Statistical Process Control Using Hidden Markov Models

Bob Moore
Senior Principal Process Engineer
www.biztransform.net

What Is A Hidden Markov Model?

- Hidden Markov Models (HMMs) are a method for analyzing the output and underlying structure (states) of a random or seemingly random process using only the data sequences observed on the process.
- HMMs provide a fast and efficient mathematical method for analyzing and drawing conclusions about systems where the underlying processes cannot be directly observed.
- HMMs have found successful applications in speech processing, speaker recognition, cryptology, signal processing, queuing theory, coding theory, and communications systems.

OK, First: Markov Chains

- Let P = the probability that a process is in state "X" at time t, given that we know the state of the process at *all previous* times.
- Let Q = the probability that a process is in state "X" at time t, given that we know *only* the state of the process at previous time mark (t-1).
- A random process is a Markov chain if P = Q.

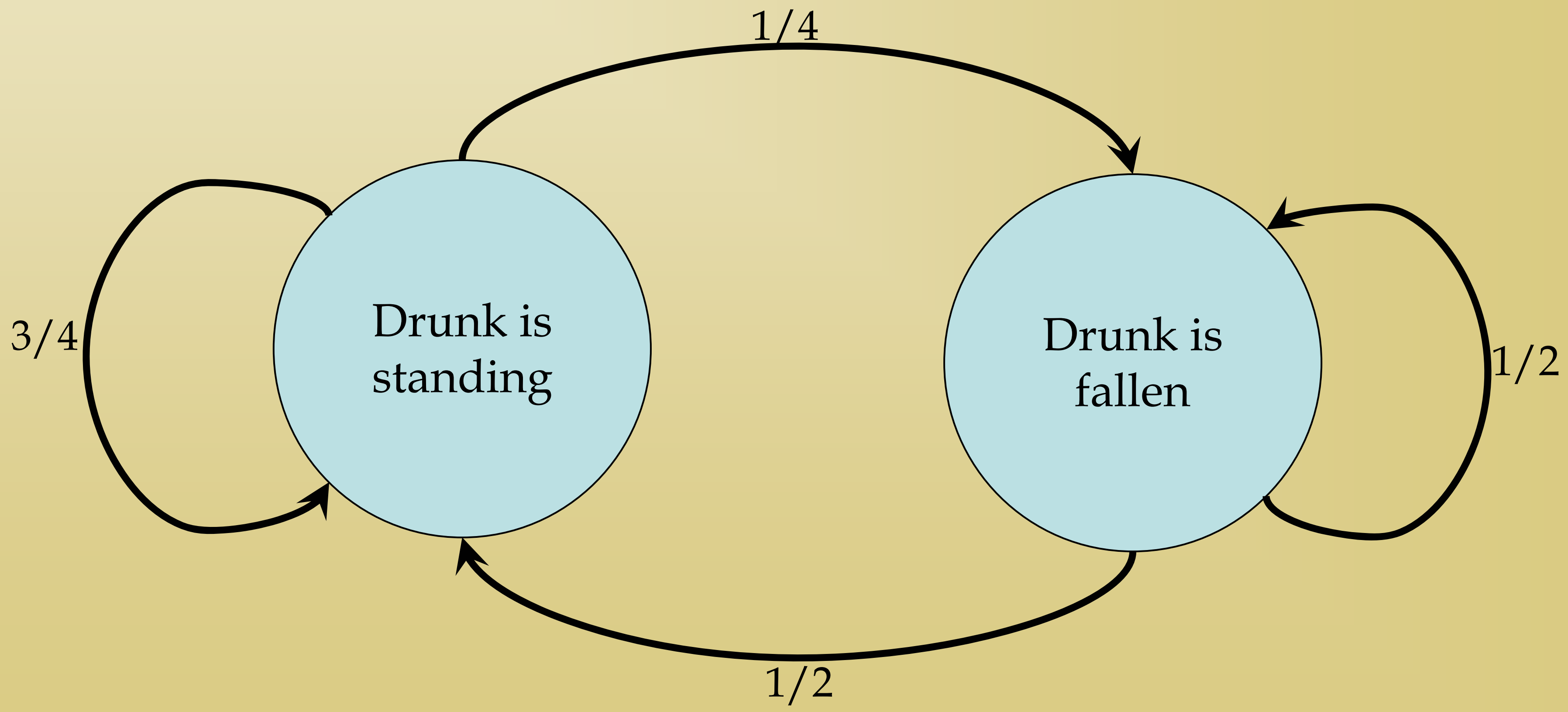
Markov Chain Example (1)

- We are approximately 750 miles from “The Strip” in Las Vegas—good for a long weekend trip in a car . . .
- A politically incorrect but classic Markov chain example:
  - Consider someone who is drunk.
- Our drunk is either standing up or fallen down.
  - If the drunk is standing now, then there is a ¾ chance that he/she is still standing when we check again in one minute.
  - If the drunk is standing now, then there is a ¼ chance that he/she is fallen in one minute
  - If the drunk is fallen now, then there is a ½ chance that he/she is standing in one minute.
  - If the drunk is fallen now, then there is a there is a ½ chance that he/she is still fallen in one minute.

