Topics

• Finish Chapter 7 material on testing runs, $\chi^2$ test
• Chapter 8 – generating random variables with a desired distribution
From Last Time –
Kolmogorov-Smirnov Test

Uniform Distribution

F(x)
1

f(x)

x
From Last Time –
Kolmogorov-Smirnov Test

\[ D = \max \left( \sup_{x \in [0,1]} |F(x) - \frac{x}{1}|, \inf_{x \in [0,1]} |F(x) - \frac{x}{1}| \right) \]
From Last Time –
Kolmogorov-Smirnov Test

$F(x)$

Uniform Distribution

$f(x)$

$x$
From Last Time –
Kolmogorov-Smirnov Test

Uniform Distribution

F(x)
1

x

f(x)
1
From Last Time –
Kolmogorov-Smirnov Test

F(x)
1

Uniform Distribution

f(x)

x

1

1

Uniform Distribution
From Last Time –
Kolmogorov-Smirnov Test

Uniform Distribution
\( \chi^2 \) Tests

- **Observed Random variates**
- **Assumed Distribution**

Determine class intervals

- \( E_i \geq 5, \)
  - uniform → equal class sizes
  - nonuniform → equal probability classes

- **Observed # in interval \( i, O_i \)**
- **Expected # in interval \( i, E_i \)**

\[ \chi^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i} \]

- Do not reject \( H_0 \) if \( \chi^2 \leq \chi^2_{\text{critical}} \)
- Reject \( H_0 \) if \( \chi^2 > \chi^2_{\text{critical}} \)
Example 7.7

Observed values

\[
\begin{bmatrix}
0.34 & 0.90 & 0.25 & 0.89 & 0.87 & 0.44 & 0.12 & 0.21 & 0.46 & 0.67 \\
0.83 & 0.76 & 0.79 & 0.64 & 0.70 & 0.81 & 0.94 & 0.74 & 0.22 & 0.74 \\
0.96 & 0.99 & 0.77 & 0.67 & 0.56 & 0.41 & 0.52 & 0.73 & 0.99 & 0.02 \\
0.47 & 0.30 & 0.17 & 0.82 & 0.56 & 0.05 & 0.45 & 0.31 & 0.78 & 0.05 \\
0.79 & 0.71 & 0.23 & 0.19 & 0.82 & 0.93 & 0.65 & 0.37 & 0.39 & 0.42 \\
0.99 & 0.17 & 0.99 & 0.46 & 0.05 & 0.66 & 0.10 & 0.42 & 0.18 & 0.39 \\
0.37 & 0.51 & 0.54 & 0.01 & 0.81 & 0.28 & 0.69 & 0.34 & 0.75 & 0.49 \\
0.72 & 0.43 & 0.56 & 0.97 & 0.30 & 0.94 & 0.96 & 0.58 & 0.73 & 0.05 \\
0.06 & 0.39 & 0.84 & 0.24 & 0.40 & 0.64 & 0.40 & 0.19 & 0.79 & 0.62 \\
0.18 & 0.26 & 0.97 & 0.88 & 0.64 & 0.47 & 0.60 & 0.11 & 0.29 & 0.78
\end{bmatrix}
\]

Compute terms:

Intervals: Observed: Expected: Difference: Difference^2: Normalized:

\[
\begin{align*}
a &= \begin{bmatrix}
0 & 1 \cdot 10^{-15} \\
0 & 8 \\
1 & 0.1 \\
2 & 0.2 \\
3 & 0.3 \\
4 & 0.4 \\
5 & 0.5 \\
6 & 0.6 \\
7 & 0.7 \\
8 & 0.8 \\
9 & 0.9 \\
10 & 1
\end{bmatrix} &
E &= \begin{bmatrix}
0 & 10 \\
0 & 10 \\
1 & 10 \\
2 & 10 \\
3 & 10 \\
4 & 10 \\
5 & 10 \\
6 & 10 \\
7 & 10 \\
8 & 10 \\
9 & 10 \\
10 & 10
\end{bmatrix}
\end{align*}
\]

\[
\begin{align*}
O - E &= \begin{bmatrix}
0 & -2 \\
1 & -2 \\
2 & 0 \\
3 & -1 \\
4 & 2 \\
5 & -2 \\
6 & 0 \\
7 & 4 \\
8 & 0 \\
9 & 1
\end{bmatrix}
\end{align*}
\]

