Bayesian Methods for Data Analysis in Software Engineering

Information Theory and Stochastic Sampling

Mohan Sridharan[1] Akbar Siami Namin[2]

[1] Stochastic Estimation and Autonomous Robotics (SEAR) Lab
Department of Computer Science
Texas Tech University

[2] AdVanced Empirical Software Testing and Analysis (AVESTA) Research Group
Department of Computer Science
Texas Tech University

May 3, 2010; Cape Town

Session 3

- **9.00–10.30**:
  - Introduction.
  - Statistical analysis; hypothesis testing.
  - Basic probability, Bayes' rule.
- **11.00–12.30**:
  - Bayesian classification.
  - Bayesian regression.
  - Bayesian inference.
- **14.00–15.30**:
  - Information theory.
  - Stochastic sampling.
- **16.00–17.30**:
  - Markov decision processes.
  - Partially observable Markov decision processes.
  - Discussion.

Session 3: Information Theory

- **9.00–10.30**:
  - Introduction.
  - Statistical analysis; hypothesis testing.
  - Basic probability, Bayes' rule.
- **11.00–12.30**:
  - Bayesian classification.
  - Bayesian regression.
  - Bayesian inference.
- **14.00–15.30**:
  - Information theory.
  - Stochastic sampling.
- **16.00–17.30**:
  - Markov decision processes.
  - Partially observable Markov decision processes.
  - Discussion.

Information Content

- Measure information gained by specific observations of a random variable X.
- Occurrence of a highly improbable event provides more information than the occurrence of a very likely event.
- The measure of information content therefore depends on probability distribution $p(x)$.
- Need a monotonic function of probability $p(x)$ that expresses the information content of the variable.

Information Content

- Measure information gained by specific observations of a random variable X.
- Occurrence of a highly improbable event provides more information than the occurrence of a very likely event.
- The measure of information content therefore depends on probability distribution $p(x)$.
- Need a monotonic function of probability $p(x)$ that expresses the information content of the variable.

Information Content

- Measure information gained by specific observations of a random variable X.
- Occurrence of a highly improbable event provides more information than the occurrence of a very likely event.
- The measure of information content therefore depends on probability distribution $p(x)$.
- Need a monotonic function of probability $p(x)$ that expresses the information content of the variable.

Information Content

- Measure information gained by specific observations of a random variable X.
- Occurrence of a highly improbable event provides more information than the occurrence of a very likely event.
- The measure of information content therefore depends on probability distribution $p(x)$.
- Need a monotonic function of probability $p(x)$ that expresses the information content of the variable.

Requirements

- For two unrelated events x, y information gained by observing both of them is the sum of information gained by observing each event separately:

$$h(x, y) = h(x) + h(y) \tag{1}$$

- If two events x, y are *statistically independent*:

$$p(x, y) = p(x) \cdot p(y) \tag{2}$$

- $h(x)$ is therefore a *logarithmic function* of $p(x)$:

$$h(x) = -\log\, p(x) \tag{3}$$

- Negative sign ensures information gain is ≥ 0. Logarithm to base 2 implies the units of $h(x)$ are bits.

Requirements

- For two unrelated events x, y information gained by observing both of them is the sum of information gained by observing each event separately:

$$h(x, y) = h(x) + h(y) \tag{1}$$

- If two events x, y are *statistically independent*:

$$p(x, y) = p(x) \cdot p(y) \tag{2}$$

- $h(x)$ is therefore a *logarithmic function* of $p(x)$:

$$h(x) = -\log\, p(x) \tag{3}$$

- Negative sign ensures information gain is ≥ 0. Logarithm to base 2 implies the units of $h(x)$ are bits.

Requirements

- For two unrelated events x, y information gained by observing both of them is the sum of information gained by observing each event separately:

$$h(x, y) = h(x) + h(y) \tag{1}$$

- If two events x, y are *statistically independent*:

$$p(x, y) = p(x) \cdot p(y) \tag{2}$$

- $h(x)$ is therefore a *logarithmic function* of $p(x)$:

$$h(x) = -\log\, p(x) \tag{3}$$

- Negative sign ensures information gain is ≥ 0. Logarithm to base 2 implies the units of $h(x)$ are bits.

Requirements

- For two unrelated events x, y information gained by observing both of them is the sum of information gained by observing each event separately:

$$h(x, y) = h(x) + h(y) \tag{1}$$

- If two events x, y are *statistically independent*:

$$p(x, y) = p(x) \cdot p(y) \tag{2}$$

- $h(x)$ is therefore a *logarithmic function* of $p(x)$:

$$h(x) = -\log\, p(x) \tag{3}$$

- Negative sign ensures information gain is ≥ 0. Logarithm to base 2 implies the units of $h(x)$ are bits.

Definition

- Entropy of the random variable is the *expectation* of $h(x)$ with respect to $p(x)$:

$$H[X] = -\sum_i p(x_i) \log p(x_i) \tag{4}$$

- $\lim_{p \to 0} p \log(p) = 0$ i.e. if $p(x) = 0$ then $p(x) \log p(x) = 0$.
- Extension to continuous random variables:

$$H[X] = -\int p(x) \log p(x) \, dx \tag{5}$$

Definition

- Entropy of the random variable is the *expectation* of $h(x)$ with respect to $p(x)$:

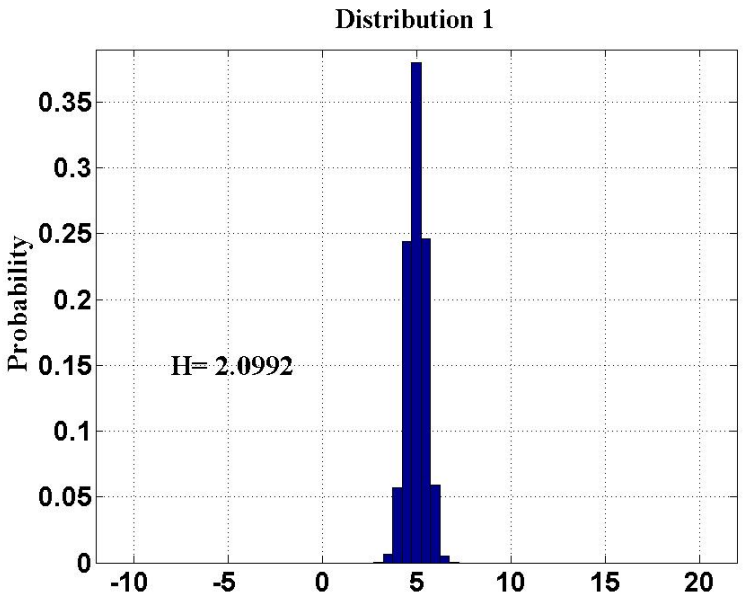
$$H[X] = -\sum_i p(x_i) \log p(x_i) \tag{4}$$

