Random Processes

Probability and hypothesis testing

ECE 111 Introduction to ECE

Jake Glower - Week #15

Please visit Bison Academy for corresponding
lecture notes, homework sets, and solutions

Random Processes

Every time you do an experiment, you will get slightly different results

- The sum of 10 6-sided dice is different each roll
- The gain of each transistor you measure is slightly different

These are random processes

Statistics is a branch of mathematics geared to random processes

- It allows you to analyze your lab data

This lecture covers common statistical tests used to analyze such data.

- Chi-Squred Test: Is a die really fair?
- Monte-Carlo Simulations: Collect *lots* of data to determine probabilities
- Normal Approximation: Calculate probabilities when the mean and variance is known
- t-Test: Calculate probabilities when the mean and variance is estimated

Chi-Squared Test

Test of a distribution

- If a die fair? (all numbers have equal probability)
- Was data fudged? (data is too prefect)

Procedure

- You collect a bunch of data
- Separate the data in to N bins (such as 6 numbers for testing a 6-sided die).
- Count the number of times the data wound up in each bin
- Compare it to the expected frequency using the metric

$$\chi^2 = \sum \left(\frac{(np_i - N_i)^2}{np_i} \right)$$

- Use a chi-squared table to convert this to a probability.

Chi-Squared Table

- df is the degrees of freedom (number of bins minus 1)
- % is the probability level
- The number in the table is the chi-square value

Chi-Squared Table

Probability of rejecting the null hypothesis
http://people.richland.edu/james/lecture/m170/tbl-chi.html

df	99.5%	99%	97.5%	95%	90%	10%	5%	2.5%	1%	0.5%
1	7.88	6.64	5.02	3.84	2.71	0.02	0	0	0	0
2	10.6	9.21	7.38	5.99	4.61	0.21	0.1	0.05	0.02	0.01
3	12.84	11.35	9.35	7.82	6.25	0.58	0.35	0.22	0.12	0.07
4	14.86	13.28	11.14	9.49	7.78	1.06	0.71	0.48	0.3	0.21
5	16.75	15.09	12.83	11.07	9.24	1.61	1.15	0.83	0.55	0.41
6	18.55	16.81	14.45	12.59	10.65	2.2	1.64	1.24	0.87	0.68
7	20.28	18.48	16.01	14.07	12.02	2.83	2.17	1.69	1.24	0.99
8	21.96	20.09	17.54	15.51	13.36	3.49	2.73	2.18	1.65	1.34
9	23.59	21.67	19.02	16.92	14.68	4.17	3.33	2.7	2.09	1.74
10	25.19	23.21	20.48	18.31	15.99	4.87	3.94	3.25	2.56	2.16

Example: Fair Die:

- In Matlab's random number generator truly random?
- Does it favor certain numbers?

Step 1: Set up an experiment where we roll a 6-sided die 120 times.

```
result = zeros(6,1);
for i=1:120
   D6 = ceil( 6 * rand );
   result(D6) = result(D6) + 1;
   end

result

chi2 = sum(  (result - 20).^2 / 20  )
```

Step 2: Compute the chi-squred score

$$\chi^2 = 2.4000$$

Number	probability (p)	Expected Frequency (np)	Actual Frequency (N)	$\chi^2 = \frac{(np-N)^2}{np}$
1	1/6	20	22	0.2000
2	1/6	20	17	0.4500
3	1/6	20	17	0.4500
4	1/6	20	19	0.0500
5	1/6	20	25	1.2500
6	1/6	20	20	0.0000
			Sum	**2.4**

Step 3: Convert to a probability

- Use a chi-squared table (5 degrees of freedom, $\chi^2 = 2.40$)
- $10\% < p < 90\%$

Chi-Squared Table

Probability of rejecting the null hypothesis
http://people.richland.edu/james/lecture/m170/tbl-chi.html

df	99.5%	99%	97.5%	95%	90%	10%	5%	2.5%	1%	0.5%
1	7.88	6.64	5.02	3.84	2.71	0.02	0	0	0	0
2	10.6	9.21	7.38	5.99	4.61	0.21	0.1	0.05	0.02	0.01
3	12.84	11.35	9.35	7.82	6.25	0.58	0.35	0.22	0.12	0.07
4	14.86	13.28	11.14	9.49	7.78	1.06	0.71	0.48	0.3	0.21
5	16.75	15.09	12.83	11.07	9.24	1.61	1.15	0.83	0.55	0.41
6	18.55	16.81	14.45	12.59	10.65	2.2	1.64	1.24	0.87	0.68
7	20.28	18.48	16.01	14.07	12.02	2.83	2.17	1.69	1.24	0.99
8	21.96	20.09	17.54	15.51	13.36	3.49	2.73	2.18	1.65	1.34
9	23.59	21.67	19.02	16.92	14.68	4.17	3.33	2.7	2.09	1.74
10	25.19	23.21	20.48	18.31	15.99	4.87	3.94	3.25	2.56	2.16

Step 3: Use StatTrek.com

- p = 0.21
- *Based upon this data, there is a 21% chance that the die is loaded*

- Enter a value for degrees of freedom.
- Enter a value for one, and only one, of the remaining unshaded text boxes.
- Click the **Calculate** button to compute values for the other text boxes.

Degrees of freedom	5
Chi-square critical value (CV)	2.40
$P(X^2 < 2.40)$	0.21
$P(X^2 > 2.40)$	0.79

Chi-Sqared Result from StatTrek.com.

Example 2: Fair Die, Non-Random Process

Instead of using the *rand* function, generate the die roll by going through the sequence {1, 2, 3, 4, 5, 6} and repeating. Is this a fair die?

```
result = zeros(6,1);
for i=1:120
   D6 = mod(i, 6) + 1;
   result(D6) = result(D6) + 1;
   end

result

chi2 = sum(  (result - 20).^2 / 20  )
```

Now set up a table where we compare the actual frequency of each number (N) vs. the expected results (np):

Fair die, non-random process

- The chi-squared score of 0.0000 tells you this is a deterministic proces (not random)

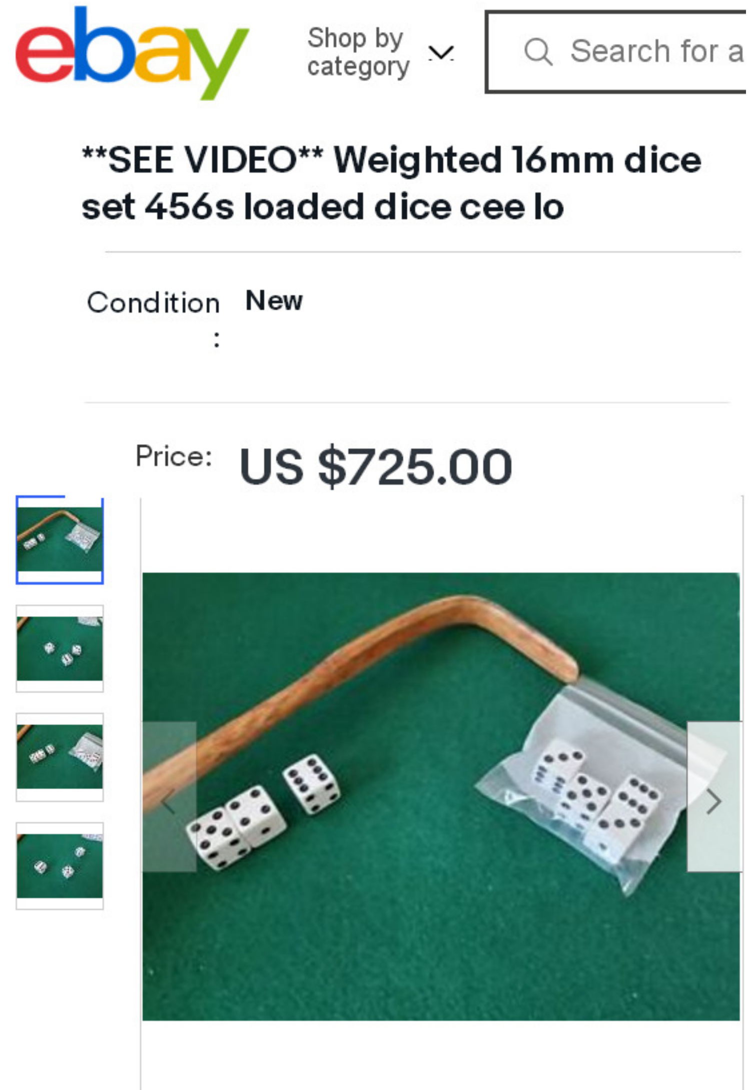
Number	probability (p)	Expected Frequency (np)	Actual Frequency (N)	$\frac{(np-N)^2}{np}$
1	1/6	20	20	0
2	1/6	20	20	0
3	1/6	20	20	0
4	1/6	20	20	0
5	1/6	20	20	0
6	1/6	20	20	0
			Sum	**0.000**

The chi-squared value for going through the sequence {1,,6} is 0.000

Example3: Loaded Die:

Suppose 10% of the time you cheat: the die is forced to be a six. Can you detect this?

```
result = zeros(6,1);
for i=1:120
   D6 = ceil( 6 * rand );
   if (rand < 0.1)
      D6 = 6;
      end
   result(D6) = result(D6) + 1;
   end

result

chi = sum(  (result - 20).^2 / 20 )
```



Loaded Die (cont'd)

- Again, set up a chi-squared table:
- From StatTrek, p = 0.48 (48% chance the die is loaded)
- From the Matlab code, we *know* the die is loaded.
- From the data, I can't tell with only 120 data points.