Business
Transformation
Institute

Markov Chain Example (2)

- The state of the drunk at any given one minute interval forms a Markov chain.



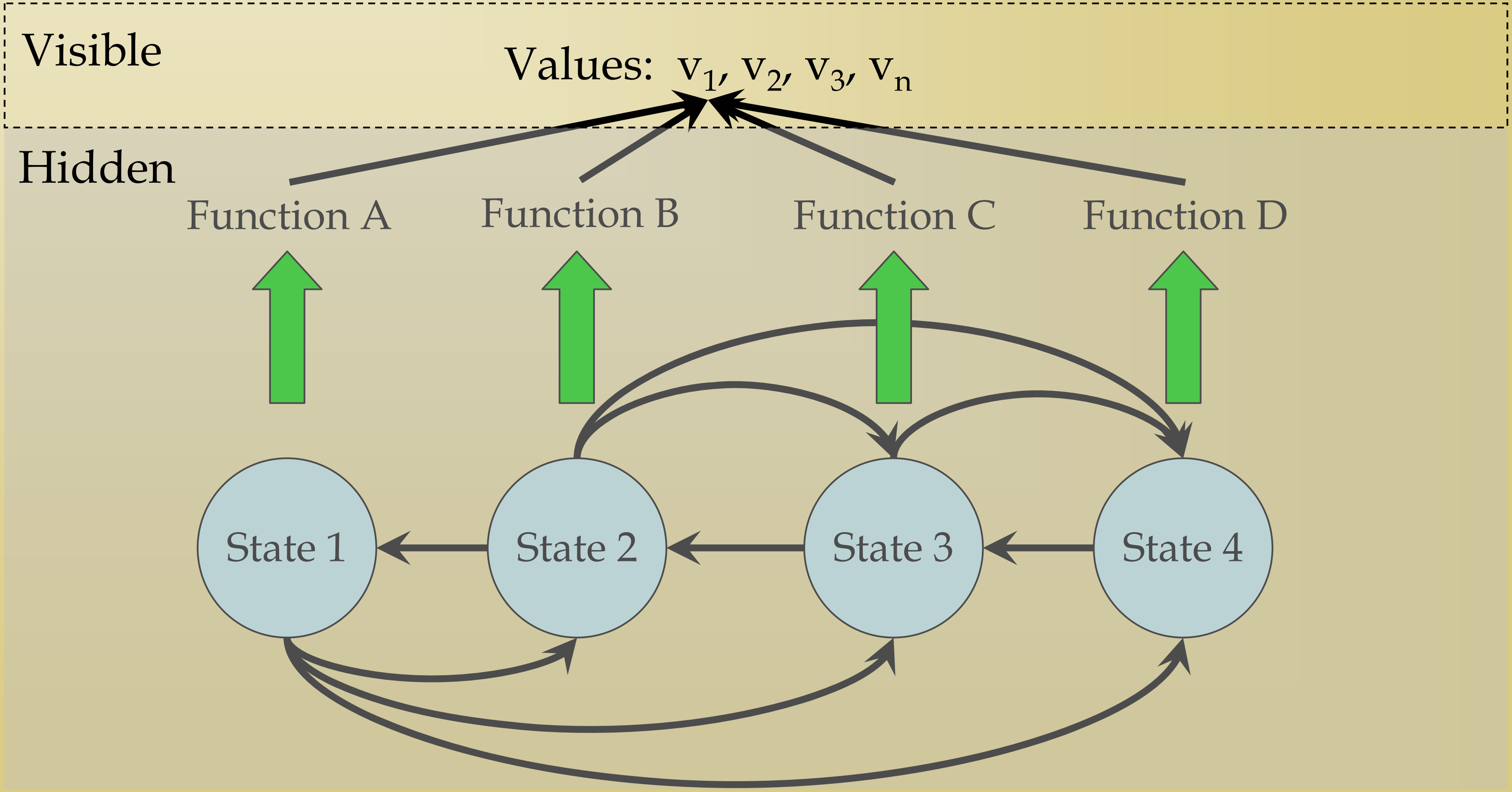
Markov Chain Example (3)

- We can represent this Markov chain as a state transition matrix T =

	Standing	Fallen
Standing	3/4	1/4
Fallen	1/2	1/2

HMMs Illustrated

In a HMM, we don’t get to observe the Markov chain— only values generated by functions that are set by the Markov chain. (This is why it is called “hidden”!)



Visible

Values: v_1, v_2, v_3, v_n

Hidden

Function A

Function B

Function C

Function D

State 1

State 2

State 3

State 4

An HMM Example (1)

- Since we are still approximately 750 miles from “The Strip” in Las Vegas . . .
- Consider a slot machine!
- Every time the slot machine’s “lever” is pulled, a symbol sequence is generated.

