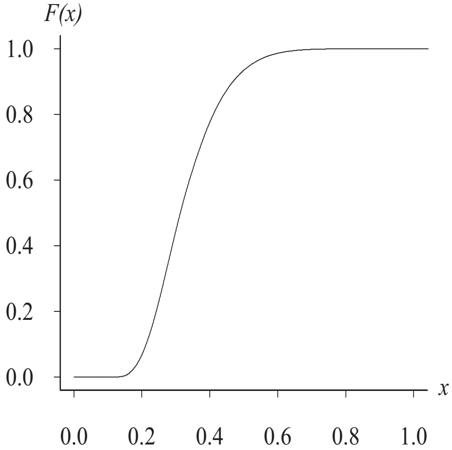
CASE II: $n = 2$

$$F_{D_2}(t) = \Pr(D_2 \le t) = \begin{cases} 0 & t \le \frac{1}{4} \\ 8\left(t - \frac{1}{4}\right)^2 & \frac{1}{4} < t < \frac{1}{2} \\ 1 - 2(1 - t)^2 & \frac{1}{2} < t < 1 \\ 1 & t \ge 1 \end{cases}$$

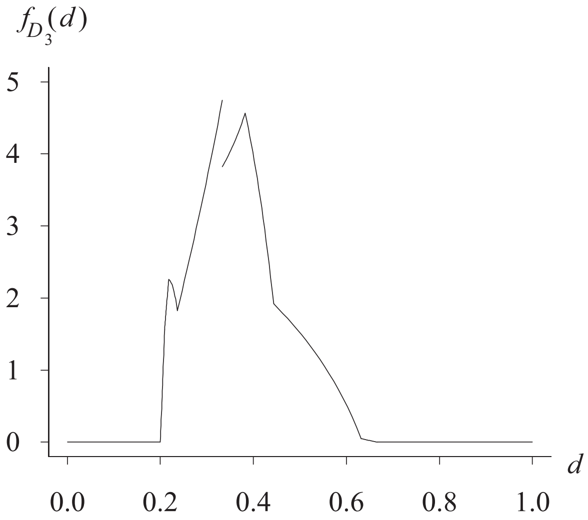
Application 2: Kolmogorov–Smirnov test statistic (continued)

Goal: `X := KSRV(n);`

CASE III: $n = 6$

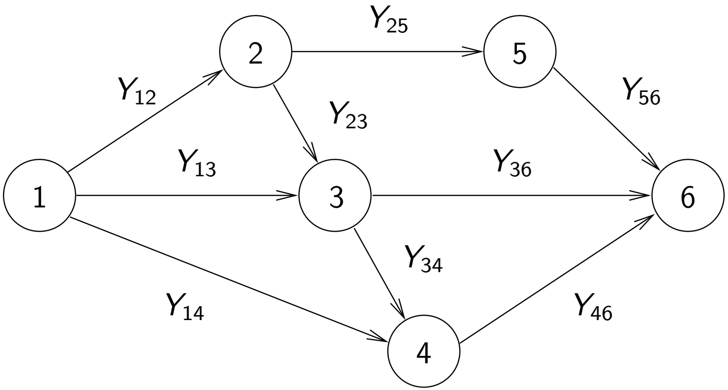


Application 2: Kolmogorov–Smirnov test statistic (continued)



Application 3: Stochastic activity networks

Stochastic activity networks arise in *project management*



Our goal: find the distribution of T_6, the time to complete the network

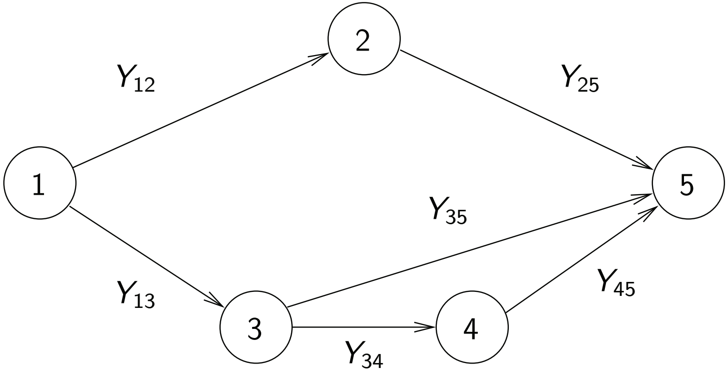
Application 3: Stochastic activity networks