Number	probability (p)	Expected Frequency (np)	Actual Frequency (N)	$\frac{(np-N)^2}{np}$
1	1/6	20	17	0.4500
2	1/6	20	21	0.0500
3	1/6	20	23	0.4500
4	1/6	20	18	0.2000
5	1/6	20	15	1.2500
6	1/6	20	26	1.8000
			Sum	**4.2000**

Loaded Die with 600 samples:

- $\chi^2 = 19.34$
- p = 0.998 (StatTrek)
- With enough data, I can tell that the die is loaded

Number	probability (p)	Expected Frequency (np)	Actual Frequency (N)	$\frac{(np-N)^2}{np}$
1	1/6	100	95	0.2500
2	1/6	100	85	2.2500
3	1/6	100	91	0.8100
4	1/6	100	91	0.8100
5	1/6	100	99	0.0100
6	1/6	100	139	15.2100
			Sum	**19.3400**

How Loaded is Too Loaded?

How loaded can you make a die so that you won't be detected with 60 die rolls?

- Assume "detect" means p > 0.9 (Chi-Squared = 9.24)
- Assume x too many sixes, x/5 too few ones, twos, etc.
- $0.12x^2 = 9.24 \quad \Rightarrow \quad x = 8.77 \quad p = \frac{8.77}{60} = 14.6\%$
- You can load the die so that a six comes up 14.6% of the time (fair otherwise)

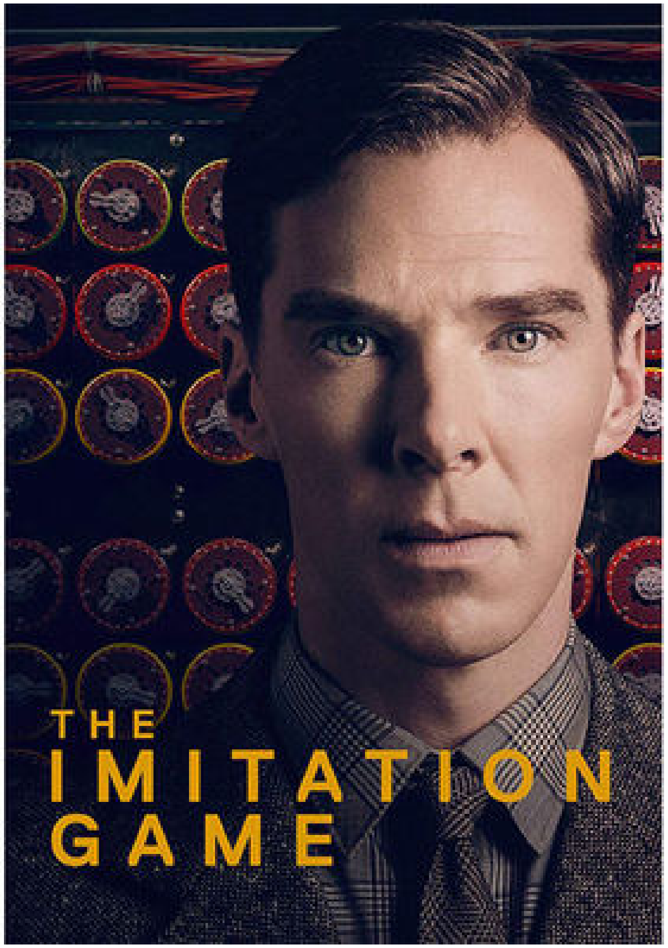
Number	probability (p)	Expected Frequency (np)	Actual Frequency (N)	$\frac{(np-N)^2}{np}$
1	1/6	10	10 - x/5	$0.004\ x^2$
2	1/6	10	10 - x/5	$0.004\ x^2$
3	1/6	10	10 - x/5	$0.004\ x^2$
4	1/6	10	10 - x/5	$0.004\ x^2$
5	1/6	10	10 - x/5	$0.004\ x^2$
6	1/6	10	10 + x	$0.1\ x^2$
			Sum	$0.12\ x^2$

The Imitation Game:

- The British broke the German code, Enigma
- Instead of reading and responding to *every* communications, statistics were used to determine how many communications the British could respond to without the Germans knowing that their code was broken.

This is a Chi-Squared test

- Same problem as "How loaded can you make a die without being detected?"



Example 4: Fudging the Data:

Chi-Squared tests can also detect data which is *too* good

- Task: Roll a die 600 times
- Actual: Roll the die 60 times and add 90 to the result

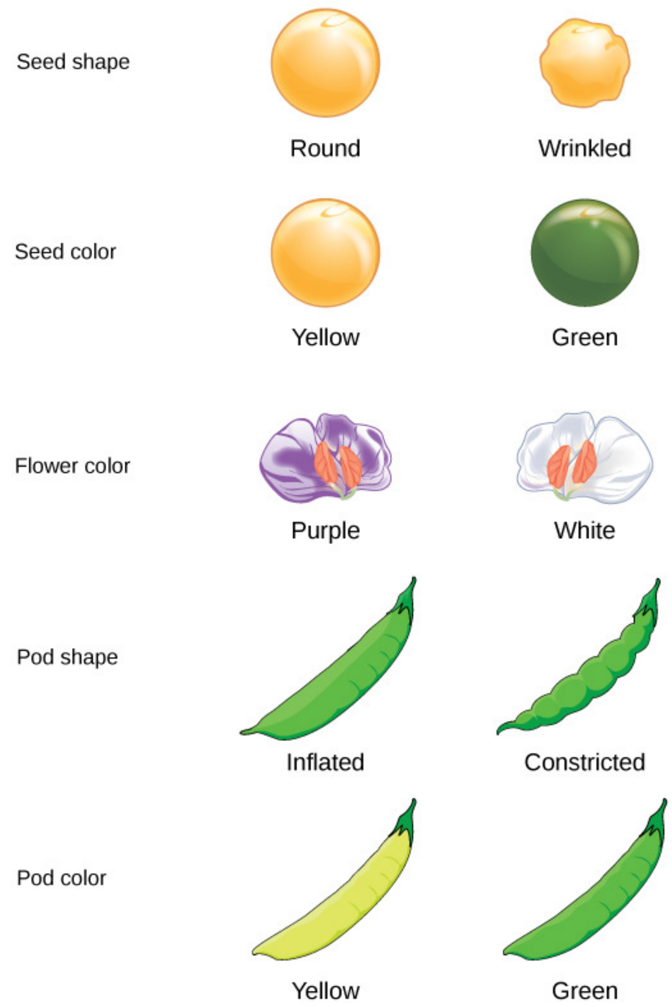
```
result = zeros(6,1);
for i=1:60
   D6 = ceil( 6*rand);
   result(D6) = result(D6) + 1;
   end

result = result + 90

chi2 = sum(  (result - 100).^2 / 100  )
```

Result: Fudging the data:

- $\chi^2 = 0.12$
- probability = 0.0003 (StatTrek)
- Odds against getting such good data is 3333 : 1

Number	probability (p)	Expected (np)	Actual (N)	$\frac{(np-N)^2}{np}$
1	1/6	100	99	0.01
2	1/6	100	101	0.01
3	1/6	100	99	0.01
4	1/6	100	101	0.01
5	1/6	100	102	0.04
6	1/6	100	98	0.04
			Sum	**0.12**

Note: Mendel's experiments with peas and genetics were *too* good. Chi-squared tests show that Mendel fudged his data.



Handout

1) A 6-sided die is rolled 90 times.

- Use a chi-squared test to determine probability that the die is not fair

Number	probability (p)	Expected (np)	Actual (N)	$\chi^2 = \frac{(np-N)^2}{np}$
1	1/6	15	12	
2	1/6	15	10	
3	1/6	15	13	
4	1/6	15	13	
5	1/6	15	25	
6	1/6	15	17	
			Sum	

New Topic:

- Monte-Carlo Simulation
- Normal Distribution,
- Student-t Distribution

Problem: What does my product look like?

