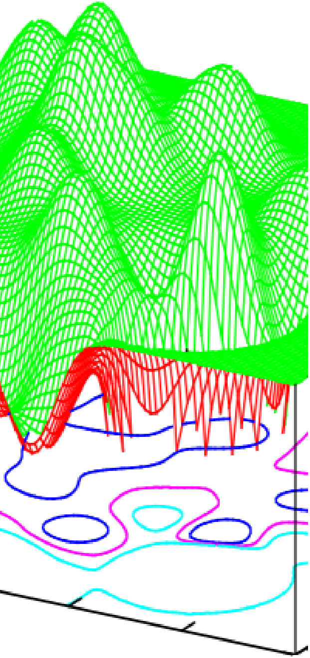


Machine Learning

Probability Basics

Marc Toussaint
University of Stuttgart

Summer 2016

The need for modelling

- Given a real world problem, translating it to a well-defined learning problem is non-trivial

- The "framework" of plain regression/classification is restricted: input x, output y.

- Graphical models (probabilstic models with multiple random variables and dependencies) are a more general framework for modelling "problems"; regression & classification become a special case; Reinforcement Learning, decision making, unsupervised learning, but also language processing, image segmentation, can be represented.

Thomas Bayes (1702-1761)

~~Rev. T. Bayes~~

"Essay Towards Solving a Problem in the Doctrine of Chances"

- Addresses problem of *inverse probabilities*:
  Knowing the conditional probability of B given A, what is the conditional probability of A given B?

- Example:
  40% Bavarians speak dialect, only 1% of non-Bavarians speak (Bav.) dialect
  Given a random German that speaks non-dialect, is he Bavarian?
  (15% of Germans are Bavarian)

Inference

- "Inference" = Given some pieces of information (prior, observed variabes) what is the implication (the implied information, the posterior) on a non-observed variable

- **Learning as Inference:**
  - given pieces of information: data, assumed model, *prior* over β
  - non-observed variable: β

Probability Theory

- Why do we need probabilities?
  - Obvious: to express inherent stochasticity of the world (data)

- But beyond this: (also in a "deterministic world"):
  - lack of knowledge!
  - hidden (latent) variables
  - expressing *uncertainty*
  - expressing *information* (and lack of information)

- Probability Theory: an information calculus

Probability: Frequentist and Bayesian

- Frequentist probabilities are defined in the limit of an infinite number of trials
  Example: “The probability of a particular coin landing heads up is 0.43”

- Bayesian (subjective) probabilities quantify degrees of belief
  Example: “The probability of rain tomorrow is 0.3” – not possible to repeat “tomorrow”

Outline

- Basic definitions
  - Random variables
  - joint, conditional, marginal distribution
  - Bayes' theorem
- Examples for Bayes
- Probability distributions [skipped, only Gauss]
  - Binomial; Beta
  - Multinomial; Dirichlet
  - Conjugate priors
  - Gauss; Wichart
  - Student-t, Dirak, Particles
- Monte Carlo, MCMC [skipped]

  These are generic slides on probabilities I use throughout my lecture. Only parts are mandatory for the AI course.

Basic definitions

Probabilities & Random Variables

- For a random variable X with discrete domain $\mathrm{dom}(X) = \Omega$ we write:
  $\forall_{x\in\Omega}:\ 0 \le P(X=x) \le 1$
  $\sum_{x\in\Omega} P(X=x) = 1$

  Example: A dice can take values $\Omega = \{1, .., 6\}$.
  X is the random variable of a dice throw.
  $P(X=1) \in [0, 1]$ is the probability that X takes value 1.

- A bit more formally: a random variable is a map from a measureable space to a domain (sample space) and thereby introduces a probability measure on the domain ("assigns a probability to each possible value")

Probabilty Distributions

- $P(X=1) \in \mathbb{R}$ denotes a specific probability
  $P(X)$ denotes the probability distribution (function over Ω)

Probabilty Distributions

- $P(X=1) \in \mathbb{R}$ denotes a specific probability
  $P(X)$ denotes the probability distribution (function over Ω)

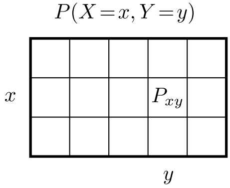
  Example: A dice can take values $\Omega = \{1,2,3,4,5,6\}$.
  By $P(X)$ we discribe the full distribution over possible values $\{1,..,6\}$. These are 6 numbers that sum to one, usually stored in a *table*, e.g.: $[\frac{1}{6},\frac{1}{6},\frac{1}{6},\frac{1}{6},\frac{1}{6},\frac{1}{6}]$

- In implementations we typically represent distributions over discrete random variables as tables (arrays) of numbers

- Notation for summing over a RV:
  In equation we often need to sum over RVs. We then write
  $$\sum_X P(X) \; \cdots$$
  as shorthand for the explicit notation $\sum_{x \in \mathrm{dom}(X)} P(X=x) \; \cdots$

Joint distributions

Assume we have *two* random variables X and Y



- Definitions:
  Joint: $P(X, Y)$
  Marginal: $P(X) = \sum_Y P(X, Y)$
  Conditional: $P(X|Y) = \frac{P(X,Y)}{P(Y)}$

  The conditional is normalized: $\forall_Y : \sum_X P(X|Y) = 1$

- X is *independent* of Y iff: $P(X|Y) = P(X)$
  (table thinking: all columns of $P(X|Y)$ are equal)

Joint distributions

joint: $P(X,Y)$
marginal: $P(X) = \sum_Y P(X,Y)$
conditional: $P(X|Y) = \frac{P(X,Y)}{P(Y)}$