Popular analysis techniques

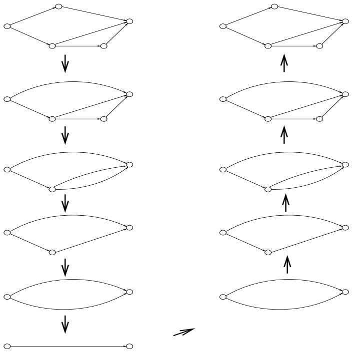
- CPM
- PERT
- Simulation

Application 3: Stochastic activity networks (continued)

Series–parallel networks constitute a class of stochastic activity networks that are easy to analyze. This sample series–parallel network is from Elmaghraby (1977, p. 261).



Application 3: Stochastic activity networks (continued)



Application 3: Stochastic activity networks (continued)

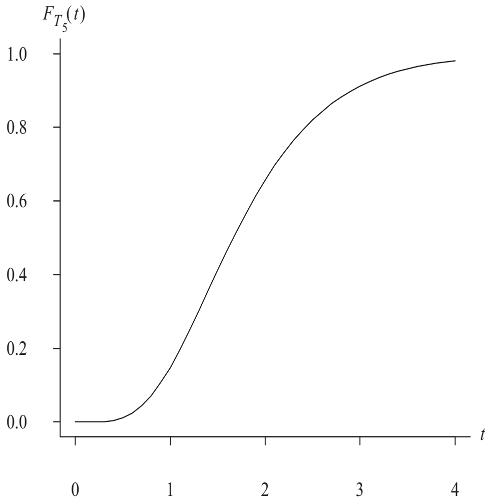
When all arc durations are independent exponential(b) random variables, where b is a rate, the time to complete the network T_5 has cdf

$$F_{T_5}(t) = 1 - 3bte^{-bt} - \frac{b^2t^2}{2}e^{-bt} - 3e^{-2bt} + \frac{5b^2t^2}{2}e^{-2bt} + \frac{b^3t^3}{2}e^{-2bt}$$

$$+2e^{-3bt} + 3bte^{-3bt} + b^2t^2e^{-3bt} \qquad t > 0$$

Application 3: Stochastic activity networks (continued)

The cdf is shown below for $b = 2$



Application 3: Stochastic activity networks (continued)

Bonus material: for exponential(2) arc durations

Paths π_k and critical path probabilities $p(\pi_k)$

k	Node sequence	π_k	$p(\pi_k)$
1	$1 \to 2 \to 5$	$\{a_{12}, a_{25}\}$	$115/432 \cong 0.266$
2	$1 \to 3 \to 5$	$\{a_{13}, a_{35}\}$	$317/1728 \cong 0.183$
3	$1 \to 3 \to 4 \to 5$	$\{a_{13}, a_{34}, a_{45}\}$	$317/576 \cong 0.550$

Criticalities ρ_{ij}

Arc	Paths	ρ_{ij}
a_{12}	π_1	$115/432 \cong 0.266$
a_{13}	π_2, π_3	$317/432 \cong 0.734$
a_{25}	π_1	$115/432 \cong 0.266$
a_{35}	π_2	$317/1728 \cong 0.183$
a_{34}	π_3	$317/576 \cong 0.550$
a_{45}	π_3	$317/576 \cong 0.550$

Application 4: Lower bound on system reliability

Andy Glen
United States Military Academy
Larry Leemis
William & Mary

Bootstrapping in systems reliability

Use bootstrapping to determine a 95% lower confidence bound on the system reliability for a series system of three independent components using the binary failure data (y_i, n_i), where

- y_i is the number of components of type i that pass the test
- n_i is the number of components of type i on test

for $i = 1, 2, 3$

Component number	$i = 1$	$i = 2$	$i = 3$
Number passing (y_i)	21	27	82
Number on test (n_i)	23	28	84

Point estimate for the system reliability:

$$\frac{21}{23} \cdot \frac{27}{28} \cdot \frac{82}{84} = \frac{1107}{1288} \cong 0.8595$$

Application 4: Lower bound on system reliability

Bootstrapping in systems reliability

Use bootstrapping to determine a 95% lower confidence bound on the system reliability for a series system of three independent components using the binary failure data (y_i, n_i), where