\[
\begin{align*}
(O - E)^2 &= \begin{bmatrix}
0 & 4 \\
1 & 4 \\
2 & 0 \\
3 & 1 \\
4 & 4 \\
5 & 4 \\
6 & 0 \\
7 & 16 \\
8 & 0 \\
9 & 1
\end{bmatrix}
\end{align*}
\]

\[
T &= \begin{bmatrix}
0 & 4 \\
1 & 0.4 \\
2 & 0 \\
3 & 0.1 \\
4 & 0.4 \\
5 & 0.4 \\
6 & 0 \\
7 & 1.6 \\
8 & 0 \\
9 & 0.1
\end{bmatrix}
\]

\[
\chi^2 := \sum_{i=0}^{9} \frac{(O_i - E_i)^2}{E_i}
\]

\[
\chi^2 = 3.4
\]

\[
\chi^2_{0.05,9} = 16.9
\]
Why use Frequency Tests?

• Consider the sequence of random numbers:
  .03, .01, .04, .01,
  .15, .19, .12, .16,
  .25, .24, .23, .21,
  .34, .31, .35, .39,
  .42, .46, .45, .44,
  ..., 
  .93, .91, .94, .91

• They would pass the K-S and chi-square test, but are not suitable as random numbers

• Frequency tests will find their deficiencies
Run Tests

• A *run* is a succession of similar events

• Coin flipping example:
  
  \[ H \ T \ T \ H \ T \ T \ H \ T \]

  – six runs are marked

• For sequences of random numbers, define *up* runs and *down* runs, depending on whether successive numbers are increasing or decreasing. E.g.:

  \[
  \begin{array}{ccccccccccccc}
  .87 & .15 & .23 & .45 & .69 & .32 & .30 & .19 & .24 & .18 & .65 & .82 & .93 & .22 & .81 \\
  \downarrow & \uparrow & \uparrow & \uparrow & \downarrow & \downarrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow \\
  .08 & .18 & .23 & .36 & .42 & .55 & .63 & .72 & .89 & .91 \\
  \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow \\
  .08 & .93 & .15 & .96 & .26 & .84 & .28 & .79 & .36 & .57 \\
  \uparrow & \downarrow & \uparrow & \downarrow & \uparrow & \downarrow & \uparrow & \downarrow & \uparrow & \downarrow & \uparrow & \downarrow & \uparrow & \downarrow & \uparrow \\
  \end{array}
  \]

  These don’t look like any realistic random sequence
Testing Runs

• For a truly random sequence with \( N \) samples, mean and variance of number of runs, \( a \), are:

\[
\mu_a = \frac{2N - 1}{3} \\
\sigma^2_a = \frac{16N - 29}{90}
\]

• For \( N>20 \), this can be approximated by a normal distribution, \( N(\mu_a, \sigma^2_a) \)

• Observe the number of runs in the data, \( a \)
• Calculate test statistic:

\[
Z_0 = \frac{a - \mu_a}{\sigma_a} = \frac{a - \left[ (2N - 1)/3 \right]}{\sqrt{(16N - 29)/90}}
\]

• Compare to a normal distribution
Testing Runs

- Find $z_{\alpha/2}$ for significance $\alpha/2$ from Normal distribution tables.
- Check to see if statistic exceeds $z_{\alpha/2}$.
Random-Variate Generation

• Given a process to generate uniformly distributed random numbers, how to generate any arbitrary distribution
  – continuous and discrete valued R.V.s

• Techniques:
  – Inverse Transform
  – Convolution Method
  – Acceptance-Rejection

• Doesn’t the simulation environment have the distributions needed?
  – Not always, especially for special distributions
  – It’s useful to know how they work to understand constraints
Inverse Transform Technique

• Use exponential distribution to illustrate technique

• Given $R_i$ drawn from $U(0, 1)$, generate $X_i$ drawn from exponential distribution

• Most useful when the c.d.f., $F(x)$ can be readily inverted

\[
f(x) = \begin{cases} \lambda e^{-\lambda x}, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad F(x) = \int_{-\infty}^{x} f(t) dt = \begin{cases} 1-e^{-\lambda x}, & x \geq 0 \\ 0, & x < 0 \end{cases} \]

\[
\begin{align*}
1-e^{-\lambda X} & = R \\
e^{-\lambda X} & = 1 - R \\
-\lambda X & = \ln(1 - R) \\
X & = -\frac{1}{\lambda} \ln(1 - R)
\end{align*}
\]

