

# Markov Chains and Numerical Transitions

Jeff Cope

CS567 – Fall 2009

Graduate Project Presentation

# Andrey Markov (1856 – 1922)



- “A Markov chain is a kind of probabilistic state machine that can be easily trained given a set of training data”
  - *Artificial Intelligence: A Systems Approach* by M. Tim Jones

# What is a Numerical Transition?

Within a value, it's the change from one group of digits to another group of digits.

# Numerical Transition Example

- Given the value: 12345
- Possible transitions sets include:
  - {1, 2, 3, 4, 5}
  - {123, 234, 345}
  - {1, 12, 23, 34, 45, 5}
  - {1, 12, 123, 234, 345, 45, 5}

# Why is this useful?

- Finding patterns in value, while maintaining the state at each stage.

# What kind I chose.

- Basic Rules:

- Add the character “ ” to the beginning and end of the value as an *empty character*.
- “ ” evaluates the same as “ ”, so “ 1” = “ 1” and “5 ” = “5 ”

- Example:

- Given: Value = 12345, Max length = 3
- Result set:

- {“ 1”, “ 12”, “123”, “234”, “345”, “45 ”, “5 ”}

# Program Motivation

- Finding patterns that could potentially help in determining if a value is prime or not.

# Program Implementation

- Java v1.6
- Netbeans IDE

# Development Stages

- Created:
  - N-Dimensional Markov Chain
  - Transition Iterator for values
    - Letter, Word, Transition
  - Value Generators
    - Primes, Integer, Multiple\*Primes, Multiple\*Integer
  - GUI Interface
    - Table displays and exports

# Program Demonstration

# Program Results

- Lots of Data, lots of patterns, but nothing I can say definitively about the patterns without a mathematical proof.

# Future Versions?

- Evaluation of transitions in different base
- Regular expression searches
- Probabilistic evaluations of counts

Questions?

Thank You