- y_i is the number of components of type i that pass the test
- n_i is the number of components of type i on test

for $i = 1, 2, 3$

Component number	$i = 1$	$i = 2$	$i = 3$
Number passing (y_i)	21	27	82
Number on test (n_i)	23	28	84

Point estimate for the system reliability:

$$\frac{21}{23} \cdot \frac{27}{28} \cdot \frac{82}{84} = \frac{1107}{1288} \cong 0.8595$$

Application 4: Lower bound on system reliability

Bootstrapping in systems reliability

Use bootstrapping to determine a 95% lower confidence bound on the system reliability for a series system of three independent components using the binary failure data (y_i, n_i), where

- y_i is the number of components of type i that pass the test
- n_i is the number of components of type i on test

for $i = 1, 2, 3$

Component number	$i = 1$	$i = 2$	$i = 3$
Number passing (y_i)	21	27	82
Number on test (n_i)	23	28	84

Point estimate for the system reliability:

$$\frac{21}{23} \cdot \frac{27}{28} \cdot \frac{82}{84} = \frac{1107}{1288} \cong 0.8595$$

Application 4: Lower bound on system reliability


```
> X1 := BinomialRV(23, 21 / 23);
> X1 := Transform(X1, [[x -> x / 23], [0, 23]]);
> X2 := BinomialRV(28, 27 / 28);
> X2 := Transform(X2, [[x -> x / 28], [0, 28]]);
> X3 := BinomialRV(84, 82 / 84);
> X3 := Transform(X3, [[x -> x / 84], [0, 84]]);
> T := Product(X1, X2, X3);
```

- There are a possible $24 \cdot 29 \cdot 85 = 59,160$ potential mass values for T
- Of these, only 6633 are distinct because the Product procedure combines repeated values
- The lower 95% bootstrap confidence interval bound is the 0.05 fractile of the distribution of T, which is

$$6723/9016 \cong 0.7457$$

Application 4: Lower bound on system reliability

```
> X1 := BinomialRV(23, 21 / 23);
> X1 := Transform(X1, [[x -> x / 23], [0, 23]]);
> X2 := BinomialRV(28, 27 / 28);
> X2 := Transform(X2, [[x -> x / 28], [0, 28]]);
> X3 := BinomialRV(84, 82 / 84);
> X3 := Transform(X3, [[x -> x / 84], [0, 84]]);
> T := Product(X1, X2, X3);
```

- There are a possible $24 \cdot 29 \cdot 85 = 59,160$ potential mass values for T
- Of these, only 6633 are distinct because the Product procedure combines repeated values
- The lower 95% bootstrap confidence interval bound is the 0.05 fractile of the distribution of T, which is

$$6723/9016 \cong 0.7457$$

Application 4: Lower bound on system reliability


```
> X1 := BinomialRV(23, 21 / 23);
> X1 := Transform(X1, [[x -> x / 23], [0, 23]]);
> X2 := BinomialRV(28, 27 / 28);
> X2 := Transform(X2, [[x -> x / 28], [0, 28]]);
> X3 := BinomialRV(84, 82 / 84);
> X3 := Transform(X3, [[x -> x / 84], [0, 84]]);
> T := Product(X1, X2, X3);
```

- There are a possible $24 \cdot 29 \cdot 85 = 59,160$ potential mass values for T
- Of these, only 6633 are distinct because the Product procedure combines repeated values
- The lower 95% bootstrap confidence interval bound is the 0.05 fractile of the distribution of T, which is

$$6723/9016 \cong 0.7457$$

Application 4: Lower bound on system reliability


```
> X1 := BinomialRV(23, 21 / 23);
> X1 := Transform(X1, [[x -> x / 23], [0, 23]]);
> X2 := BinomialRV(28, 27 / 28);
> X2 := Transform(X2, [[x -> x / 28], [0, 28]]);
> X3 := BinomialRV(84, 82 / 84);
> X3 := Transform(X3, [[x -> x / 84], [0, 84]]);
> T := Product(X1, X2, X3);
```