- Due to manufacturing tolerance, no two devices off an assembly line are exactly alike.
- Example: The gain of 45 6144 NPN transistors:

```
374, 370, 359, 370, 351, 357, 352, 364, 372, 367, 349, 372, 344, 345, 348,
345, 341, 344, 345, 346, 356, 346, 344, 348, 370, 358, 365, 365, 342, 365,
367, 343, 361, 344, 367, 343, 356, 342, 347, 347, 346, 357, 356, 340, 343
```

Typical Things I'd Like to Know:

- What is the probability that the gain is more then 300?
- What range contains 90% of the gains?
- (What is the 90% confidence interval?)

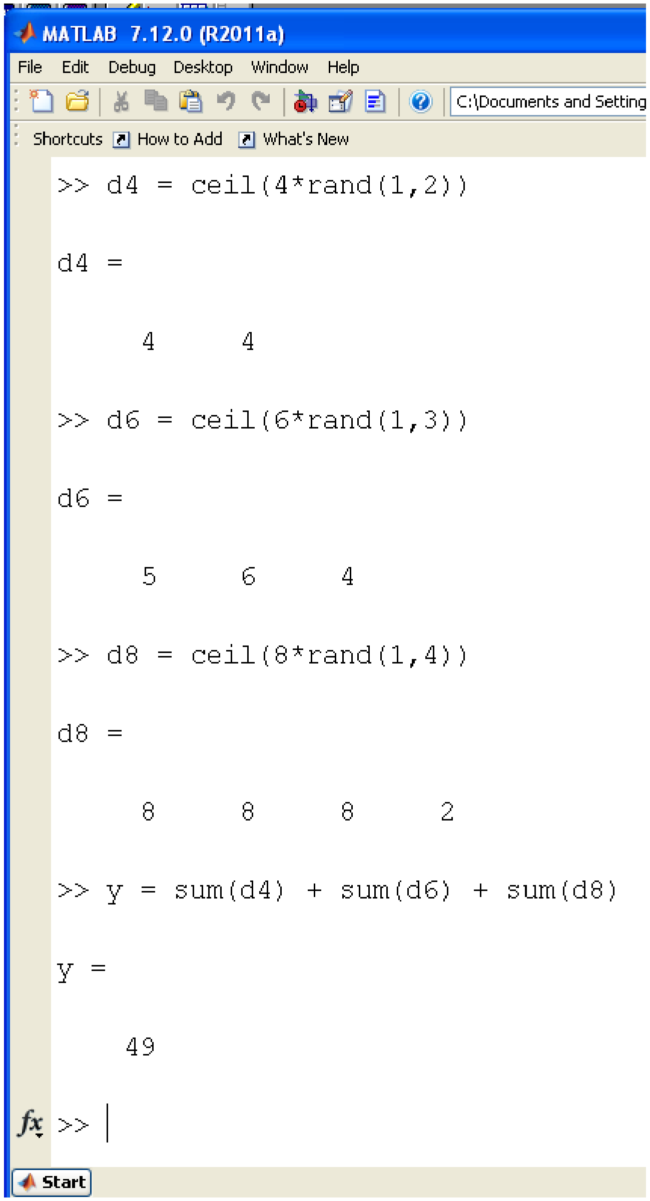
Example: Dice

Let

$$Y = 2d4 + 3d6 + 4d8$$

Determine:

- The probability that $y > 39.5$
- The lower 5% tail: $p(Y < a) = 0.05$
- The upper 5% tail: $p(Y > b) = 0.05$
- The 90% confidence interval
  $(a < Y < b)$ with a probability of 0.9



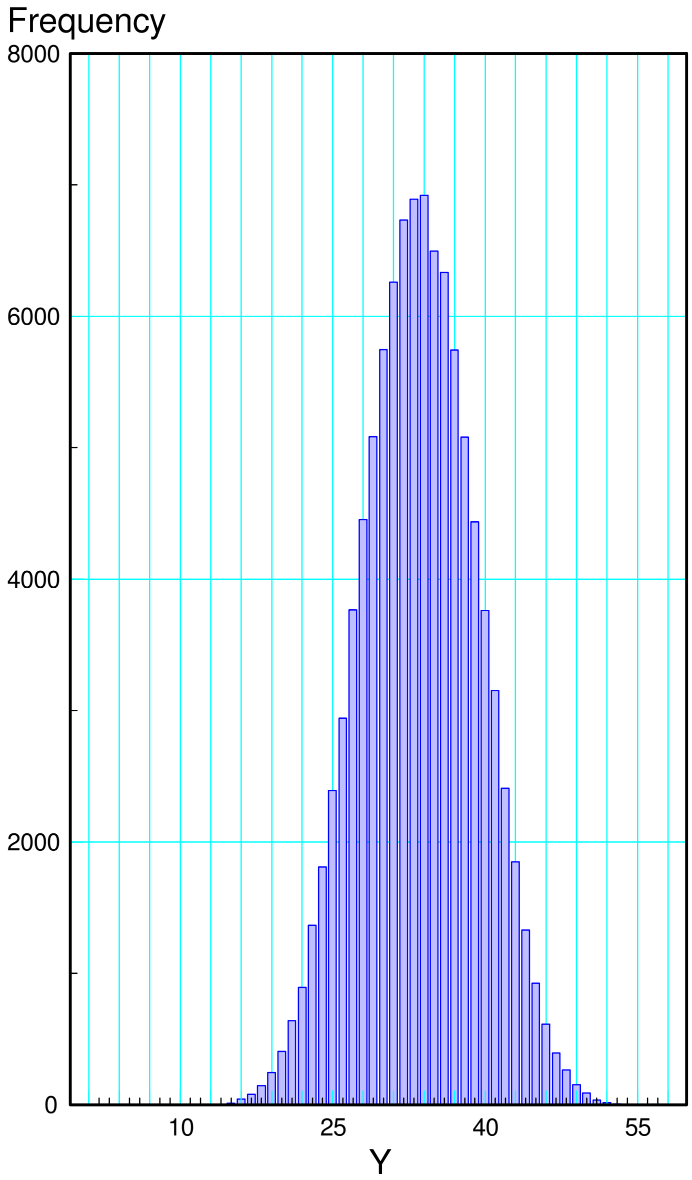
Solution #1: Monte-Carlo Simulation

Roll the dice 100,000 times

Record the frequency of each result

Matlab Code

```
RESULT = zeros(58,1);
for n=1:1e5
   d4 = ceil(4*rand(1,2));
   d6 = ceil(6*rand(1,3));
   d8 = ceil(8*rand(1,4));
   Y = sum(d4) + sum(d6) + sum(d8);
   RESULT(Y) = RESULT(Y) + 1;
   end
bar(RESULT)
```



Find p(y > 39.5)

- Count the number of times you rolled 40 or higher
- Divided by the sample size (100,000)
- **p = 14.72%**

Find x such that p(y < a) = 5%

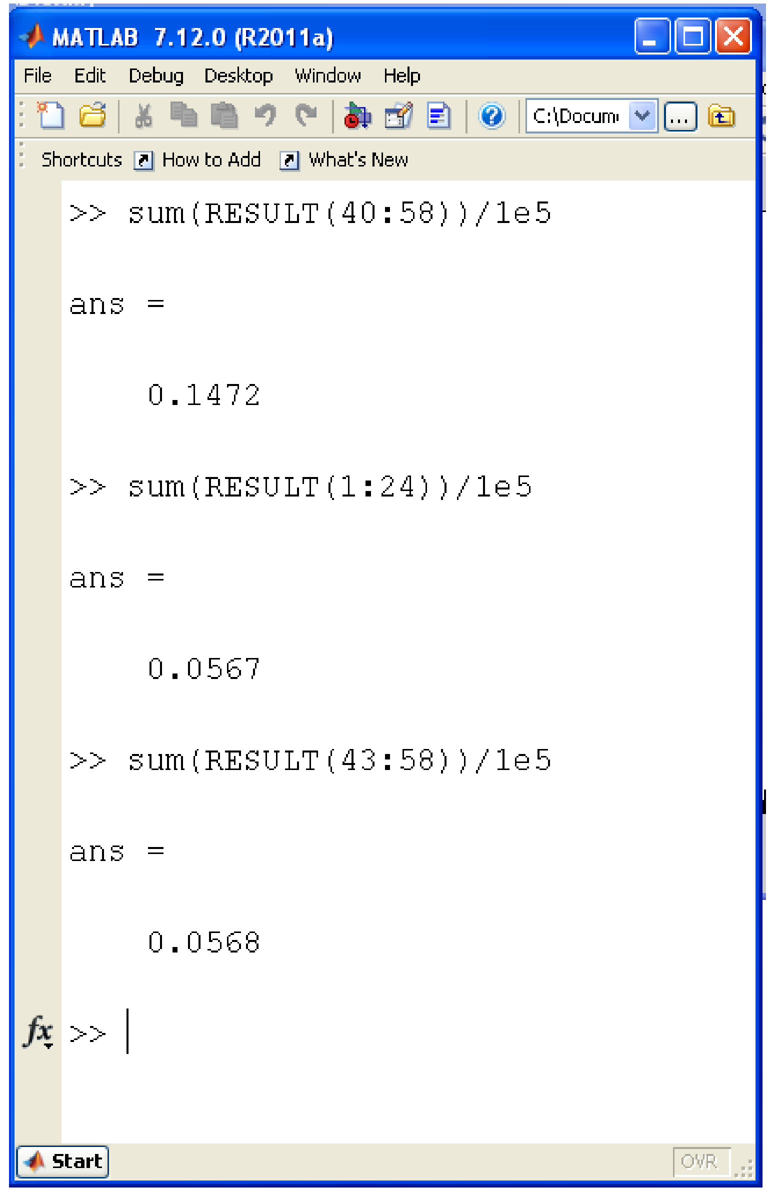
- Left 5% tail
- Find a such that 5% of the rolls are less than a
- a = 24