• Distribution of $R$ and $1-R$ are identical, allowing simplification

\[
X = -\frac{1}{\lambda} \ln(R)
\]
Inverse Transform Technique
Exponential Distribution

Example 8.1: Generate 200 exponentially distributed random numbers, plot histogram

Generating c.d.f. from histogram:
Verifying that Inverse Transform Technique Generates R.V.s with Correct Distribution

\[ P(X_1 \leq x_0) = P(R_i \leq F(x_0)) = F(x_0) \]

- \( X_i \) is generated from \( R_i \), transformed by \( F( ) \)
- \( R_i \) is uniformly distributed on \((0,1)\)
Applying Inverse Transform Technique to Other Distributions

• **Weibull**

\[ f(x) = \begin{cases} \frac{\beta}{\alpha} x^{\beta-1} e^{-(x/\alpha)^\beta} & x \geq 0 \\ 0 & x < 0 \end{cases} \]

\[ F(x) = \begin{cases} 1 - e^{-(x/\alpha)^\beta} & x \geq 0 \\ 0 & x < 0 \end{cases} \]

\[ X = \alpha \left[ -\ln(R) \right]^{1/\beta} \]

• **Triangular**

\[ f(x) = \begin{cases} x, & 0 \leq x \leq 1 \\ 2 - x, & 1 < x \leq 2 \\ 0, & \text{otherwise} \end{cases} \]

\[ F(x) = \begin{cases} 0, & x \leq 0 \\ \frac{x^2}{2}, & 0 < x \leq 1 \\ 1 - \frac{(2-x)^2}{2}, & 1 < x \leq 2 \\ 1, & x > 2 \end{cases} \]

\[ X = \begin{cases} \sqrt{2R}, & 0 \leq R \leq \frac{1}{2} \\ 2 - \sqrt{2(1-R)}, & \frac{1}{2} < R \leq 1 \end{cases} \]
Examples of Inverse Transform Technique

Weibull Distribution

\[
N := 2000 \\
i := 0.. N - 1 \\
\alpha := 1 \\
R := \text{runif}(N, 0, 1) \\
\beta := 1.5 \\
X := \alpha \cdot (\ln(R))^{1/\beta} \\
M := 20 \\
h := \text{histogram}(M, X) \\
k := 0.. M - 1 \\
\frac{h_{k,1}}{N}
\]

Triangular Distribution

\[
N := 20000 \\
i := 0.. N - 1 \\
R := \text{runif}(N, 0, 1) \\
X_i := \begin{cases} 
\frac{1}{\beta}, & \text{if } R_i \leq \frac{1}{2} \\
\frac{2}{\beta} R_i - 1, & \text{if } \frac{1}{2} < R_i \leq \frac{3}{2} \\
2 - \frac{2}{\beta} R_i, & \text{if } \frac{3}{2} < R_i \leq 1 
\end{cases} \\
M := 50 \\
h := \text{histogram}(M, X) \\
k := 0.. M - 1 \\
\frac{h_{k,1} M}{N}
\]
Empirical Continuous Distributions

- If observed distribution is believed to be discrete, we can use lookup table method previously discussed. What if the distribution is known to be continuous?
  - Interpolate intermediate values: Example 8.2: 5 samples of data are available

<table>
<thead>
<tr>
<th>$i$</th>
<th>Interval $x_{(i-1)} &lt; x \leq x_{(i)}$</th>
<th>Probability $1/n$</th>
<th>Cumulative Probability, $i/n$</th>
<th>Slope, $a_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$0.0 &lt; x \leq 0.80$</td>
<td>0.2</td>
<td>0.2</td>
<td>4.00</td>
</tr>
<tr>
<td>2</td>
<td>$0.80 &lt; x \leq 1.24$</td>
<td>0.2</td>
<td>0.4</td>
<td>2.20</td>
</tr>
<tr>
<td>3</td>
<td>$1.24 &lt; x \leq 1.45$</td>
<td>0.2</td>
<td>0.6</td>
<td>1.05</td>
</tr>
<tr>
<td>4</td>
<td>$1.45 &lt; x \leq 1.83$</td>
<td>0.2</td>
<td>0.8</td>
<td>1.90</td>
</tr>
<tr>
<td>5</td>
<td>$1.83 &lt; x \leq 2.76$</td>
<td>0.2</td>
<td>1.0</td>
<td>4.65</td>
</tr>
</tbody>
</table>