- There are a possible $24 \cdot 29 \cdot 85 = 59,160$ potential mass values for T
- Of these, only 6633 are distinct because the `Product` procedure combines repeated values
- The lower 95% bootstrap confidence interval bound is the 0.05 fractile of the distribution of T, which is

$$6723/9016 \cong 0.7457$$

Application 4: Lower bound on system reliability


```
> X1 := BinomialRV(23, 21 / 23);
> X1 := Transform(X1, [[x -> x / 23], [0, 23]]);
> X2 := BinomialRV(28, 27 / 28);
> X2 := Transform(X2, [[x -> x / 28], [0, 28]]);
> X3 := BinomialRV(84, 82 / 84);
> X3 := Transform(X3, [[x -> x / 84], [0, 84]]);
> T := Product(X1, X2, X3);
```

- There are a possible $24 \cdot 29 \cdot 85 = 59,160$ potential mass values for `T`
- Of these, only 6633 are distinct because the `Product` procedure combines repeated values
- The lower 95% bootstrap confidence interval bound is the 0.05 fractile of the distribution of `T`, which is

$$6723/9016 \cong 0.7457$$

Application 5: Benford's law

Benford's law

- concerns the distribution of the leading digit in a data set
- Simon Newcomb (1881) noticed that the early pages of logarithm tables were more worn than the later pages
- if X denotes the leading digit

$$f_X(x) = P(X = x) = \log_{10}\left(1 + \frac{1}{x}\right) \qquad x = 1, 2, \ldots, 9$$

- $P(X = 1) = 0.301$; $P(X = 9) = 0.0458$
- Frank Benford (1938) fit the distribution to a wide variety of data sets
- applications: election fraud, accounting fraud
- the goal: search for probability distributions satisfying Benford's law exactly

Application 5: Benford's law

Benford's law

- concerns the distribution of the leading digit in a data set
- Simon Newcomb (1881) noticed that the early pages of logarithm tables were more worn than the later pages
- if X denotes the leading digit

$$f_X(x) = P(X = x) = \log_{10}\left(1 + \frac{1}{x}\right) \qquad x = 1, 2, \ldots, 9$$

- $P(X = 1) = 0.301$; $P(X = 9) = 0.0458$
- Frank Benford (1938) fit the distribution to a wide variety of data sets
- applications: election fraud, accounting fraud
- the goal: search for probability distributions satisfying Benford's law exactly

Application 5: Benford's law

Benford's law

- concerns the distribution of the leading digit in a data set
- Simon Newcomb (1881) noticed that the early pages of logarithm tables were more worn than the later pages
- if X denotes the leading digit

$$f_X(x) = P(X = x) = \log_{10}\left(1 + \frac{1}{x}\right) \qquad x = 1, 2, \ldots, 9$$

- $P(X = 1) = 0.301$; $P(X = 9) = 0.0458$
- Frank Benford (1938) fit the distribution to a wide variety of data sets
- applications: election fraud, accounting fraud
- the goal: search for probability distributions satisfying Benford's law exactly

Application 5: Benford's law

Benford's law

- concerns the distribution of the leading digit in a data set
- Simon Newcomb (1881) noticed that the early pages of logarithm tables were more worn than the later pages
- if X denotes the leading digit

$$f_X(x) = P(X = x) = \log_{10}\left(1 + \frac{1}{x}\right) \qquad x = 1, 2, \ldots, 9$$

- $P(X = 1) = 0.301$; $P(X = 9) = 0.0458$
- Frank Benford (1938) fit the distribution to a wide variety of data sets
- applications: election fraud, accounting fraud
- the goal: search for probability distributions satisfying Benford's law exactly

Application 5: Benford's law

Benford's law

- concerns the distribution of the leading digit in a data set
- Simon Newcomb (1881) noticed that the early pages of logarithm tables were more worn than the later pages
- if X denotes the leading digit

$$f_X(x) = P(X = x) = \log_{10}\left(1 + \frac{1}{x}\right) \qquad x = 1, 2, \ldots, 9$$

- $P(X = 1) = 0.301$; $P(X = 9) = 0.0458$
- Frank Benford (1938) fit the distribution to a wide variety of data sets
- applications: election fraud, accounting fraud
- the goal: search for probability distributions satisfying Benford's law exactly

Application 5: Benford's law

Benford's law

- concerns the distribution of the leading digit in a data set
- Simon Newcomb (1881) noticed that the early pages of logarithm tables were more worn than the later pages
- if X denotes the leading digit

$$f_X(x) = P(X = x) = \log_{10}\left(1 + \frac{1}{x}\right) \qquad x = 1, 2, \ldots, 9$$