An HMM Example (2)

- We do not have any direct knowledge about the process that is generating this symbol sequence.
- The only observations that we have on the machine are the symbol sequences.
- We do know:
  - The mechanism generating the sequences assumes states which are more favorable or less favorable for a winning combination depending on how often the machine is played.
  - From our perspective, the machine's internal mechanism transitions between these states at apparently random times.
    - The progression may be structured in terms of which state passes into which state.

An HMM Example (3)

- HMMs can help us answer the following:
  - Can we determine the states of the slot machine's internal mechanism in terms of generating symbol sequences? (Can we pick a machine that is ripe for a winner?)
  - Can we calculate the chance of seeing any particular symbol sequence or collection of symbol sequences? (Can we figure out the chance of winning big?)
  - How can we adjust our model of the slot machine so that it most accurately describes the machine's behavior? (Can we make our predictions accurate enough and easy enough to be useful?)

Statistical Process Control Using HMMs

- HMMs may be used for SPC applications by recognizing that we are observing measurements about a process—but that we cannot directly observe the process itself.
- Our model consists of data (observed measurements) generated from our process. The data is generated by the characteristics of the process as determined by an underlying state model.
- The underlying Markov chain state model will have one or more hidden states corresponding to common cause behavior—and one or more hidden states corresponding to special cause behavior.

HMM Calculations

- This presentation is not intended to teach the mathematics of SPC calculations using HMMs.
  - Mathematics software such as MATLAB with the "Statistics Toolbox" provides all of the tools without any programming needed.
  - MATLAB also does traditional SPC calculations—we'll use it when comparing HMMs and control charts.

HMM Application (1)

- Rugby Software, Inc. monitors the health of its software development processes by measuring the number of lines of code (or function points, if preferred) produced by each development team per day.
  - We'll presume that Rugby has a good definition of what a "line of code" means in each development language being used.
  - Each development team consists of 5 individuals.
  - Rugby uses the scrum (agile) development life cycle (what else?)

HMM Application (2)

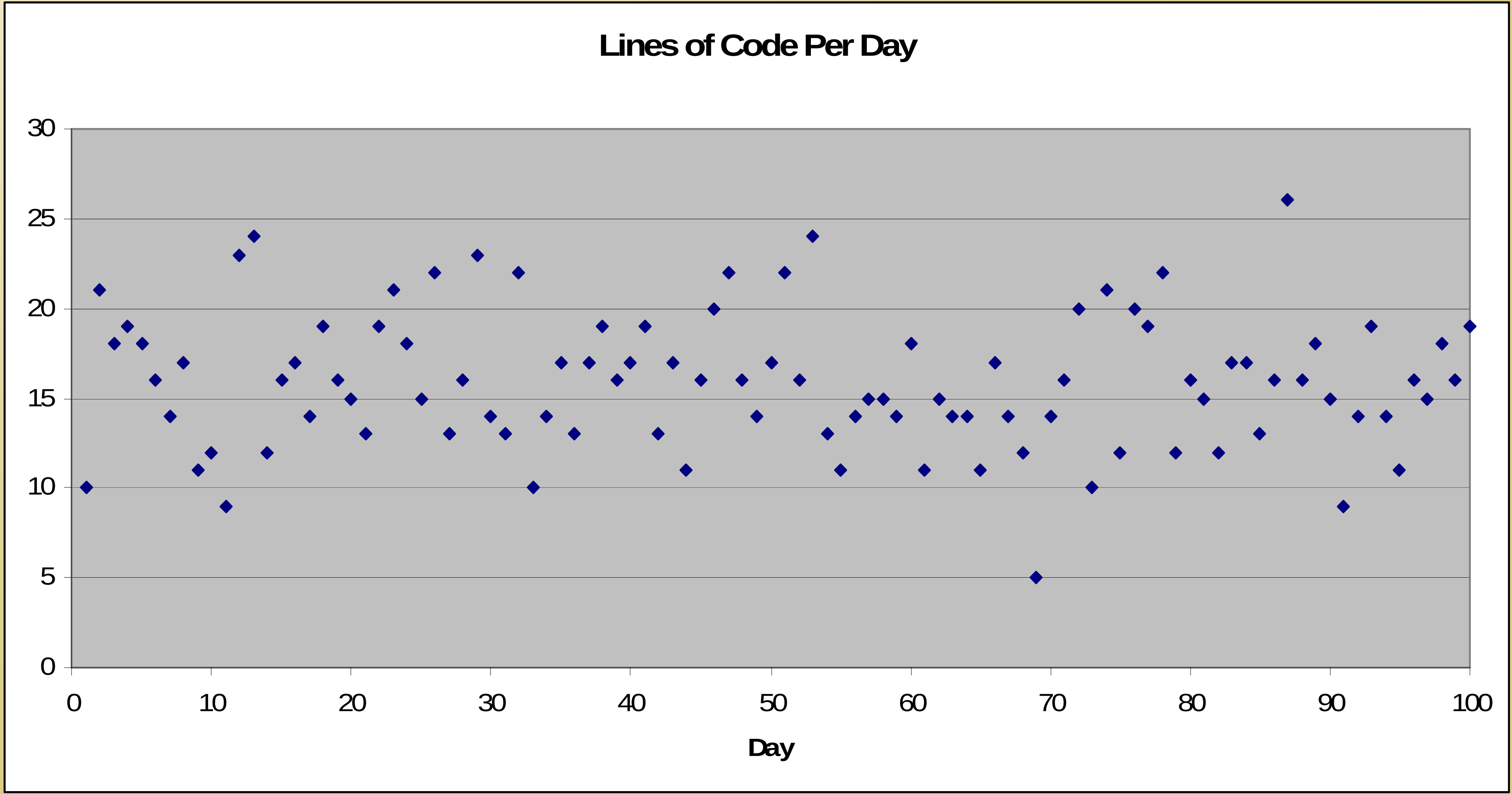
- The number of lines of code produced by a team on any given day is a Poisson distribution, with expected value λ = 20.
- Probability(n lines of code generated today) =