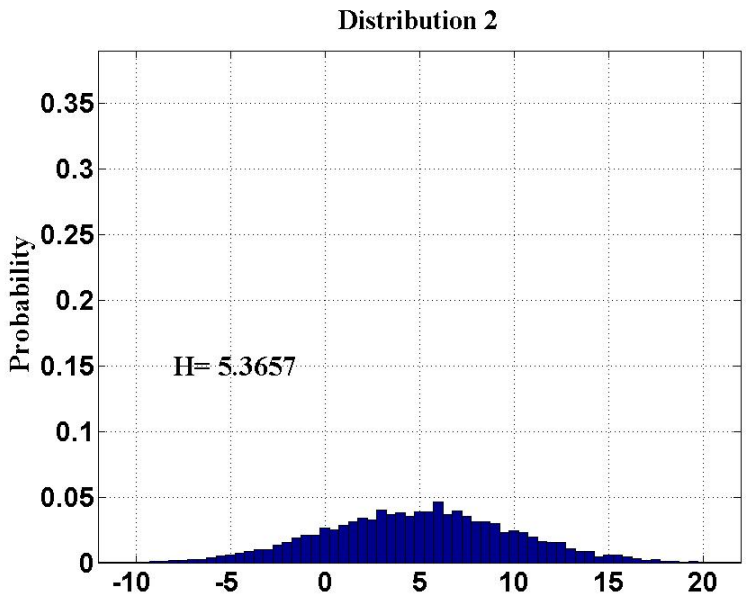
- $\lim_{p \to 0} p \log(p) = 0$ i.e. if $p(x) = 0$ then $p(x) \log p(x) = 0$.
- Extension to continuous random variables:

$$H[X] = -\int p(x) \log p(x) \, dx \tag{5}$$

Illustrative Example

- Uniform distribution: *high entropy*; distribution with sharp peaks: *low entropy*.





Encoding Information

- Entropy of a random variable X with eight possible states that are equally likely:

$$H[X] = \sum_{i=1}^{8} \frac{1}{8} \log_2 \frac{1}{8} = -8 \times \frac{1}{8} \log_2 \frac{1}{8} = 3(\text{bits}). \quad (6)$$

- Compute lower bound on number of bits needed to represent state of a random variable.
- Used in Shannon's *noiseless coding theorem*.
- Measures degree of disorder in the system.

Encoding Information

- Entropy of a random variable X with eight possible states that are equally likely:

$$H[X] = \sum_{i=1}^{8} \frac{1}{8} \log_2 \frac{1}{8} = -8 \times \frac{1}{8} \log_2 \frac{1}{8} = 3(\text{bits}). \quad (6)$$

- Compute lower bound on number of bits needed to represent state of a random variable.

- Used in Shannon's *noiseless coding theorem*.
- Measures degree of disorder in the system.

Encoding Information

- Entropy of a random variable X with eight possible states that are equally likely:

$$H[X] = \sum_{i=1}^{8} \frac{1}{8} \log_2 \frac{1}{8} = -8 \times \frac{1}{8} \log_2 \frac{1}{8} = 3(\text{bits}). \quad (6)$$

- Compute lower bound on number of bits needed to represent state of a random variable.

- Used in Shannon's *noiseless coding theorem*.
- Measures degree of disorder in the system.

Relative Entropy

- Relative entropy or Kullback-Leibler Divergence of probability distributions $p(x), q(x)$:

$$KL(p\|q) = -\int p(x) \log \frac{q(x)}{p(x)} dx \tag{7}$$

- Measure *divergence* between the *unknown* distribution: $p(x)$ and the approximate estimate: $q(x)$.

- Not symmetrical: $KL(p\|q) \neq KL(q\|p)$.

Relative Entropy

- Relative entropy or Kullback-Leibler Divergence of probability distributions $p(x), q(x)$:

$$KL(p\|q) = -\int p(x) \log \frac{q(x)}{p(x)} dx \tag{7}$$

- Measure *divergence* between the *unknown* distribution: $p(x)$ and the approximate estimate: $q(x)$.

- Not symmetrical: $KL(p\|q) \neq KL(q\|p)$.

Relative Entropy

- Relative entropy or Kullback-Leibler Divergence of probability distributions $p(x), q(x)$:

$$KL(p\|q) = -\int p(x) \log \frac{q(x)}{p(x)} dx \tag{7}$$

- Measure *divergence* between the *unknown* distribution: $p(x)$ and the approximate estimate: $q(x)$.

- Not symmetrical: $KL(p\|q) \neq KL(q\|p)$.