- $P(X = 1) = 0.301$; $P(X = 9) = 0.0458$
- Frank Benford (1938) fit the distribution to a wide variety of data sets
- applications: election fraud, accounting fraud
- the goal: search for probability distributions satisfying Benford's law exactly

Application 5: Benford's law

Benford's law

- concerns the distribution of the leading digit in a data set
- Simon Newcomb (1881) noticed that the early pages of logarithm tables were more worn than the later pages
- if X denotes the leading digit

$$f_X(x) = P(X = x) = \log_{10}\left(1 + \frac{1}{x}\right) \qquad x = 1, 2, \ldots, 9$$

- $P(X = 1) = 0.301$; $P(X = 9) = 0.0458$
- Frank Benford (1938) fit the distribution to a wide variety of data sets
- applications: election fraud, accounting fraud
- the goal: search for probability distributions satisfying Benford's law exactly

Application 5: Benford's law

Benford's law

- concerns the distribution of the leading digit in a data set
- Simon Newcomb (1881) noticed that the early pages of logarithm tables were more worn than the later pages
- if X denotes the leading digit

$$f_X(x) = P(X = x) = \log_{10}\left(1 + \frac{1}{x}\right) \qquad x = 1, 2, \ldots, 9$$

- $P(X = 1) = 0.301$; $P(X = 9) = 0.0458$
- Frank Benford (1938) fit the distribution to a wide variety of data sets
- applications: election fraud, accounting fraud
- the goal: search for probability distributions satisfying Benford's law exactly

Application 5: Benford's law (continued)

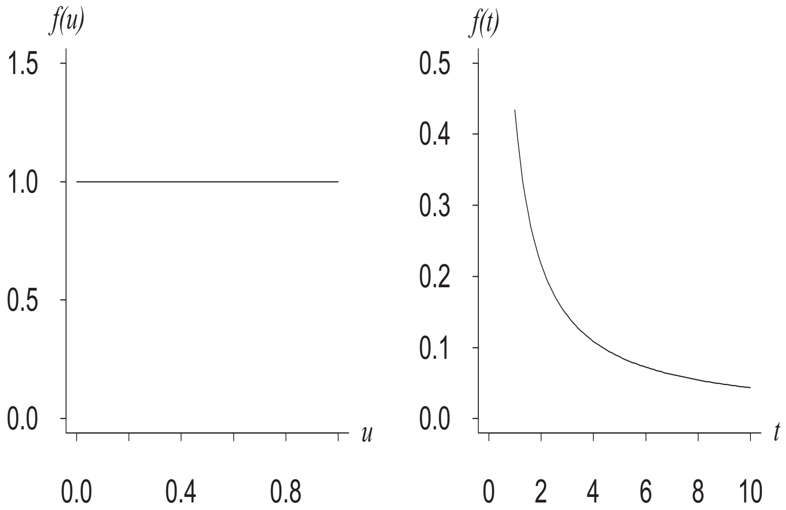
Eureka!
The variate generation algorithm

$$X \leftarrow \lfloor 10^U \rfloor$$

yields probability distribution that satisfies Benford's law exactly: If $U \sim U(0, 1)$ and $T = 10^U$ then T has probability density function

$$f_T(t) = \frac{1}{t \ln 10} \qquad 1 < t < 10$$

Application 5: Benford's law (continued)



Application 6: Control chart constants

Constants d_2, d_3 relate to the sample range R

Given a random sample $X_1, X_2, \ldots, X_n$ from a population with

- cumulative distribution function $F(x)$
- probability density function $f(x)$
- finite unknown variance σ_X^2
- associated order statistics $X_{(1)}, X_{(2)}, \ldots, X_{(n)}$

The sample range, R, is

$$R = X_{(n)} - X_{(1)}$$

The expected value of R can be expressed two ways

$$E[R] = d_2 \sigma_X$$

$$E[R] = E\left[X_{(n)}\right] - E\left[X_{(1)}\right]$$

Thus

$$d_2 = \frac{E[R]}{\sigma_X} = \frac{E\left[X_{(n)}\right] - E\left[X_{(1)}\right]}{\sigma_X}$$

Application 6: Control chart constants (continued)

Example 1. Calculate d_2 for a random sample X_1, X_2, X_3 drawn from a normal population