$$\frac{e^{-\lambda} \quad * \quad \lambda^{n}}{n!}$$

HMM Application (3)

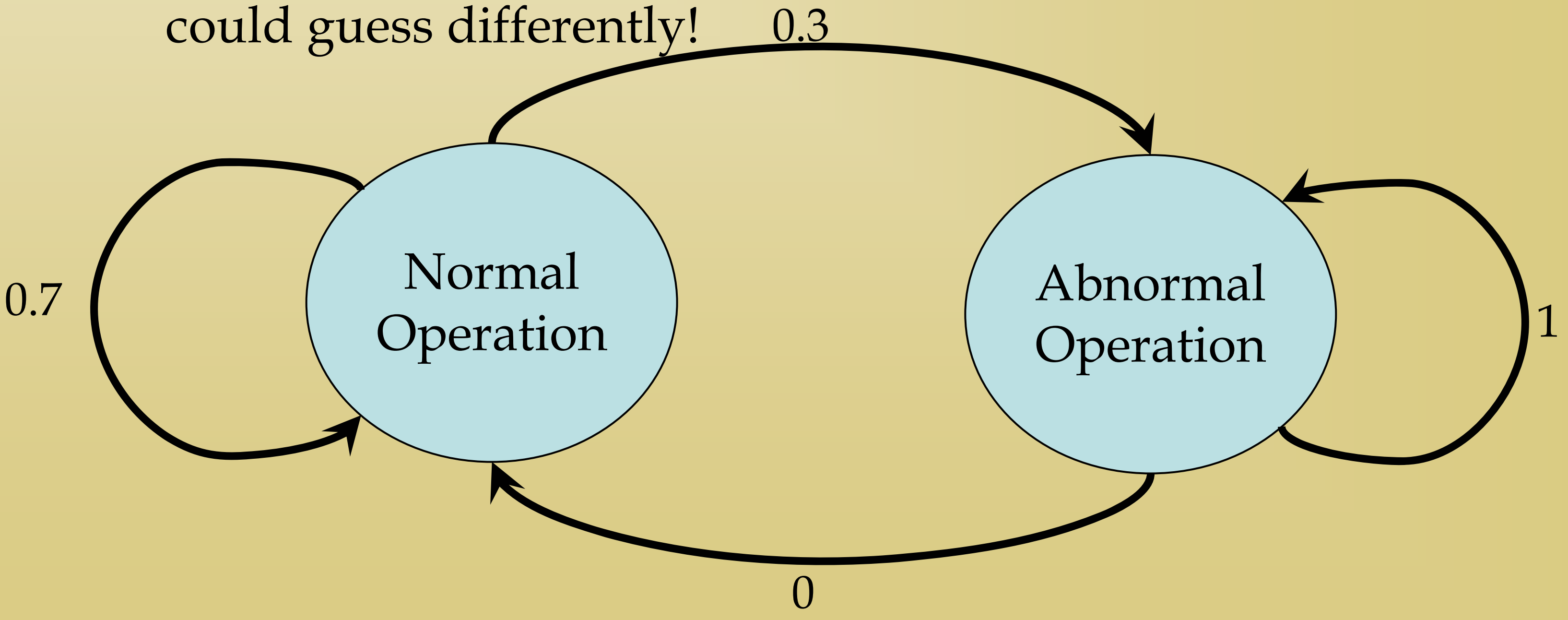
- Although not important for our model, λ might reasonably vary based on the team's mean experience and the number of people on the team.
- Over the course of 100 days, the number of lines of code team "A" generates looks like the plot on the next slide.