Mutual Information

- Given joint distribution $p(x, y)$ between variables X, Y, Mutual Information is defined as:

$$I[X, Y] = KL(p(x, y) \| p(x)p(y)) \tag{8}$$
$$= -\int\int p(x, y) \log \frac{p(x)p(y)}{p(x, y)} \, dx \, dy$$

- Related to conditional entropy—reduction in uncertainty about X as a result of knowledge about Y:

$$I[X, Y] = H[X] - H[X|Y] = H[Y] - H[Y|X] \tag{9}$$

Mutual Information

- Given joint distribution $p(x, y)$ between variables X, Y, Mutual Information is defined as:

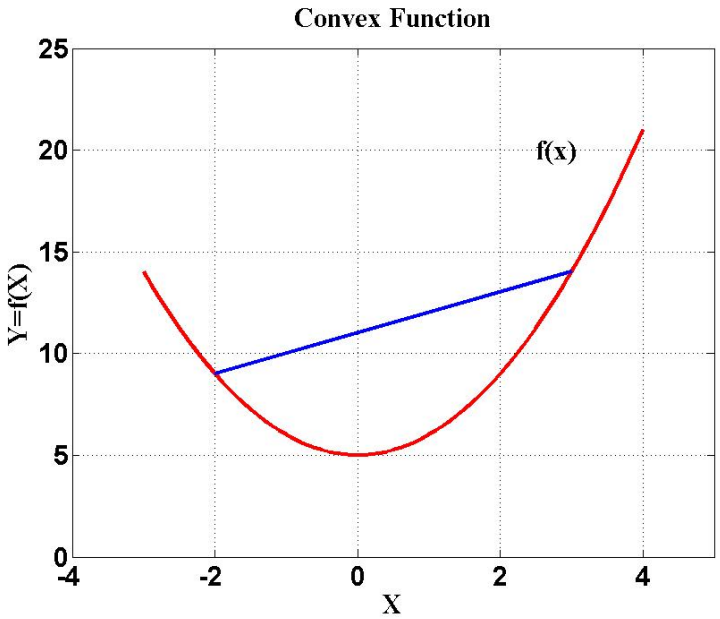
$$I[X, Y] = KL(p(x, y) \| p(x)p(y)) \tag{8}$$
$$= -\int\int p(x, y) \log \frac{p(x)p(y)}{p(x, y)} \, dx \, dy$$

- Related to conditional entropy—reduction in uncertainty about X as a result of knowledge about Y:

$$I[X, Y] = H[X] - H[X|Y] = H[Y] - H[Y|X] \tag{9}$$

Convexity

- Function $f(x)$ is *convex* if every chord is on or above the function.



- If $f(x)$ is convex, $-f(x)$ is concave.

Further Reading


Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Second Edition, Wiley-Interscience, 2005.


C. Bishop. *Pattern Recognition and Machine Learning*. Springer publishing house, 2007.

Session 3: Stochastic Sampling

- **9.00–10.30**:
  - Introduction.
  - Statistical analysis; hypothesis testing.
  - Basic probability, Bayes' rule.
- **11.00–12.30**:
  - Bayesian classification.
  - Bayesian regression.
  - Bayesian inference.
- **14.00–15.30**:
  - Information theory.
  - Stochastic sampling.
- **16.00–17.30**:
  - Markov decision processes.
  - Partially observable Markov decision processes.
  - Discussion.

Introduction

- Applicable in domains where multiple hypotheses need to be tracked.
- The functional form of the true underlying distribution is *unknown*.
- Probabilistic representation for each hypothesis.
- Iteratively identify the most likely hypotheses, i.e., direct focus towards the more important hypotheses.

Introduction

- Applicable in domains where multiple hypotheses need to be tracked.
- The functional form of the true underlying distribution is *unknown*.
- Probabilistic representation for each hypothesis.
- Iteratively identify the most likely hypotheses, i.e., direct focus towards the more important hypotheses.

Introduction

- Applicable in domains where multiple hypotheses need to be tracked.
- The functional form of the true underlying distribution is *unknown*.
- Probabilistic representation for each hypothesis.
- Iteratively identify the most likely hypotheses, i.e., direct focus towards the more important hypotheses.

Introduction

- Applicable in domains where multiple hypotheses need to be tracked.
- The functional form of the true underlying distribution is *unknown*.
- Probabilistic representation for each hypothesis.
- Iteratively identify the most likely hypotheses, i.e., direct focus towards the more important hypotheses.

Sampling Methods

- Standard sampling algorithms: adaptive rejection sampling, importance sampling, sampling importance resampling.
- Advanced sampling algorithms: Markov Chain Monte Carlo (MCMC), Gibbs sampling, slice sampling, hybrid approach.
- Several applications:
  - Tracking multiple humans in image sequences.
  - Finding most likely robot location, i.e., robot localization.
  - Finding likely locations of celestial objects, i.e., in astronomy.

Sampling Methods

- Standard sampling algorithms: adaptive rejection sampling, importance sampling, sampling importance resampling.
- Advanced sampling algorithms: Markov Chain Monte Carlo (MCMC), Gibbs sampling, slice sampling, hybrid approach.
- Several applications:
  - Tracking multiple humans in image sequences.
  - Finding most likely robot location, i.e., robot localization.
  - Finding likely locations of celestial objects, i.e., in astronomy.

Sampling Methods

- Standard sampling algorithms: adaptive rejection sampling, importance sampling, sampling importance resampling.
- Advanced sampling algorithms: Markov Chain Monte Carlo (MCMC), Gibbs sampling, slice sampling, hybrid approach.
- Several applications:
  - Tracking multiple humans in image sequences.
  - Finding most likely robot location, i.e., robot localization.
  - Finding likely locations of celestial objects, i.e., in astronomy.

Mathematical Formulation

- Bayes filter:

$$\forall x_t : \overline{bel}(x_t) = \int p(x_t|u_t, x_{t-1})\, bel(x_{t-1})\, dx_{t-1} \qquad (10)$$