Find x sigh that p(y > b) = 5%

- Right 5% tail
- Find b such that 5% of the rolls are more than b
- b = 43

The 90% confidence interval is then

- **24.5 < y < 42.5   p = 0.9**



Problem with Monte-Carlo Simulations

Expense

- If it costs $10 for each experiment, 100,000 samples = $1 million

Time

- If it takes 10 minutes to measure each y, 100,000 samples = 1.9 years

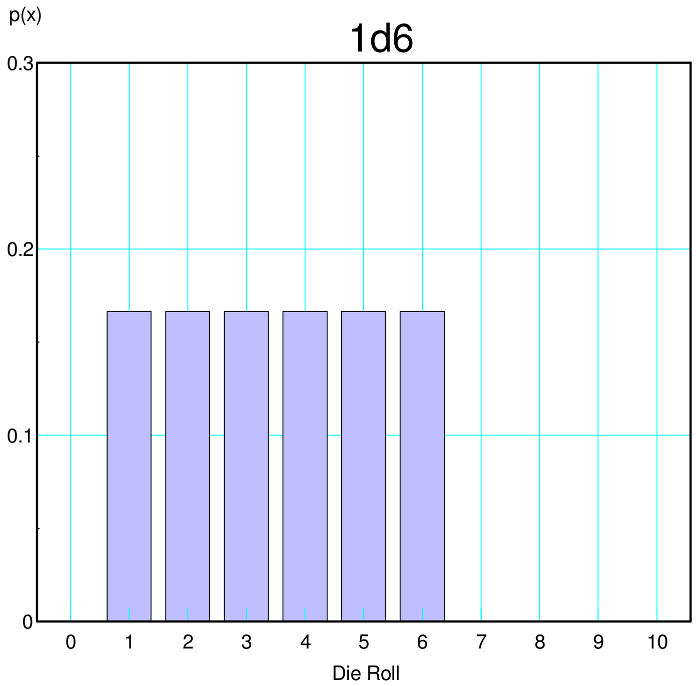
Can you come up with the same results using fewer measurements?

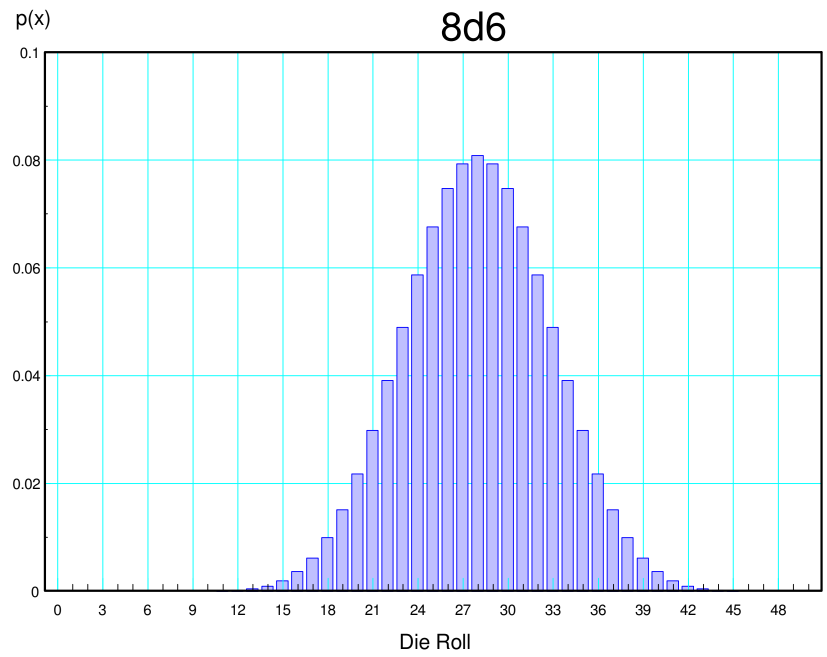
- Yes
- Requires statistics

Central Limit Theorem

- All distributions converge to a normal distribution
- Normal + Normal = Normal
- Once you have a normal distribution, you remain with a normal distribution.

For example, 1d6 (not normal) vs. 8d6 (approx normal)

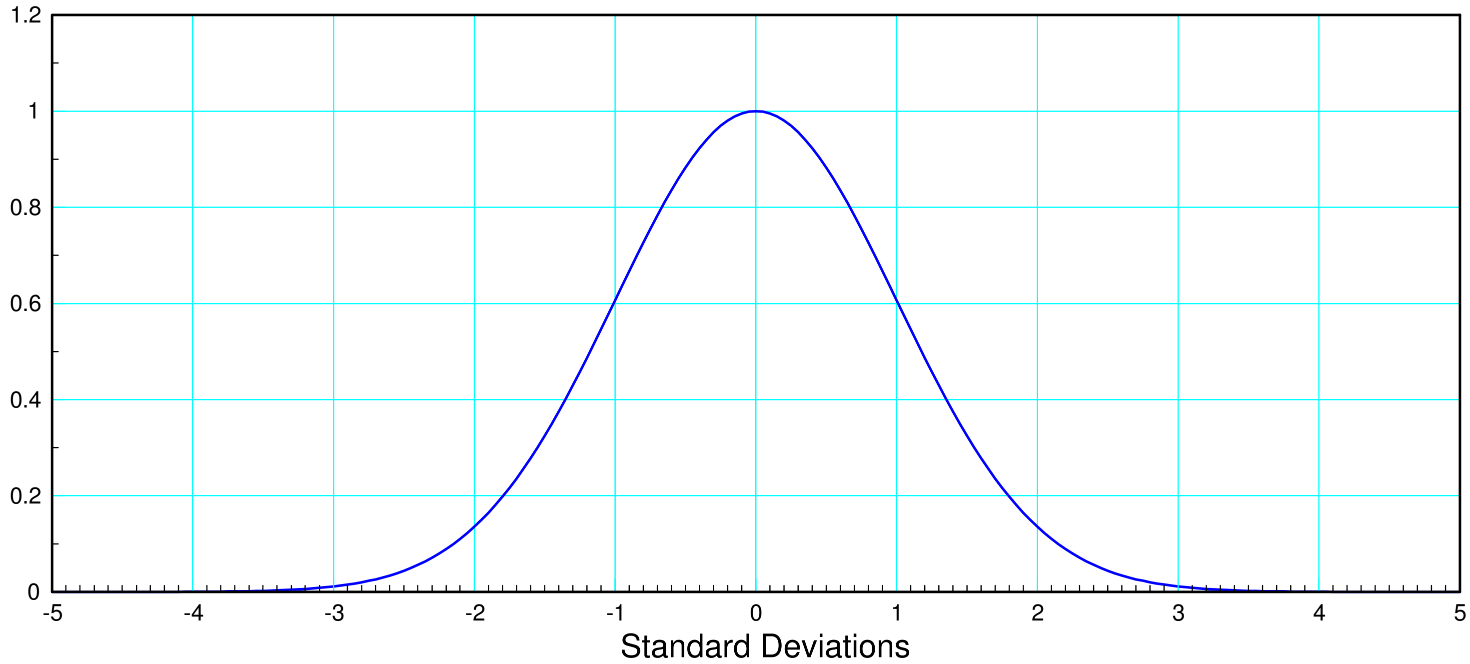




Normal (Gaussian) Distributions:

$$N(\bar{x}, \sigma^2)$$

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left(\frac{-(x-\bar{x})^2}{2\sigma^2}\right)$$



Standard Normal Distribution (normalized so the peak is 1.000)