HMM Application (4)



HMM Application (5)

- We'll use the following model as our initial guess at the hidden Markov chain that is generating these measurements.
  - Our guess model has two states: "normal" and "abnormal" operation.
  - We also guess that the operations tend to stay "normal", but once they become abnormal then that state persists. Note: we could guess differently!



HMM Application (6)

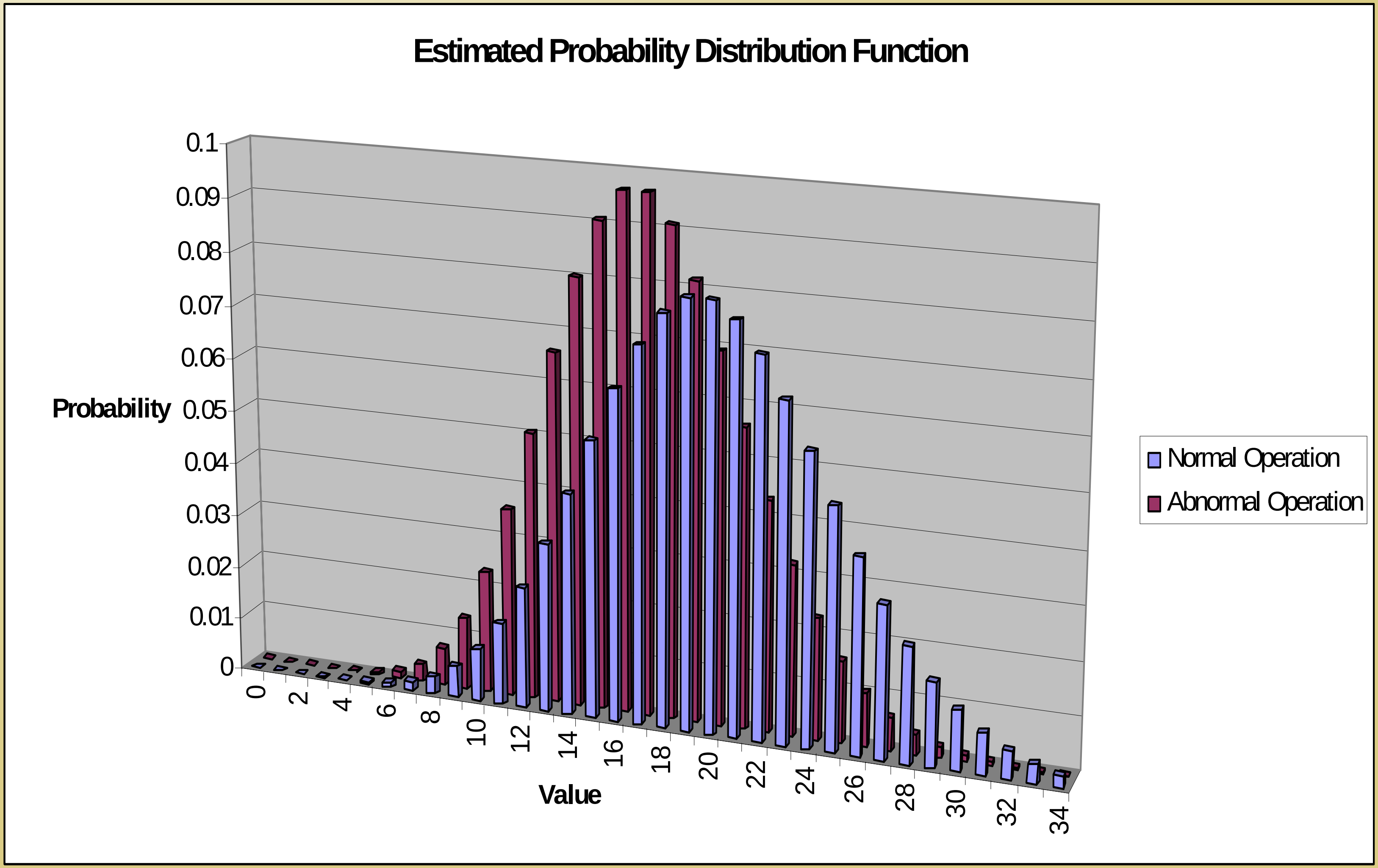
- This gives estimated state transition matrix T =

	Normal	Abnormal
Normal	0.7	0.3
Abnormal	0	1

HMM Application (7)

- We also need to guess at the probability of the number of lines of code generated when under “normal” and “abnormal” operations.
  - We would represent these in a matrix V.
    - V is a 2x36 matrix—too large to display in a readable way here.
  - The values of V are graphed on the next slide.