The APPL statements

```
> n := 3;
> X := NormalRV(0, sigma):
> (Mean(OrderStat(X, n, n)) -
>    Mean(OrderStat(X, n, 1))) / sqrt(Variance(X));
```

yield the exact value

$$d_2 = 3/\sqrt{\pi}$$

Application 6: Control chart constants (continued)

Example 1. Calculate d_2 for a random sample X_1, X_2, X_3 drawn from a normal population

The APPL statements

```
> n := 3;
> X := NormalRV(0, sigma):
> (Mean(OrderStat(X, n, n)) -
>    Mean(OrderStat(X, n, 1))) / sqrt(Variance(X));
```

yield the exact value

$$d_2 = 3/\sqrt{\pi}$$

Application 6: Control chart constants (continued)

Example 1. Calculate d_2 for a random sample X_1, X_2, X_3 drawn from a normal population

The APPL statements

```
> n := 3;
> X := NormalRV(0, sigma):
> (Mean(OrderStat(X, n, n)) -
>    Mean(OrderStat(X, n, 1))) / sqrt(Variance(X));
```

yield the exact value

$$d_2 = 3/\sqrt{\pi}$$

Application 6: Control chart constants (continued)

Example 2. Calculate the bias correction factors d_2 and $d_3 = \sigma_R/\sigma_X$ for a random sample X_1, X_2, X_3 from an exponential(λ) population. The APPL statements

```
> n := 3:
> X := ExponentialRV(lambda):
> R := RangeStat(X, n):
> d2 := Mean(R) / sqrt(Variance(X));
> d3 := sqrt(Variance(R)) / sqrt(Variance(X));
```

yield

$$d_2 = 3/2 \quad \text{and} \quad d_3 = \sqrt{5/2} \cong 1.118$$

Likewise, when $n = 18$,

$$d_2 = \frac{42142223}{12252240} \cong 3.440 \quad \text{and} \quad d_3 = \frac{\sqrt{238357395880861}}{12252240} \cong 1.260$$

Application 6: Control chart constants (continued)

Example 2. Calculate the bias correction factors d_2 and $d_3 = \sigma_R/\sigma_X$ for a random sample X_1, X_2, X_3 from an exponential(λ) population. The APPL statements

```
> n := 3:
> X := ExponentialRV(lambda):
> R := RangeStat(X, n):
> d2 := Mean(R) / sqrt(Variance(X));
> d3 := sqrt(Variance(R)) / sqrt(Variance(X));
```

yield

$$d_2 = 3/2 \quad \text{and} \quad d_3 = \sqrt{5/2} \cong 1.118$$

Likewise, when $n = 18$,

$$d_2 = \frac{42142223}{12252240} \cong 3.440 \quad \text{and} \quad d_3 = \frac{\sqrt{238357395880861}}{12252240} \cong 1.260$$

Application 6: Control chart constants (continued)

Example 2. Calculate the bias correction factors d_2 and $d_3 = \sigma_R / \sigma_X$ for a random sample X_1, X_2, X_3 from an exponential(λ) population. The APPL statements

```
> n := 3:
> X := ExponentialRV(lambda):
> R := RangeStat(X, n):
> d2 := Mean(R) / sqrt(Variance(X));
> d3 := sqrt(Variance(R)) / sqrt(Variance(X));
yield
```

$$d_2 = 3/2 \quad \text{and} \quad d_3 = \sqrt{5/2} \cong 1.118$$

Likewise, when $n = 18$,

$$d_2 = \frac{42142223}{12252240} \cong 3.440 \quad \text{and} \quad d_3 = \frac{\sqrt{238357395880861}}{12252240} \cong 1.260$$

Application 6: Control chart constants (continued)

Example 2. Calculate the bias correction factors d_2 and $d_3 = \sigma_R / \sigma_X$ for a random sample X_1, X_2, X_3 from an exponential(λ) population. The APPL statements

```
> n := 3:
> X := ExponentialRV(lambda):
> R := RangeStat(X, n):
> d2 := Mean(R) / sqrt(Variance(X));
> d3 := sqrt(Variance(R)) / sqrt(Variance(X));
yield
```

$$d_2 = 3/2 \quad \text{and} \quad d_3 = \sqrt{5/2} \cong 1.118$$

Likewise, when
$n = 18$,

$$d_2 = \frac{42142223}{12252240} \cong 3.440 \quad \text{and} \quad d_3 = \frac{\sqrt{238357395880861}}{12252240} \cong 1.260$$