Properties of Normal Distributions

Two parameters define a normal distribution

- Mean
- Standard Deviation (or Variance)

Mean: average of the data

$$\mu = \frac{1}{n} \sum y_i$$

Variance: average squared distance to mean

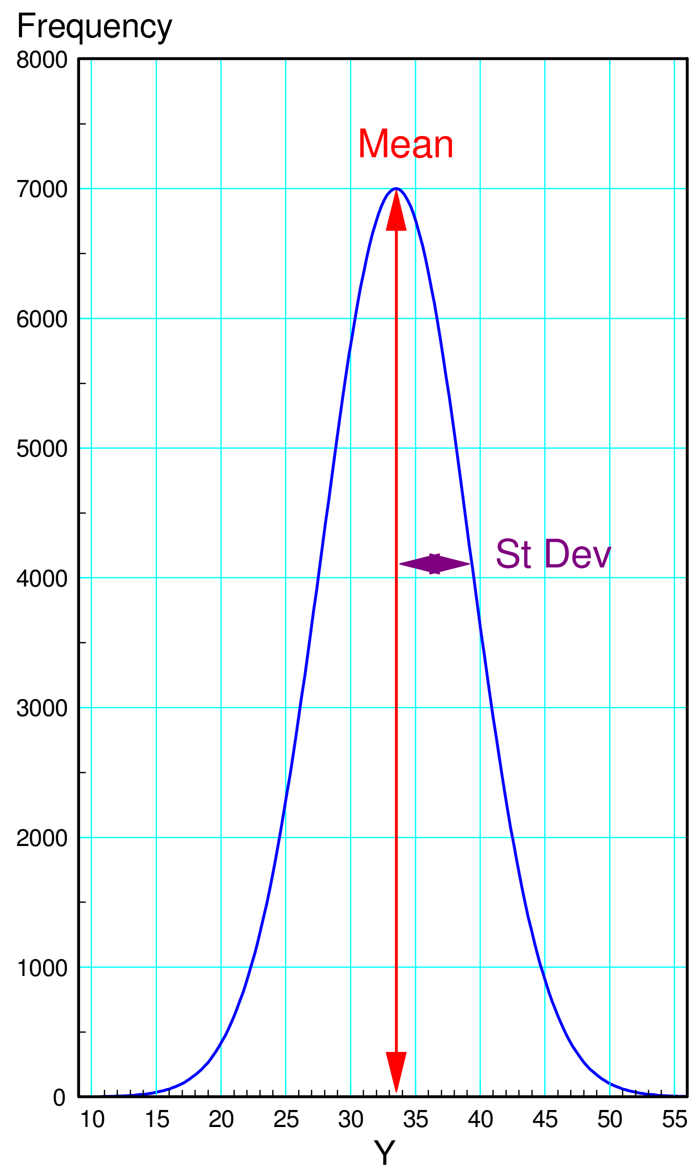
$$\sigma^2 = \frac{1}{n} \sum (y_i - \mu)^2$$

Standard Deviation (spread)

$$\sigma = \sqrt{\sigma^2}$$

When you add normal distributions

- The means add
- The variances add



Example: Dice

The mean and variance for a 4-sided die

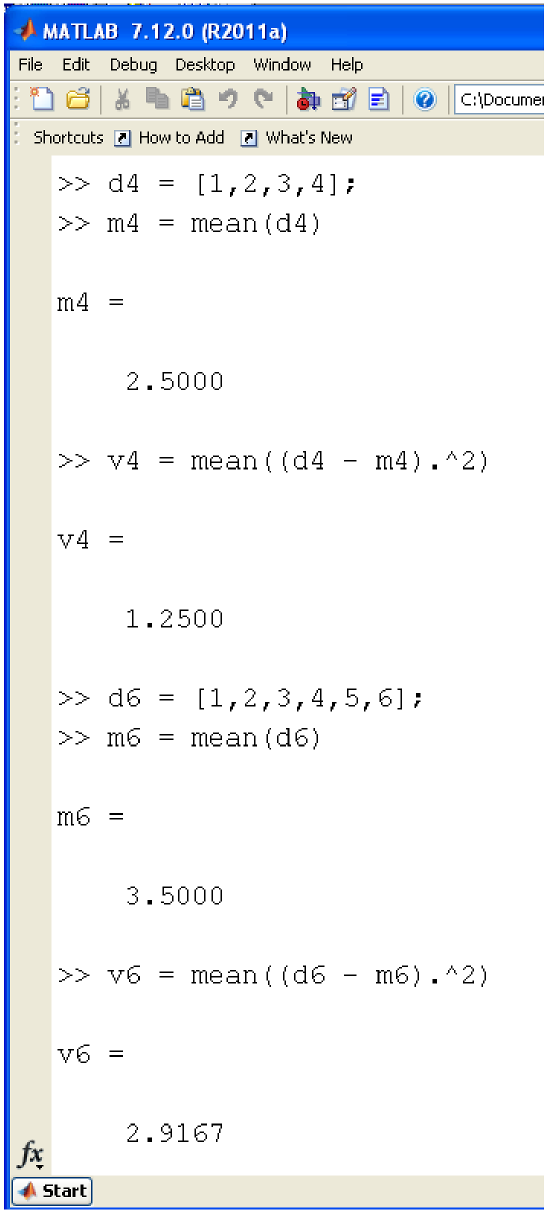
```
>> d4 = [1,2,3,4];
>> m4 = sum(d4) / 4
       m4 =    2.5000
>> v4 = sum( (d4 - m4).^2 )/4
       v4 =    1.2500
```

The mean and variance for a 6-sided die

```
>> d6 = [1,2,3,4,5,6];
>> m6 = sum(d6) / 6
       m6 =    3.5000
>> v6 = sum( (d6 - m6).^2 )/6
       v6 =    2.9167
```

The mean and variance for an 8-sided die

```
>> m8 = sum(d8) / 8
m8 =    4.5000
>> v8 = sum( (d8 - m8).^2 )/8
v8 =    5.2500
```



2d4 + 3d6 + 4d8

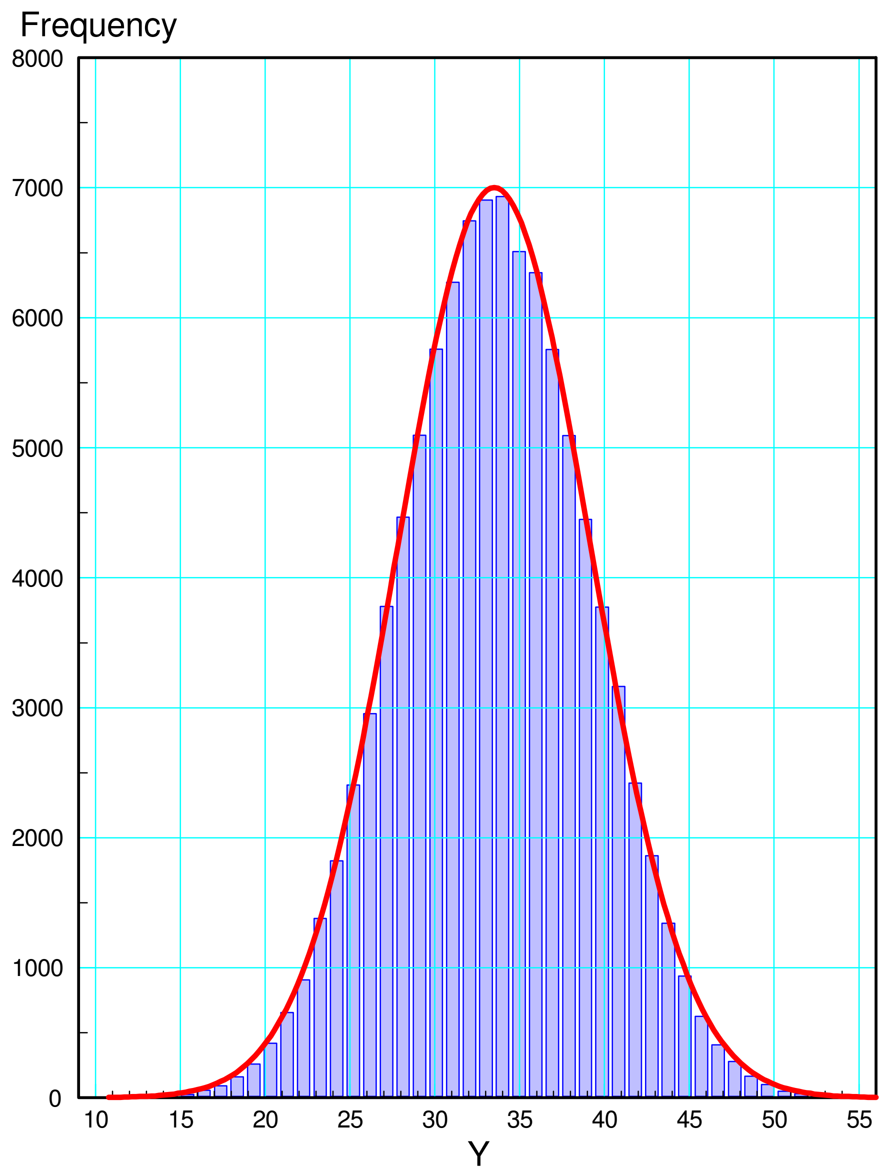
- The means add
- The variances add

```
>> my = 2*m4 + 3*m6 + 4*m8
my =   33.5000

>> vy = 2*v4 + 3*v6 + 4*v8
vy =   32.2500

>> sy = sqrt(vy)
sy =    5.6789
```

To plot the normal pdf

```
s = [-4:0.01:4]';
p = exp(-s.^2 / 2);
plot(s*sy+my,p*7000);
xlabel('Die Roll')
```

Central Limit Theorem in action...