- Line segment slopes:
  \[ a_i = \frac{x_{(i)} - x_{(i-1)}}{i/n - (i-1)/n} = \frac{x_{(i)} - x_{(i-1)}}{1/n} \]

- Inverse c.d.f.:
  \[ X = x_{(i-1)} + a_i \left( R - \frac{(i-1)}{n} \right) \] when \( (i-1)/n < R \leq i/n \)
Empirical Continuous Distributions

• Sometimes a large number of data samples are available to generate empirical distribution
  – not efficient, or always necessary to generate large number of interpolation segments
  – summarize available data into frequency distribution with smaller number of bins
  – fit continuous empirical c.d.f. to frequency distribution
Direct Transform for Normal Distributions

• Normal c.d.f.:

\[ \Phi(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt \]

– Inverse transform technique cannot be applied - inverse c.d.f. cannot be expressed in closed form.
Direct Transform for Normal Distributions

- $Z_1$ and $Z_2$ are normal R.V.s. In polar coordinates:
  \[ Z_1 = B \cos \theta \]
  \[ Z_2 = B \sin \theta \]

- $B^2$ has a chi-square distribution with 2 degrees of freedom - equivalent to an exponential distribution with mean 2. Using Inverse Transform technique:
  \[ B = \sqrt{-2 \ln R} \]

- $\theta$ is uniformly distributed on $(0, 2\pi)$ and is independent of $B$.

- Given independent, uniformly distributed $R_1$ and $R_2$, generate normal $Z_1$, $Z_2$:
  \[ Z_1 = \sqrt{-2 \ln R_1} \cos(2\pi R_2) \]
  \[ Z_2 = \sqrt{-2 \ln R_1} \sin(2\pi R_2) \]
Direct Transform for Normal Distributions

- Computation of \( \sqrt{\ln(R)} \) is CPU intensive, but is reused for \( Z_1 \) and \( Z_2 \)
- To obtain normal variates \( X_i \) with mean \( \mu \) and variance \( \sigma^2 \):
  \[
  X_i = \mu + \sigma Z_i 
  \]

Normal Distribution using Direct Transform

\[
\begin{align*}
N &:= 1000 \\
i &:= 0..N-1 \\
R_1 &:= \text{runif}(N, 0, 1) \\
R_2 &:= \text{runif}(N, 0, 1) \\
Z_1 &:= \sqrt{2 \ln(R_1)} \cos(2\pi R_2) \quad \mu := 2 \\
Z_2 &:= \sqrt{2 \ln(R_1)} \sin(2\pi R_2) \quad \sigma := 2.5 \quad \sigma^2 := 6.25 \\
X_1 &:= \mu + \sigma Z_1 \\
X_2 &:= \mu + \sigma Z_2 \\
M &:= 50 \\
h_1 &:= \text{histogram}(M, X_1) \\
h_2 &:= \text{histogram}(M, X_2) \\
k &:= 0..M-1 \\
h_{1,k,1} &:= \frac{h_{1,k,1}}{N} \\
h_{2,k,1} &:= \frac{h_{2,k,1}}{N} \\
\text{mean}(X_1) &:= 2.092 \\
\text{mean}(X_2) &:= 2.015 \\
\text{var}(X_1) &:= 6.584 \\
\text{var}(X_2) &:= 6.491 
\end{align*}
\]
Convolution Method

• Probability distribution of sum of independent R.V.'s is convolution of distributions of variables

• Erlang R.V. $X$ with parameters $(K, \theta)$ is sum of $K$ independent exponential R.V.s $X_i$ each with mean $1/K\theta$

$$X = \sum_{i=1}^{K} X_i$$

• From Inverse Transform Technique, each $X_i$ is generated with $1/\lambda=1/K\theta$

$$X = \sum_{i=1}^{K} -\frac{1}{K\theta} \ln R_i$$

$$= -\frac{1}{K\theta} \ln \left( \prod_{i=1}^{K} R_i \right)$$
Acceptance-Rejection Technique

- Use the Acceptance-Rejection Technique when other methods have no straightforward solution (e.g., no closed form solution)
- Efficiency depends on fraction of generated random numbers that are rejected
Using Acceptance-Rejection Technique for Poisson Distribution

• Poisson R.V., $N$, with mean $\alpha$ has a probability mass function

\[ p(n) = P(N = n) = \frac{e^{-\alpha} \alpha^n}{n!}, \quad n = 0, 1, 2... \]