HMM Application (8)



HMM Application (9)

- We proceed as follows:
  - Use the Baum-Welch algorithm to iterate our guesses on T and V until we get convergence.
    - Implemented using the MATLAB function hmmtrain.
  - Once we've got good estimates, we can apply the Viterbi algorithm to assess the possible state transitions from "normal" to "abnormal".
    - Implemented using the MATLAB function hmmviterbi.
  - Estimate the conditional probability that, given the sequence of lines of code generated that we actually saw, a transition from "normal" to "abnormal" happened at any given point.
    - Implemented using the MATLAB function hmmdecode.
    - From an SPC viewpoint, this is the information that we are most interested in, since it gives us the probability that, at any given, point, we've changed from "normal" to "abnormal".

HMM Application (10)

- Our improved estimate of T =

0.8263	0.1737
0	1

- The actual T used for this discussion =

0.9	0.1
0	1

HMM Application (11)

- The ***actual*** sequence of states was:
  {1, 1, 1, 1, 1, 1, 1, 2, 2, . . ., 2}
- The ***estimated*** sequence of states that we saw is:
  {1, 1, 1, 1, 1, 2, 2, . . ., 2}
- The ***estimated*** conditional probability of state transition is:

0.9999	0.9990	0.9918	0.9397	0.8250	0.0010	0	…	0
0.0001	0.0010	0.0082	0.0603	0.1750	0.9990	1	…	0

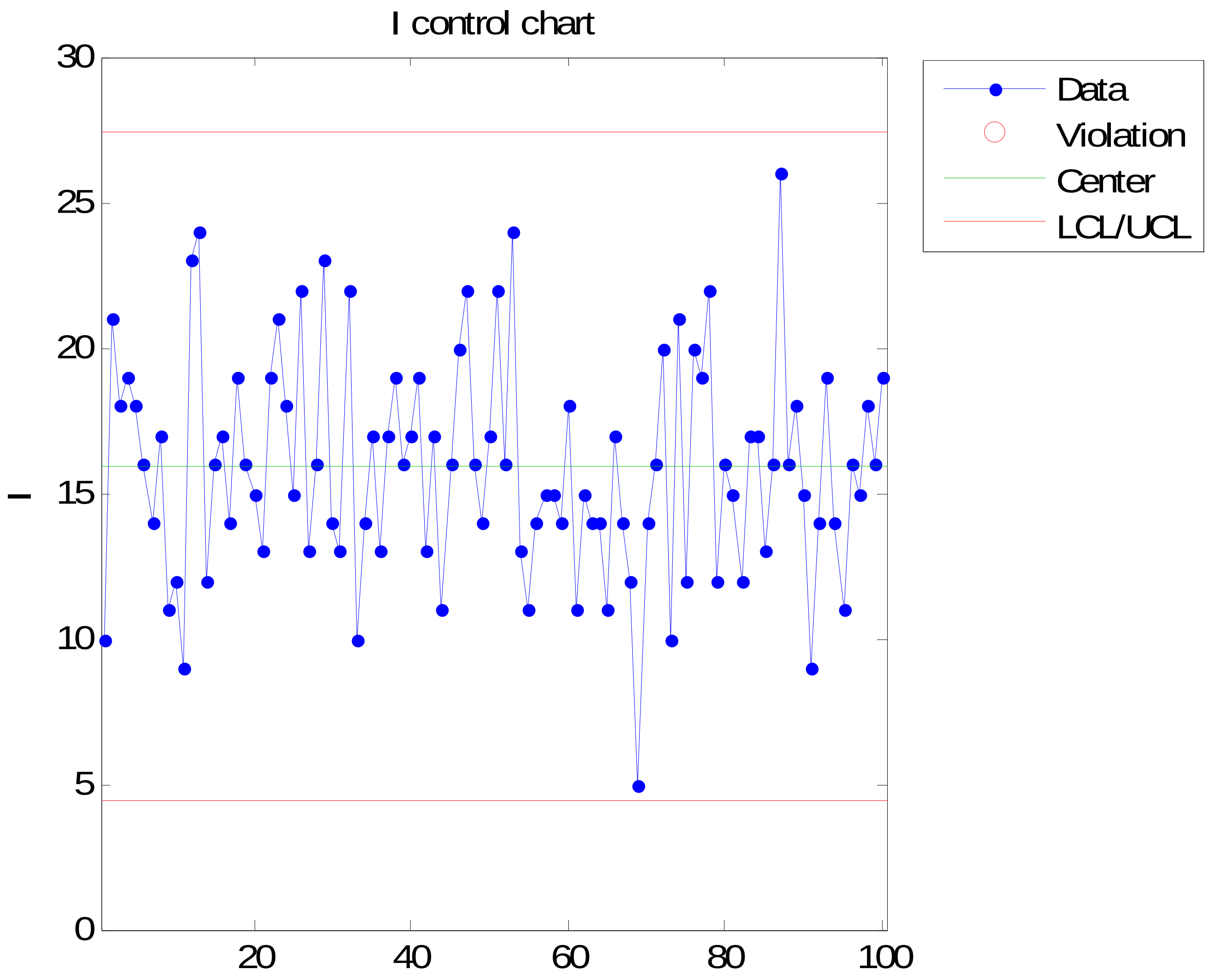
HMM Application (12)

- What does the estimated conditional probability of state transition mean?
  - At the first time point, we are 99.99% confident that we are in the normal state.
  - At time point 4, we are 93.97% confident that we are in the normal state.
  - At time point 5, our confidence drops to 82.5%.
  - At time point 6, our confidence drops to 0.1%.