What is the probability of y > 39.5?

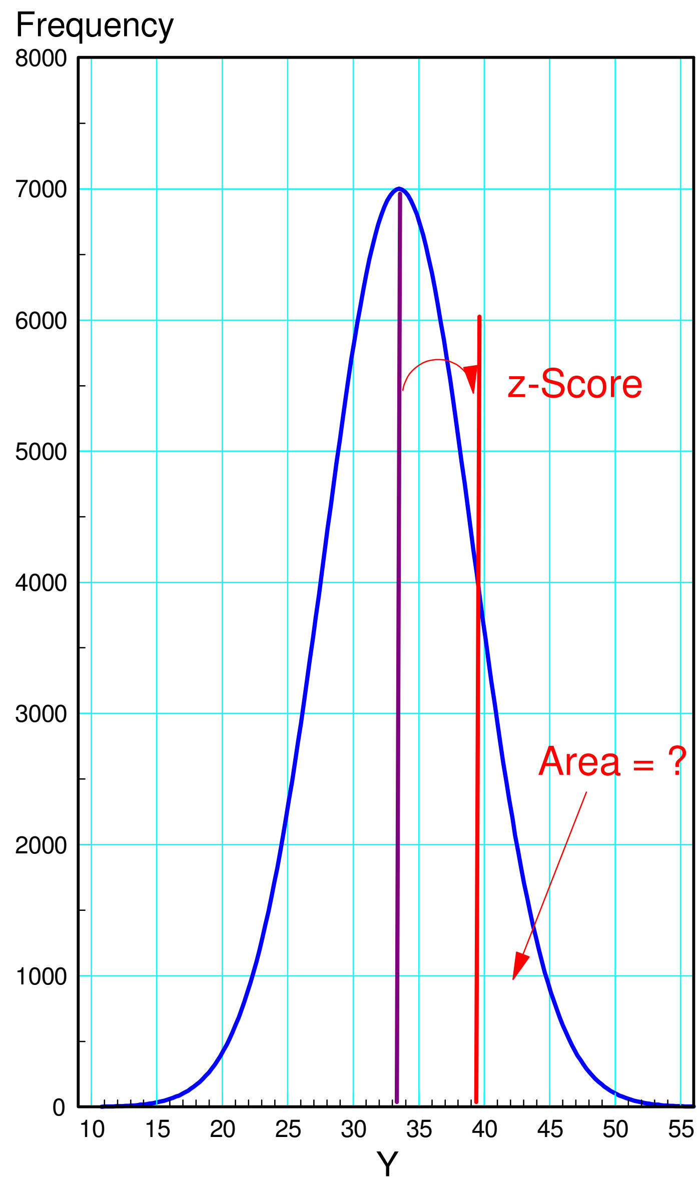
- This is the area to the rignt of 39.5

Calculate how far y is from the mean

$$z = \left(\frac{39.5 - \mu}{\sigma}\right)$$

```
>> z = (39.5 - my) / sy
z =    1.0565
```

- p(y) is the area to the right of z
- Can be found using StatTrek
- Can be found using a standard normal table



StatTrek

- www.StatTrek.com
- z-score = 1.0565

Input the z-score

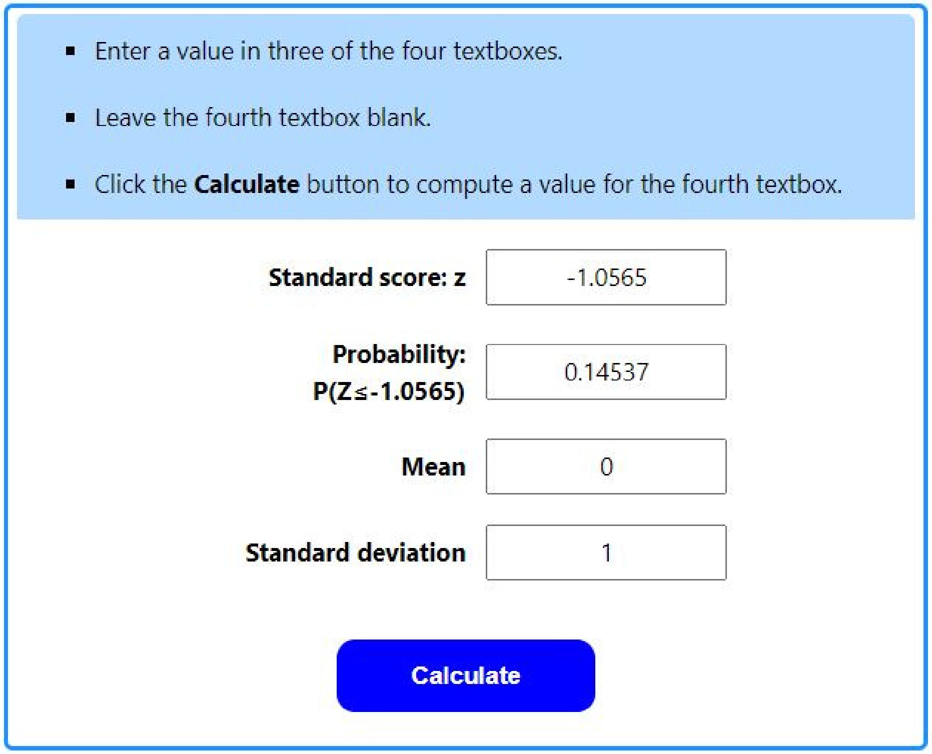
- negative value gives the area of the tail
- positive value gives the area without the tail

p = 14.573%

- vs. 14.72% with Monte Carlo

Note:

- This result is almost the same
- It required **zero** die tosses



Standard Normal Table

- z-score = 1.0565
- p = 14.573%

	0	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09
0.0	0.50000	0.49601	0.49202	0.48803	0.48405	0.48006	0.47608	0.47209	0.46812	0.46415
0.1	0.46017	0.45620	0.45224	0.44829	0.44433	0.44038	0.43644	0.43251	0.42858	0.42465
0.2	0.42074	0.41684	0.41294	0.40904	0.40516	0.40130	0.39743	0.39358	0.38974	0.38591
0.3	0.38209	0.37828	0.37448	0.37070	0.36693	0.36317	0.35942	0.35569	0.35197	0.34827
0.4	0.34458	0.34090	0.33724	0.33360	0.32997	0.32635	0.32276	0.31918	0.31561	0.31207
0.5	0.30854	0.30503	0.30153	0.29805	0.29460	0.29116	0.28774	0.28434	0.28096	0.27760
0.6	0.27425	0.27093	0.26763	0.26435	0.26109	0.25784	0.25463	0.25143	0.24825	0.24510
0.7	0.24196	0.23885	0.23576	0.23269	0.22965	0.22663	0.22363	0.22065	0.21769	0.21476
0.8	0.21186	0.20897	0.20611	0.20327	0.20046	0.19766	0.19489	0.19215	0.18943	0.18673
0.9	0.18406	0.18141	0.17879	0.17619	0.17361	0.17105	0.16853	0.16602	0.16354	0.16109
1.0	0.15865	0.15625	0.15387	0.15150	0.14917	0.14686	0.14457	0.14231	0.14007	0.13786
1.1	0.13567	0.13350	0.13136	0.12924	0.12714	0.12507	0.12302	0.12100	0.11900	0.11702
1.2	0.11507	0.11314	0.11123	0.10935	0.10749	0.10565	0.10384	0.10204	0.10027	0.09852
1.3	0.09680	0.09510	0.09342	0.09176	0.09012	0.08851	0.08691	0.08534	0.08379	0.08226
1.4	0.08076	0.07927	0.07780	0.07636	0.07493	0.07353	0.07214	0.07078	0.06944	0.06811
1.5	0.06681	0.06552	0.06426	0.06301	0.06178	0.06057	0.05938	0.05821	0.05705	0.05592