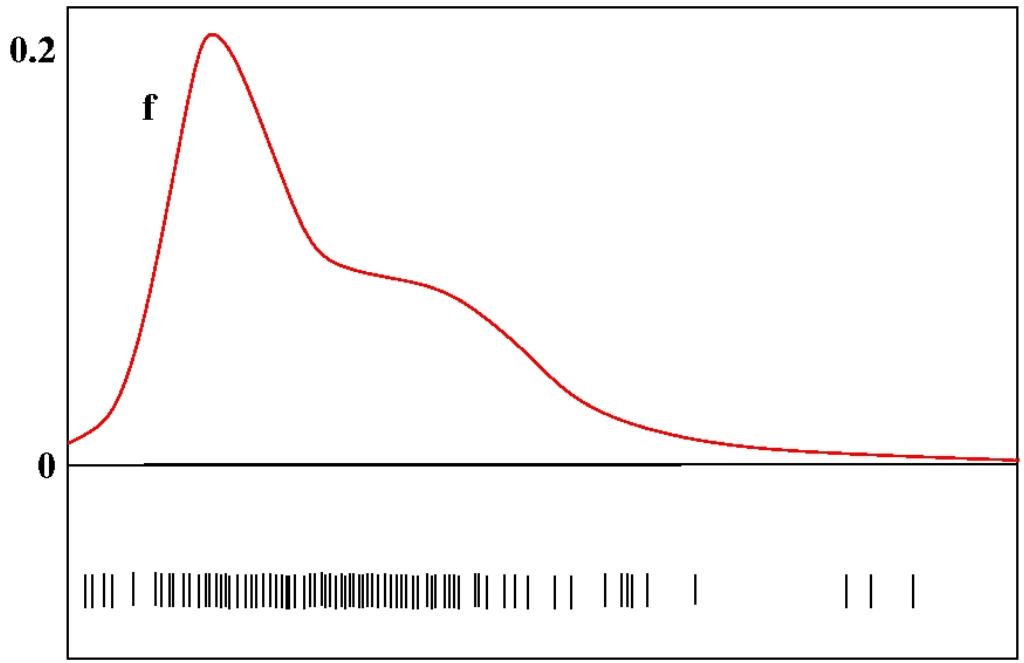
$$bel(x_t) = \eta\ p(z_t|x_t)\, \overline{bel}(x_t)$$

- Need to model *target distribution* $f(x)$ that cannot be observed directly.
- Use a *proposal distribution* $g(x)$ to estimate $f(x)$.
- Function f corresponds to $bel(x_t)$ while g corresponds to $\overline{bel}(x_t)$.

Target Distribution

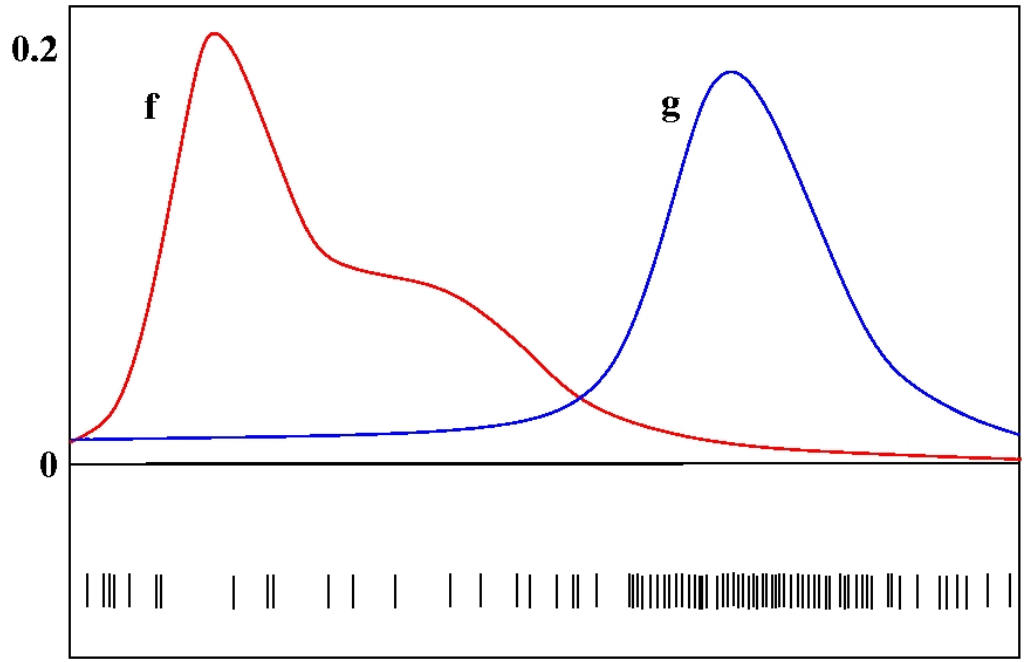
- Need to generate samples from a target distribution:

$$E[f] = \frac{1}{L}\sum_{l=1}^{L} f(\boldsymbol{z}_l) \tag{11}$$



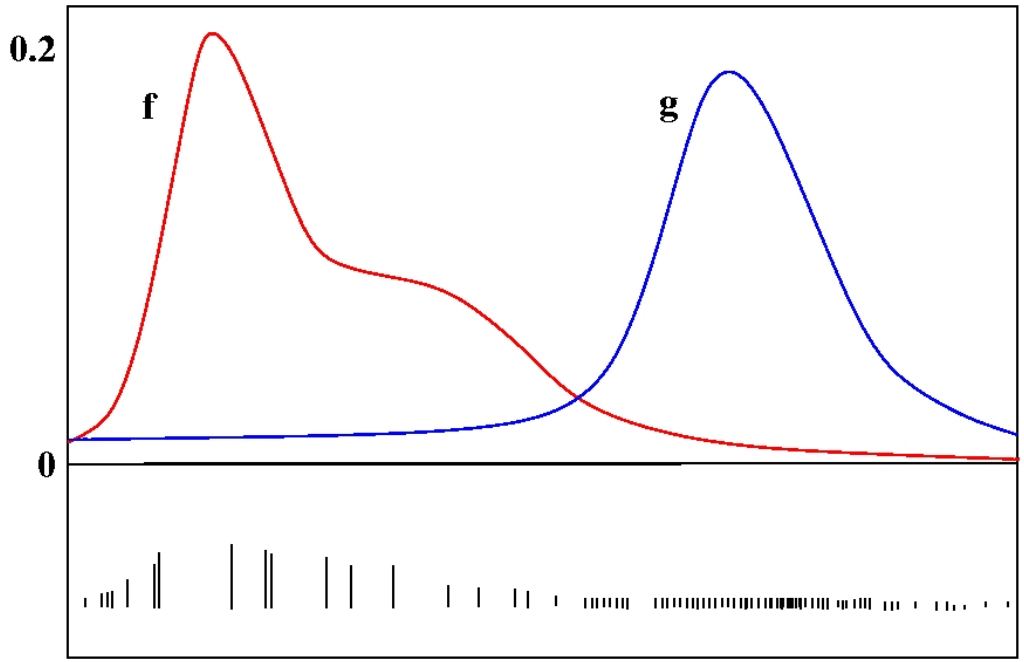
Proposal Distribution

- Generate samples from a proposal distribution.