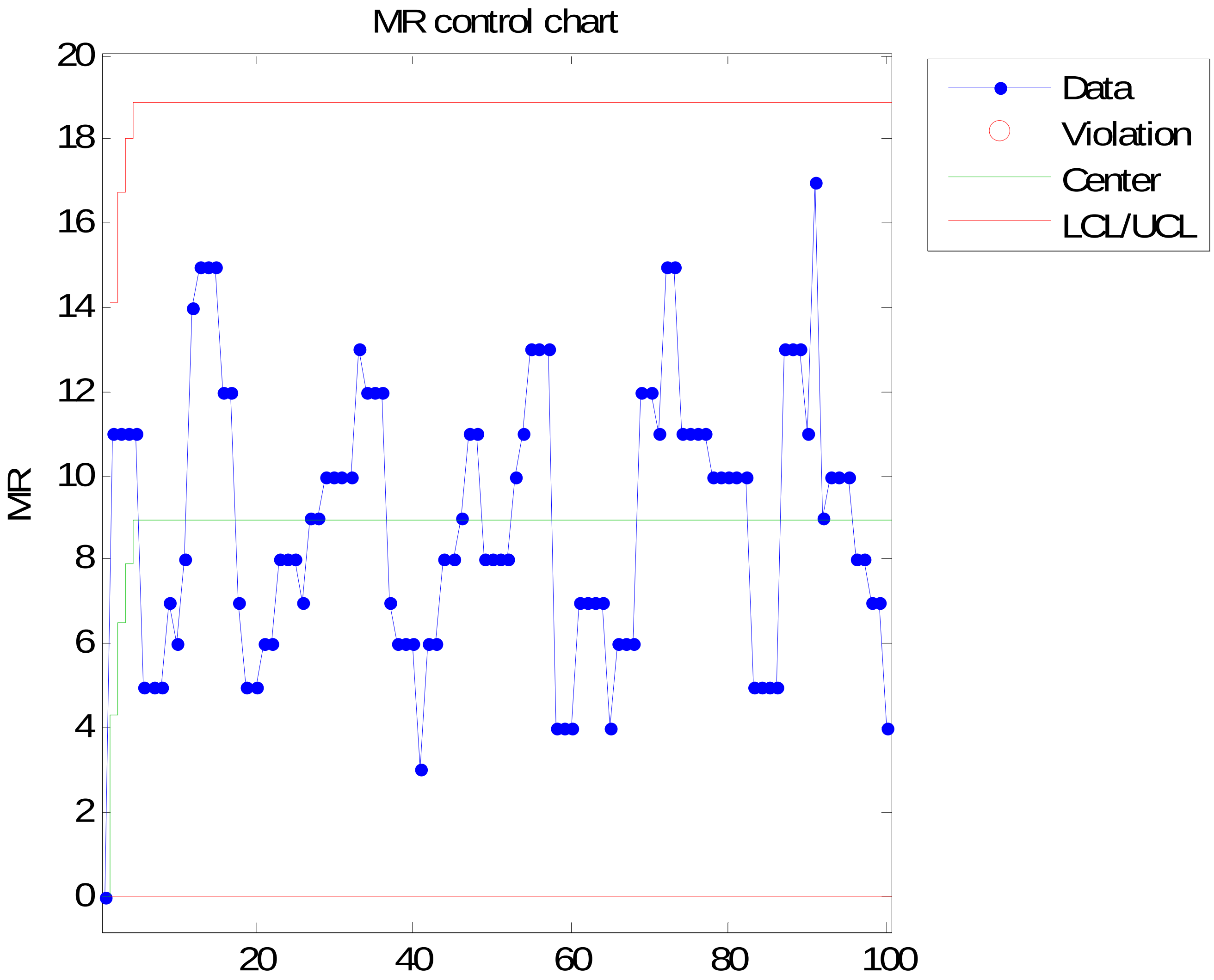
Comparison to Control Charts

- The HMM tells us that we are very confident the shift from normal to abnormal operations occurred by time point 6!
- The next three slides give, respectively:
  - An individual observation control chart for the same sequence data used above.
  - An individual observation moving range control chart.
  - An individual observation moving average control chart.
- In this example, traditional control chart methods do ***not*** show the shift from normal to abnormal!