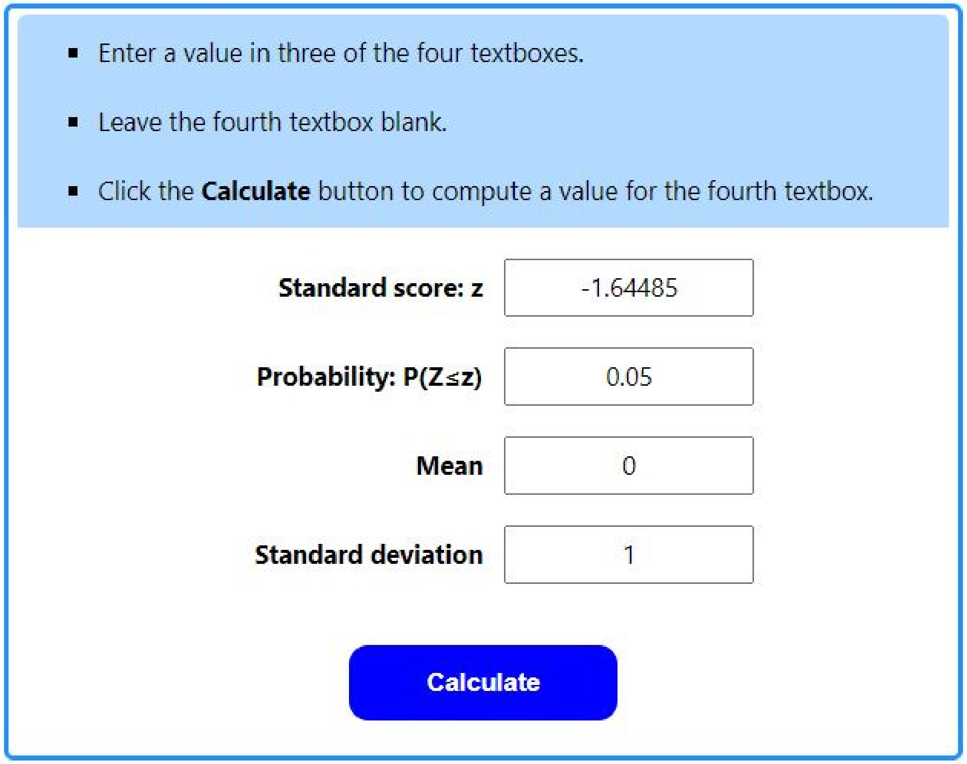
What is the 90% confidence interval?

- Two tails, each 5%

What is the z-score for 5% tails?

- z-score = 1.64485
- StatTrek works
- Stadard Normal table works

	0.03	0.04	0.05	0.06
1.5	0.06301	0.06178	0.06057	0.05938
1.6	0.05155	0.05050	0.04947	0.04846
1.7	0.04181	0.04093	0.04006	0.03920



90% Confidence Interval (cont'd)

- 5% Tails: z = 1.64485

Go 1.64485 standard deviations from the mean

- Each tail has an area of 5%
- Area in the middle is 90%

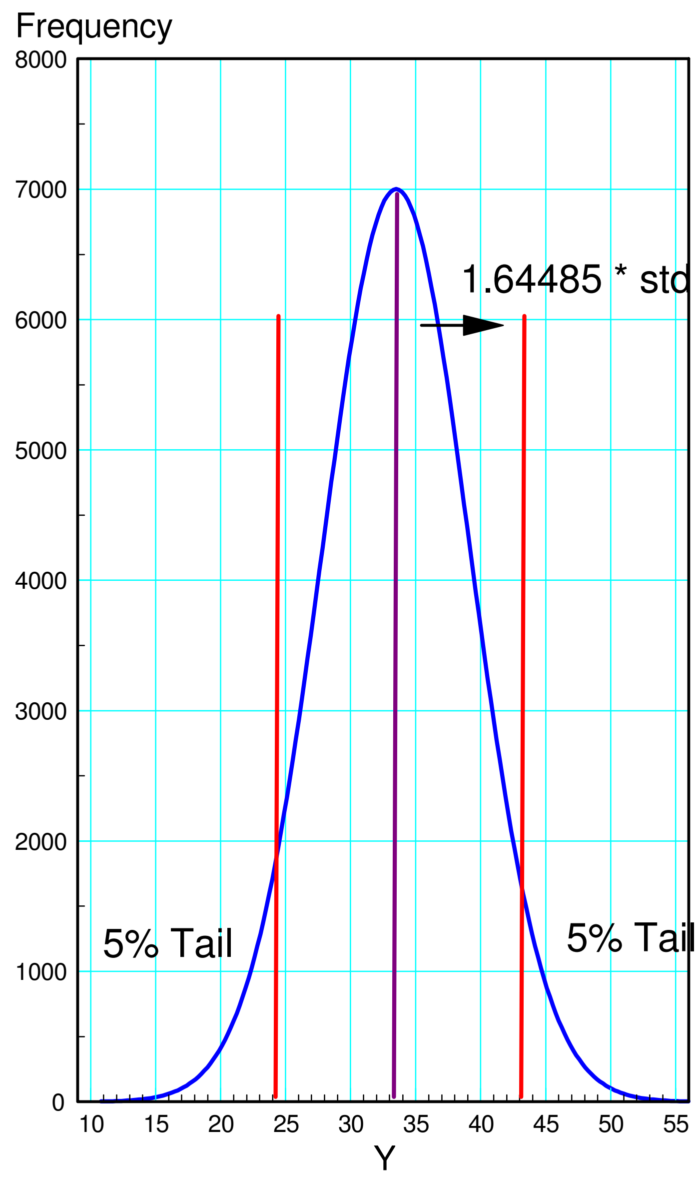
$$\mu - 1.64485\sigma < y < \mu + 1.64485y$$

$$24.159 < y < 42.841 \qquad p = 0.9$$

24.5 < y < 42.5 Monte-Carlo

Note

- Similar result to Monte Carlo
- Requires no die rolls



Summary

If you know the mean and the standard deviation, you can calculate the odds

- Find the z-score
- Distance to the mean in terms of standard deviations

Convert z-scores to probabilities

- Standard-Normal Table
- StatTrek

This requires zero die rolls

- Saving a lot of money
- Saving a lot of time

Problem: What if you *don't* know the mean and standard deviation?

Solution:

- Collect some data
- Estimate the mean and standard devitation from the data

The result is a Student-t Distribution

- Very similar to a Normal distribution
- Takes sample size into account

Example: y = 2d4 + 3d6 + 4d8

Step 1: Collect Data

Roll the dice ten times

```
DATA = [];
for i=1:10
   d4 = ceil( 4*rand(2,1) );
   d6 = ceil( 6*rand(3,1) );
   d8 = ceil( 8*rand(4,1) );
   Y = sum(d4) + sum(d6) + sum(d8);
   DATA = [DATA, Y];
   end
```

DATA = 31 35 38 32 42 33 39 34 32 24

Step 2: Compute the

- Mean
- Standard Deviation
- Sample Size

```
x = mean(DATA)
s = std(DATA)
n = length(DATA)
```

```
x =    34              mean
s =    4.9889          standard deviation
n =    10              sample size
```