Re-weighting Samples

- Re-weight samples based on how well they represent the target distribution.



Importance Sampling Formulation

- Assign probability to each hypothesis.
- Generate initial set of *samples* of each hypothesis based on the corresponding probabilities.
- In each of a finite set of iterations:
  - Adjust samples to account for dynamic changes in the system: *prediction* step.
  - Use observations of the system to update probabilities of the samples: *correction* step.
  - Resample, i.e., generate samples of each hypothesis proportional to the updated probabilities.

Importance Sampling Formulation

- Assign probability to each hypothesis.
- Generate initial set of *samples* of each hypothesis based on the corresponding probabilities.

- In each of a finite set of iterations:
  - Adjust samples to account for dynamic changes in the system: *prediction* step.
  - Use observations of the system to update probabilities of the samples: *correction* step.
  - Resample, i.e., generate samples of each hypothesis proportional to the updated probabilities.

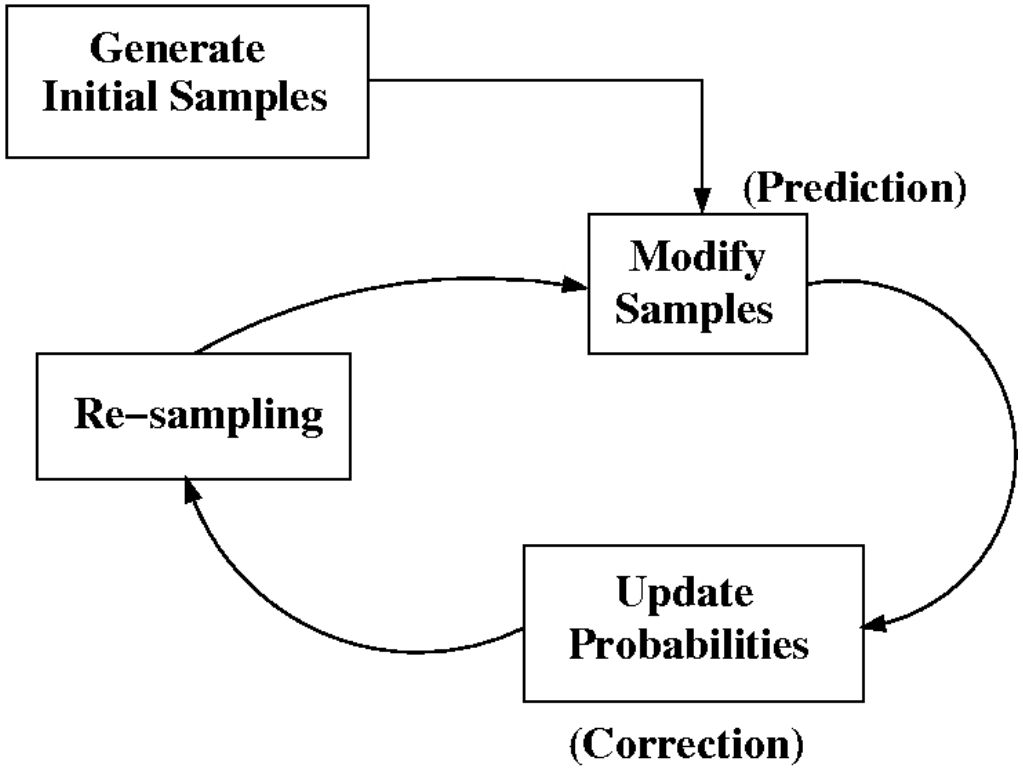
Importance Sampling Formulation

- Assign probability to each hypothesis.
- Generate initial set of *samples* of each hypothesis based on the corresponding probabilities.

- In each of a finite set of iterations:
  - Adjust samples to account for dynamic changes in the system: *prediction* step.
  - Use observations of the system to update probabilities of the samples: *correction* step.
  - Resample, i.e., generate samples of each hypothesis proportional to the updated probabilities.

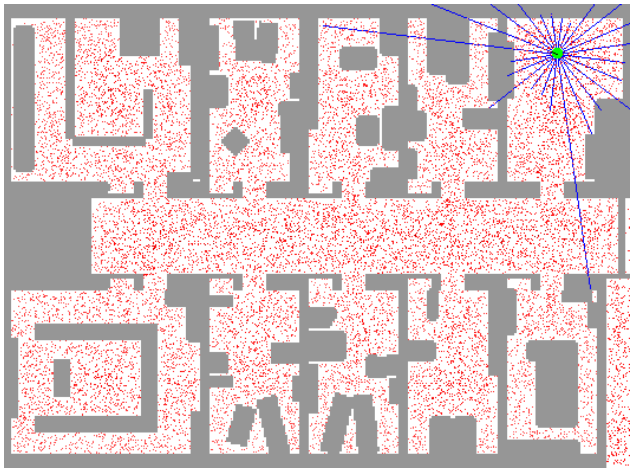
Importance Sampling

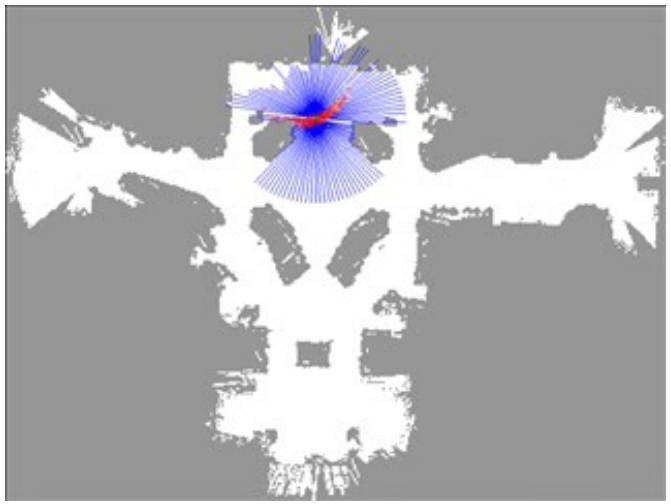
- The typical importance sampling framework:



Robot Localization

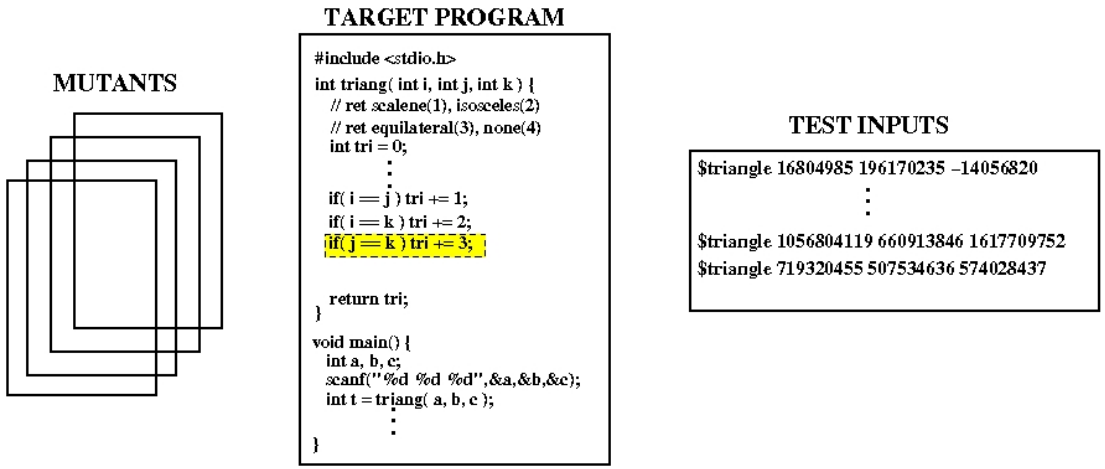
- Some video examples:





Illustrative Example: Mutation Testing

- *Mutation testing*: fault-based testing technique.
- Inject synthetic faults that are generated using well-defined mathematical transformations i.e., *mutation operators*.



- Mutants detected by test cases are considered *dead*, while those left unexposed are considered *alive*.

Mutation Testing Challenges

- The test cases are *adequate* if they produce different results on the original program and the faulty version, i.e., the *mutant*.
- Typically, the test suites are augmented to address mutants that remain unexposed.
- Cannot examine all possible mutants of all mutation operators.
- Reliable operation requires the exposure of all mutants.
- Focus attention on the *important* mutation operators whose mutants are more difficult to expose with the existing test suites.

Mutation Testing Challenges

- The test cases are *adequate* if they produce different results on the original program and the faulty version, i.e., the *mutant*.
- Typically, the test suites are augmented to address mutants that remain unexposed.
- Cannot examine all possible mutants of all mutation operators.
- Reliable operation requires the exposure of all mutants.
- Focus attention on the *important* mutation operators whose mutants are more difficult to expose with the existing test suites.

Mutation Testing Challenges

- The test cases are *adequate* if they produce different results on the original program and the faulty version, i.e., the *mutant*.
- Typically, the test suites are augmented to address mutants that remain unexposed.
- Cannot examine all possible mutants of all mutation operators.
- Reliable operation requires the exposure of all mutants.
- Focus attention on the *important* mutation operators whose mutants are more difficult to expose with the existing test suites.

Sampling-based Formulation

- Probability for each mutation operator: p_i for $\mu_i \forall i \in [1, N]$.
- Select initial (small) set of mutant samples of each operator, choosing *uniformly* or *proportional* to operator probabilities:

$$numMutantSamps_i^0 \simeq \begin{cases} c & \text{uniform} \\ \propto \frac{Nm_i}{NM} & \text{proportional} \end{cases} \quad (12)$$

Sampling-based Formulation

- Probability for each mutation operator: p_i for $\mu_i \forall i \in [1, N]$.
- Select initial (small) set of mutant samples of each operator, choosing *uniformly* or *proportional* to operator probabilities:

$$numMutantSamps_i^0 \simeq \begin{cases} c & \text{uniform} \\ \propto \frac{Nm_i}{NM} & \text{proportional} \end{cases} \quad (12)$$

Sampling Iterations

Iterate:

- Examine the ability of existing test suites to expose selected mutants.
- Increase probabilities of operators whose mutants are unexposed.