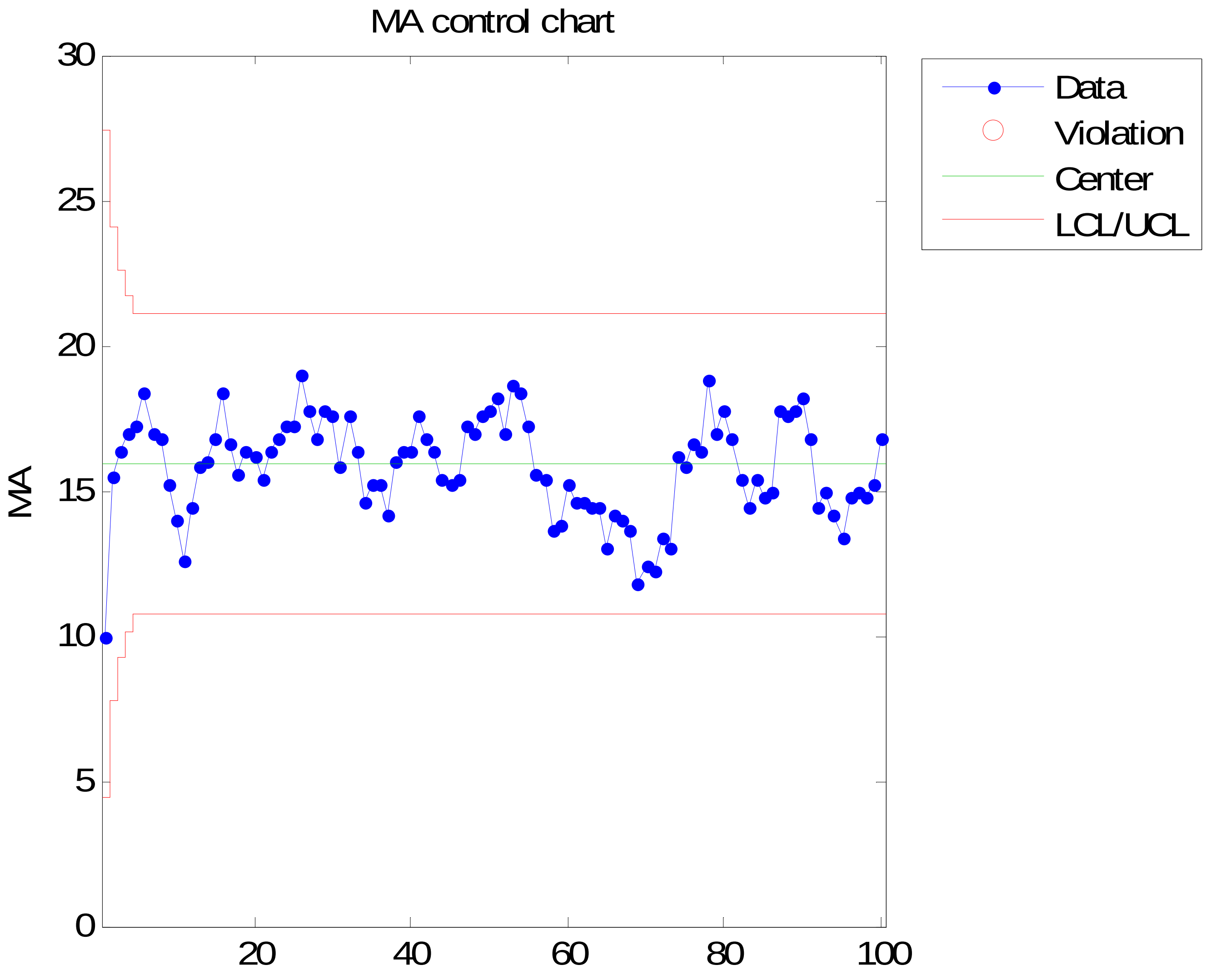
Individual Observation Control Chart



Individual Moving Range



Individual Moving Average



General Comparison to Control Charts (1)

- To compare HMM and control chart SPC techniques, 25 100-point sample sequences were run for comparison.
  - The point at which the process transitioned from normal to abnormal was randomly generated in each sequence.
  - The normal state process performance was always modeled as a Poisson distribution with λ = 20.
  - The abnormal state process performance was modeled as a Poisson distribution with λ randomly chosen, 10 ≤ λ ≤ 20.
- The number of times that the HMM or control chart detected the shift from normal to abnormal within ±5 data points is given on the next slide.

General Comparison to Control Charts (2)

		Control Chart	
		Detected Shift	Did Not Detect Shift
HMM	Detected Shift	3	11
	Did Not Detect Shift	2	9

Conclusions

- HMMs may be useful in SPC monitoring.
- HMMs may be more sensitive that traditional control charts in identifying when special causes of variation are occurring.
- More work is needed to better document applications of HMMs in this context.
  - Existing published scholarly papers on this subject exist, but are limited.
    - For a basic paper with only undergraduate level mathematics, see Tai, Ching, and Chan, “Hidden Markov Model for the Detection of Machine Failure”, http://hkumath.hku.hk/~imr/IMRPreprintSeries/2006/IMR2006-07.pdf