Probability y > 39.5

Calculate the distance to the mean

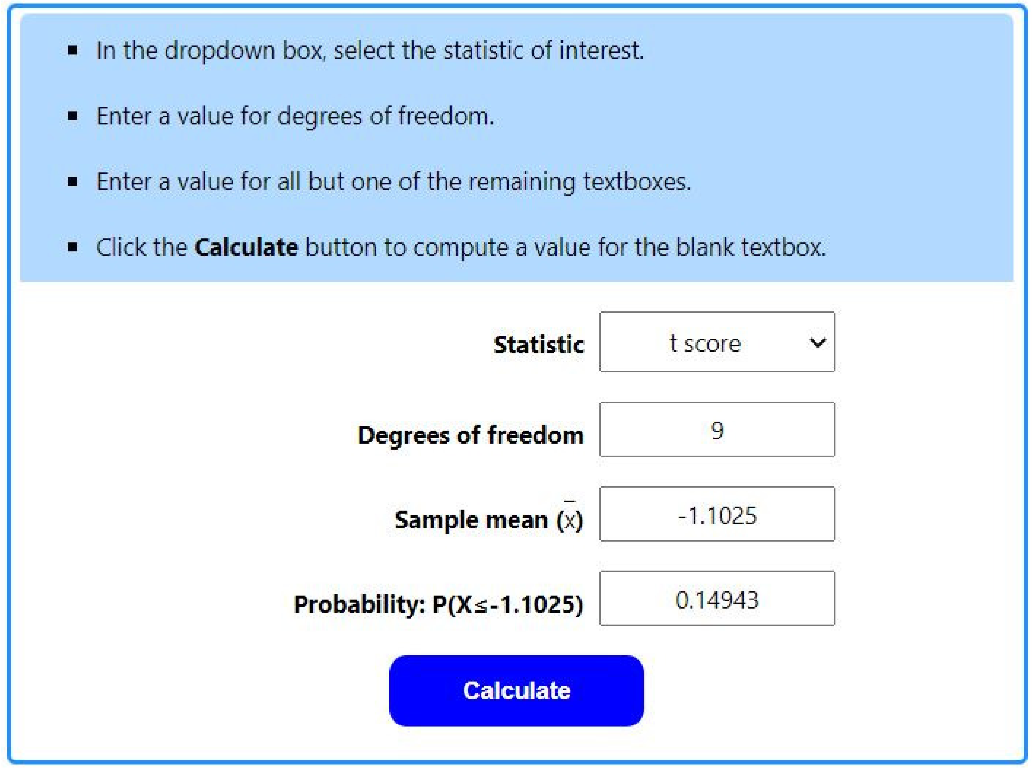
- Called the t-score

$$t = \left(\frac{39.5 - \bar{x}}{s}\right)$$

$$t = \left(\frac{39.5 - \bar{x}}{s}\right) = \left(\frac{39.5 - 34}{4.9889}\right) = 1.1025$$

Convert this to a probability using a Student-t table

- StatTrek also works
- Degrees of Freedom = Sample Size - 1
- p = 14.943%
- vs. 14.573% with Monte Carlo



Student-t Table

- df = degrees of freedom = sample size - 1
- Top = area of the tail
- Table = t-score
- t-score = 1.1025
- p = 14.943%

df \ p	0.001	0.0025	0.005	0.01	0.025	0.05	0.1	0.15	0.2
1	636.61900	318.30900	63.65670	31.82050	12.70620	6.31380	3.07770	1.96260	1.37640
2	31.59910	22.32710	9.92480	6.96460	4.30270	2.92000	1.88560	1.38620	1.06070
3	12.92400	10.21450	5.84090	4.54070	3.18240	2.35340	1.63770	1.24980	0.97850
4	8.61030	7.17320	4.60410	3.74690	2.77640	2.13180	1.53320	1.18960	0.94100
5	6.86880	5.89340	4.03210	3.36490	2.57060	2.01500	1.47590	1.15580	0.91950
6	5.95880	5.20760	3.70740	3.14270	2.44690	1.94320	1.43980	1.13420	0.90570
7	5.40790	4.78530	3.49950	2.99800	2.36460	1.89460	1.41490	1.11920	0.89600
8	5.04130	4.50080	3.35540	2.89650	2.30600	1.85950	1.39680	1.10810	0.88890
9	4.78090	4.29680	3.24980	2.82140	2.26220	1.83310	1.38300	1.09970	0.88340
10	4.58690	4.14370	3.16930	2.76380	2.22810	1.81250	1.37220	1.09310	0.87910
100	3.39050	3.17370	2.62590	2.36420	1.98400	1.66020	1.29010	1.04180	0.84520

90% confidence interval

- Each tail is 5%
- t-score = 1.83203

$$\bar{x} - 1.83203s < roll < \bar{x} + 1.83203s$$

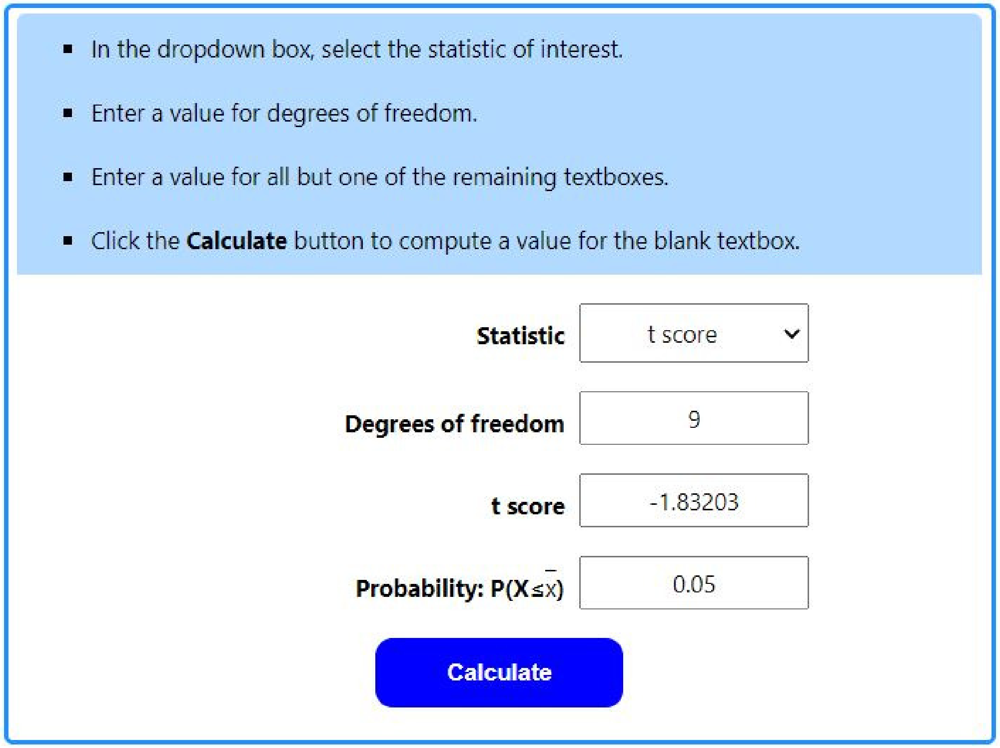
$$24.8602 < roll < 43.1398$$

vs. Monte Carlo

24.5 < y < 42.5

Note:

- Almost the same answer as Monte-Carlo
- Only required 10 die rolls vs. 100,000



What Sample Size is Needed?

- Two or more
- Less than a million
- Past 3 dof (n=4), you get diminishing returns
- A sample size of 4 to 10 is actually pretty good

df \ p	0.001	0.0025	0.005	0.01	0.025	0.05	0.1	0.15	0.2
1	636.61900	318.30900	63.65670	31.82050	12.70620	6.31380	3.07770	1.96260	1.37640
2	31.59910	22.32710	9.92480	6.96460	4.30270	2.92000	1.88560	1.38620	1.06070
3	12.92400	10.21450	5.84090	4.54070	3.18240	2.35340	1.63770	1.24980	0.97850
4	8.61030	7.17320	4.60410	3.74690	2.77640	2.13180	1.53320	1.18960	0.94100
5	6.86880	5.89340	4.03210	3.36490	2.57060	2.01500	1.47590	1.15580	0.91950
6	5.95880	5.20760	3.70740	3.14270	2.44690	1.94320	1.43980	1.13420	0.90570
7	5.40790	4.78530	3.49950	2.99800	2.36460	1.89460	1.41490	1.11920	0.89600
8	5.04130	4.50080	3.35540	2.89650	2.30600	1.85950	1.39680	1.10810	0.88890
9	4.78090	4.29680	3.24980	2.82140	2.26220	1.83310	1.38300	1.09970	0.88340
10	4.58690	4.14370	3.16930	2.76380	2.22810	1.81250	1.37220	1.09310	0.87910
20	3.84950	3.55180	2.84530	2.52800	2.08600	1.72470	1.32530	1.06400	0.86000
30	3.64600	3.38520	2.75000	2.45730	2.04230	1.69730	1.31040	1.05470	0.85380
40	3.55100	3.30690	2.70450	2.42330	2.02110	1.68390	1.30310	1.05000	0.85070
50	3.49600	3.26140	2.67780	2.40330	2.00860	1.67590	1.29870	1.04730	0.84890
100	3.39050	3.17370	2.62590	2.36420	1.98400	1.66020	1.29010	1.04180	0.84520

Handout:

2) The gain of 5 transistors were measured

- { 915, 602, 963, 839, 815 }
- mean = 826.8
- standard deviation = 138.9

a) What is the probability that any given transistor has a gain more than 600?

b) What is the 98% confidence interval for the gain of any given transistor?

Summary

Statistics is the mathematics for analyzing random processes

- You get a different result every time you run an experiment (i.e. lab data)

Chi-Squared tests let you test a distribution

- Is a die fair? Was data fudged?

Monte-Carlo simulations let you determine probabilities

- $p(y > 39.5) = ?$
- It requires a large sample size, however

Normal distribution gives the same result with no samples needed

- Assume you *know* the population's mean and standard deviation

Student-t Distributions give similar results with a small sample size

- Estimate the mean and standard deviation from your data
- More data helps, but a sample size of 4-10 is actually pretty good