• $N$ is the number of arrivals from Poisson arrival process in one unit of time
• Interarrival times $A_1, A_2, \ldots$ of successive customers are exponentially distributed with rate $\alpha$
• We know how to generate an exponential distribution from Inverse Transform Technique
• Relationship between the discrete Poisson process and continuous exponential distribution:

\[ N = n \]

if and only if

\[ A_1 + A_2 + \ldots + A_n \leq 1 < A_1 + \ldots + A_n + A_{n+1} \]

• Generate $n+1$ exponential interarrival times until some arrival occurs after $t=1$, then set $N=n$
Using Acceptance-Rejection Technique for Poisson Distribution

• Generate arrival times

\[ A_1 + A_2 + \ldots + A_n \leq 1 < A_1 + \ldots + A_n + A_{n+1} \]

or

\[ \sum_{i=1}^{n} -\frac{1}{\alpha} \ln R_i \leq 1 < \sum_{i=1}^{n+1} -\frac{1}{\alpha} \ln R_i \]

or

\[ \sum_{i=1}^{n} \ln R_i \geq -\alpha > \sum_{i=1}^{n+1} \ln R_i \]

or

\[ \ln \prod_{i=1}^{n} R_i \geq -\alpha > \ln \prod_{i=1}^{n+1} R_i \]

or

\[ \prod_{i=1}^{n} R_i \geq e^{-\alpha} > \prod_{i=1}^{n+1} R_i \]
Using Acceptance-Rejection Technique for Poisson Distribution

- Given $\alpha$:

  Set $n=0$, $P=1$

  Generate $R_{n+1}$, 
  $P *= R_{n+1}$

  Reject $n$: increment $n$

  $P<e^{-\alpha}$?
  
  Accept $n$: $N = n$
Using Acceptance-Rejection Technique for Poisson Distribution

• Example 8.11: Bus with Poisson arrival process, arrival rate $\alpha=4$ per hour. Generate number of buses arriving during a 1 hour period

• With high acceptance rate, this technique works well. As acceptance rate drops (e.g., for larger $\alpha$), this technique becomes inefficient.

• An approximation for the Poisson process for $\alpha>15$ is:

$$Z = \sqrt{-2 \ln R_1 \cos(2\pi R_2)}$$

$$N = \begin{cases} 
0 & \text{if } \alpha + \sqrt{\alpha} Z - .5 < 0 \\
\text{ceil}(\alpha + \sqrt{\alpha} Z - .5) & \text{otherwise}
\end{cases}$$

\[
\begin{align*}
\alpha_{bus} & := 4 \\
e^{-\alpha_{bus}} & = 0.018 \\
N(\alpha) & := \\
& \begin{cases} 
n \leftarrow 0 \\
P \leftarrow 1 \\
m \leftarrow e^{-\alpha} \\
\text{while } P \geq m \\
P \leftarrow P \cdot \text{rnd}(1) \\
n \leftarrow n + 1 \\
\text{return } n
\end{cases}
\end{align*}
\]

How many arrivals per hour during a given 8 hour period

\[
i := 0..7
\]

\[
\text{Arrivals}_i := N(\alpha_{bus})
\]

\[
\begin{array}{c}
3 \\
4 \\
7 \\
\end{array}
\]

\[
\begin{array}{c}
3 \\
4 \\
2 \\
4 \\
3 \\
\end{array}
\]

mean(Arrivals) = 3.75
Homework #7

• Generate 500 random numbers between 0 and 1, based on the linear congruential techniques covered last week. *Print no more than 50 of them.* Use a modulus size of at least 500. Find the number of up-runs and down-runs in the sequence. Using the run-test method discussed, see if the sequence generated appears independent with a 10% level of significance.

• Use the built-in random number generator in whatever programming environment you are familiar with (C, Matlab, Excel, Mathcad, etc.). Generate as many random numbers as you can in a reasonable period of time (e.g., 5-10 minutes of computer execution time) and apply at least one of the tests for randomness we have discussed. *(limit your homework submission for this problem to 2-3 pages at most. I don’t want to have to print 50 pages to give to the grader).* Extra credit for more than one randomness test.

• Extra credit: If you have access to a different type of computing environment, try the same experiment with the same method and compare your results. For instance, if you use Excel, you could use the same spreadsheet on a PC vs. a Macintosh vs. StarOffice with linux. If you use Matlab or C, you could try the same program on Sun/UNIX, SGI/Irix, PC/Windows, PC/linux. Different x86 PCs with different versions of Windows doesn’t count.