$$p_i^t = p_i^{t-1} + \frac{\delta p_i^t}{totalMutantSamps^t} \tag{13}$$

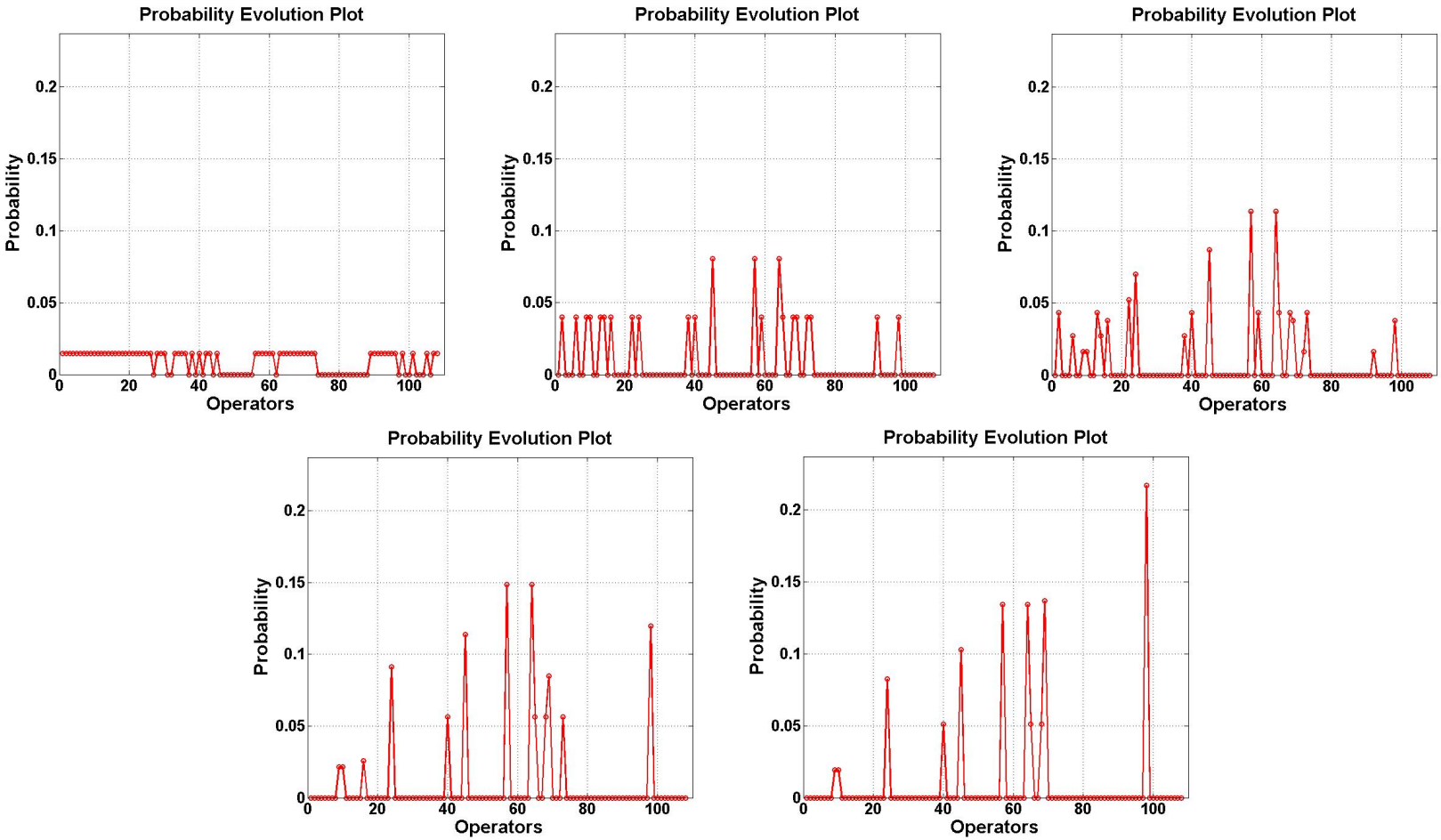
$$\delta p_i^t = -1.0 + 2.0 \frac{numAlive_i^t}{numMutantSamps_i^t} \quad : \in [-1.0, 1.0]$$

$$totalMutantSamps^t = \sum_{i=0}^{N-1} numMutantSamps_i^t$$

- Generate samples of each operator proportional to probabilities.

Probability Updates

Over a few iterations, sampling converges on operators whose mutants are difficult to expose: MATLAB results!



Innovations

- Sample without replacement: system is *stationary*.
- *Adapt* number of samples based on current uncertainty:

$$\mathcal{N}^{t+1} = \frac{1}{2\epsilon}\chi^2_{q^t-1,1-\delta} \tag{14}$$

$$\simeq \frac{q^t-1}{2\epsilon}\left\{1 - \frac{2}{9(q^t-1)} + \sqrt{\frac{2}{9(q^t-1)}}z_{1-\delta}\right\}^3$$

- Entropy in operator probability distribution:

$$E^t = -\sum_{j=0}^{N-1} p_j^t \cdot ln(p_j^t) \tag{15}$$

- *Terminate* when reduction in entropy is small:
  $E^t - E^{t-1} \leq \text{threshold}$

Innovations

- Sample without replacement: system is *stationary*.
- *Adapt* number of samples based on current uncertainty:

$$\mathcal{N}^{t+1} = \frac{1}{2\epsilon}\chi^2_{q^t-1,1-\delta} \tag{14}$$
$$\simeq \frac{q^t-1}{2\epsilon}\left\{1 - \frac{2}{9(q^t-1)} + \sqrt{\frac{2}{9(q^t-1)}}z_{1-\delta}\right\}^3$$

- Entropy in operator probability distribution:

$$E^t = -\sum_{j=0}^{N-1} p_j^t \cdot ln(p_j^t) \tag{15}$$

- *Terminate* when reduction in entropy is small:
$E^t - E^{t-1} \leq \text{threshold}$

Innovations

- Sample without replacement: system is *stationary*.
- *Adapt* number of samples based on current uncertainty:

$$\mathcal{N}^{t+1} = \frac{1}{2\epsilon}\chi^2_{q^t-1,1-\delta} \tag{14}$$
$$\simeq \frac{q^t-1}{2\epsilon}\left\{1 - \frac{2}{9(q^t-1)} + \sqrt{\frac{2}{9(q^t-1)}}z_{1-\delta}\right\}^3$$

- Entropy in operator probability distribution:

$$E^t = -\sum_{j=0}^{N-1} p_j^t \cdot ln(p_j^t) \tag{15}$$

- *Terminate* when reduction in entropy is small:
  $E^t - E^{t-1} \leq$ threshold

Innovations

- Sample without replacement: system is *stationary*.
- *Adapt* number of samples based on current uncertainty:

$$\mathcal{N}^{t+1} = \frac{1}{2\epsilon}\chi^2_{q^t-1,1-\delta} \tag{14}$$
$$\simeq \frac{q^t-1}{2\epsilon}\left\{1 - \frac{2}{9(q^t-1)} + \sqrt{\frac{2}{9(q^t-1)}}z_{1-\delta}\right\}^3$$

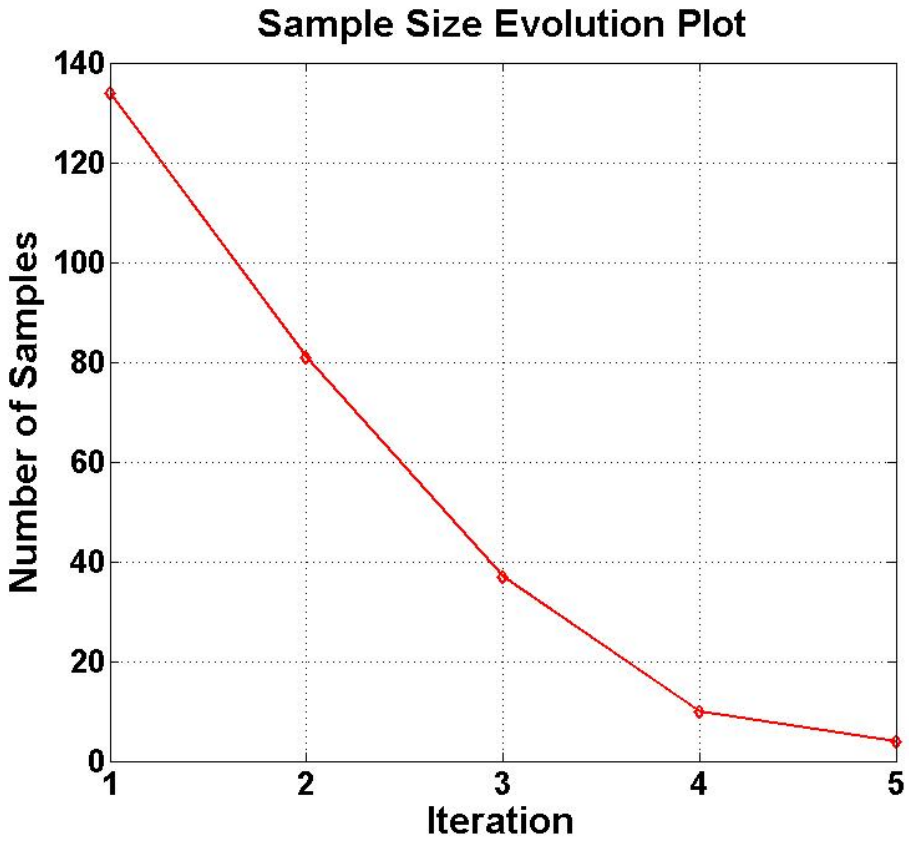
- Entropy in operator probability distribution:

$$E^t = -\sum_{j=0}^{N-1} p_j^t \cdot ln(p_j^t) \tag{15}$$

- *Terminate* when reduction in entropy is small:
  $E^t - E^{t-1} \leq \text{threshold}$

Samples Examined

- Entropy-based termination and adaptive sampling enables system to focus on important operators, while examining a small set of samples.



Summary

- Information theoretic measures an elegant way to encode information.
- Stochastic sampling ideal for tracking multiple hypotheses.
- Mutation testing used as the illustrative example.
- Adaptive sampling and information theoretic measures enable reliable and efficient program testing.
- Sampling is well-suited for many other software testing applications.

Summary

- Information theoretic measures an elegant way to encode information.
- Stochastic sampling ideal for tracking multiple hypotheses.
- Mutation testing used as the illustrative example.
- Adaptive sampling and information theoretic measures enable reliable and efficient program testing.
- Sampling is well-suited for many other software testing applications.

Summary

- Information theoretic measures an elegant way to encode information.
- Stochastic sampling ideal for tracking multiple hypotheses.
- Mutation testing used as the illustrative example.
- Adaptive sampling and information theoretic measures enable reliable and efficient program testing.
- Sampling is well-suited for many other software testing applications.

Summary

- Information theoretic measures an elegant way to encode information.
- Stochastic sampling ideal for tracking multiple hypotheses.

- Mutation testing used as the illustrative example.
- Adaptive sampling and information theoretic measures enable reliable and efficient program testing.

- Sampling is well-suited for many other software testing applications.

Summary

- Information theoretic measures an elegant way to encode information.
- Stochastic sampling ideal for tracking multiple hypotheses.

- Mutation testing used as the illustrative example.
- Adaptive sampling and information theoretic measures enable reliable and efficient program testing.

- Sampling is well-suited for many other software testing applications.

Other Sampling Methods

- First-order Markov chain: series of random variables $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(M)}$ that satisfy the first-order Markov property.

$$p(\boldsymbol{x}^{(m+1)}|\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}) = p(\boldsymbol{x}^{(m+1)}|\boldsymbol{x}^{(m)}) \qquad (16)$$

- Markov Chain Monte Carlo (MCMC) sampling.
- Gibbs sampling: MCMC algorithm that is a special case of the Metropolis-Hastings algorithm.
- Gibbs sampling updates random variables in a particular order: WinBUGS demo!

Other Sampling Methods

- First-order Markov chain: series of random variables $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(M)}$ that satisfy the first-order Markov property.

$$p(\boldsymbol{x}^{(m+1)}|\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}) = p(\boldsymbol{x}^{(m+1)}|\boldsymbol{x}^{(m)}) \qquad (16)$$

- Markov Chain Monte Carlo (MCMC) sampling.
- Gibbs sampling: MCMC algorithm that is a special case of the Metropolis-Hastings algorithm.
- Gibbs sampling updates random variables in a particular order: WinBUGS demo!

Other Sampling Methods

- First-order Markov chain: series of random variables $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(M)}$ that satisfy the first-order Markov property.

$$p(\boldsymbol{x}^{(m+1)}|\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}) = p(\boldsymbol{x}^{(m+1)}|\boldsymbol{x}^{(m)}) \qquad (16)$$

- Markov Chain Monte Carlo (MCMC) sampling.
- Gibbs sampling: MCMC algorithm that is a special case of the Metropolis-Hastings algorithm.
- Gibbs sampling updates random variables in a particular order: WinBUGS demo!

Other Sampling Methods

- First-order Markov chain: series of random variables $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(M)}$ that satisfy the first-order Markov property.

$$p(\boldsymbol{x}^{(m+1)}|\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}) = p(\boldsymbol{x}^{(m+1)}|\boldsymbol{x}^{(m)}) \qquad (16)$$

- Markov Chain Monte Carlo (MCMC) sampling.
- Gibbs sampling: MCMC algorithm that is a special case of the Metropolis-Hastings algorithm.
- Gibbs sampling updates random variables in a particular order: WinBUGS demo!

Further Reading


C. Bishop. *Pattern Recognition and Machine Learning*. Springer publishing house, 2007.


S. Thrun and W. Burgard and D. Fox. *Probabilistic Robotics*. MIT Press, 2005.


Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Second Edition, Wiley-Interscience, 2005.