- Implications of these definitions:
  Product rule: $P(X,Y) = P(X|Y)\ P(Y) = P(Y|X)\ P(X)$

  Bayes' Theorem: $P(X|Y) = \frac{P(Y|X)\ P(X)}{P(Y)}$

Bayes' Theorem

$$P(X|Y) = \frac{P(Y|X)\ P(X)}{P(Y)}$$

$$\text{posterior} \;=\; \frac{\text{likelihood} \cdot \text{prior}}{\text{normalization}}$$

Multiple RVs:

- Analogously for n random variables $X_{1:n}$ (stored as a rank n tensor)
  Joint: $P(X_{1:n})$
  Marginal: $P(X_1) = \sum_{X_{2:n}} P(X_{1:n})$,
  Conditional: $P(X_1|X_{2:n}) = \frac{P(X_{1:n})}{P(X_{2:n})}$

- X is *conditionally independent* of Y given Z iff:
  $$P(X|Y,Z) = P(X|Z)$$

- Product rule and Bayes' Theorem:

$$P(X_{1:n}) = \prod_{i=1}^n P(X_i|X_{i+1:n})$$

$$P(X_1|X_{2:n}) = \frac{P(X_2|X_1, X_{3:n})\ P(X_1|X_{3:n})}{P(X_2|X_{3:n})}$$

$$P(X,Z,Y) = P(X|Y,Z)\ P(Y|Z)\ P(Z)$$

$$P(X|Y,Z) = \frac{P(Y|X,Z)\ P(X|Z)}{P(Y|Z)}$$

$$P(X,Y|Z) = \frac{P(X,Z|Y)\ P(Y)}{P(Z)}$$

Example 1: Bavarian dialect

- 40% Bavarians speak dialect, only 1% of non-Bavarians speak (Bav.) dialect
  Given a random German that speaks non-dialect, is he Bavarian?
  (15% of Germans are Bavarian)

  $P(D=1 \mid B=1) = 0.4$
  $P(D=1 \mid B=0) = 0.01$
  $P(B=1) = 0.15$

Example 1: Bavarian dialect

- 40% Bavarians speak dialect, only 1% of non-Bavarians speak (Bav.) dialect
  Given a random German that speaks non-dialect, is he Bavarian?
  (15% of Germans are Bavarian)

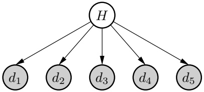
  $$P(D=1 \mid B=1) = 0.4$$
  $$P(D=1 \mid B=0) = 0.01$$
  $$P(B=1) = 0.15$$

  If follows
  $$P(B=1 \mid D=0) = \frac{P(D=0 \mid B=1)\ P(B=1)}{P(D=0)} = \frac{.6 \cdot .15}{.6 \cdot .15 + 0.99 \cdot .85} \approx 0.097$$

Example 2: Coin flipping

HHTHT



HHHHH

- What process produces these sequences?

- We compare two hypothesis:
  $H = 1$: fair coin $\quad P(d_i\!=\!\mathtt{H} \mid H\!=\!1) = \frac{1}{2}$
  $H = 2$: always heads coin $\quad P(d_i\!=\!\mathtt{H} \mid H\!=\!2) = 1$

- Bayes' theorem:
  $$P(H \mid D) = \frac{P(D \mid H)P(H)}{P(D)}$$

Coin flipping

$D = \texttt{HHTHT}$

$$P(D \mid H\!=\!1) = 1/2^5 \qquad\qquad P(H\!=\!1) = \tfrac{999}{1000}$$

$$P(D \mid H\!=\!2) = 0 \qquad\qquad P(H\!=\!2) = \tfrac{1}{1000}$$

$$\frac{P(H\!=\!1 \mid D)}{P(H\!=\!2 \mid D)} = \frac{P(D \mid H\!=\!1)}{P(D \mid H\!=\!2)}\,\frac{P(H\!=\!1)}{P(H\!=\!2)} = \frac{1/32}{0}\,\frac{999}{1} = \infty$$

Coin flipping

$D = \texttt{HHHHH}$

$$P(D \mid H{=}1) = 1/2^5 \qquad\qquad P(H{=}1) = \tfrac{999}{1000}$$

$$P(D \mid H{=}2) = 1 \qquad\qquad P(H{=}2) = \tfrac{1}{1000}$$

$$\frac{P(H{=}1 \mid D)}{P(H{=}2 \mid D)} = \frac{P(D \mid H{=}1)}{P(D \mid H{=}2)} \; \frac{P(H{=}1)}{P(H{=}2)} = \frac{1/32}{1} \; \frac{999}{1} \approx 30$$

Coin flipping

$D = \mathtt{HHHHHHHHHH}$

$$P(D \mid H{=}1) = 1/2^{10} \qquad P(H{=}1) = \tfrac{999}{1000}$$

$$P(D \mid H{=}2) = 1 \qquad P(H{=}2) = \tfrac{1}{1000}$$

$$\frac{P(H{=}1 \mid D)}{P(H{=}2 \mid D)} = \frac{P(D \mid H{=}1)}{P(D \mid H{=}2)}\,\frac{P(H{=}1)}{P(H{=}2)} = \frac{1/1024}{1}\,\frac{999}{1} \approx 1$$

Learning as Bayesian inference

$$P(\text{World}|\text{Data}) = \frac{P(\text{Data}|\text{World})\ P(\text{World})}{P(\text{Data})}$$

$P(\text{World})$ describes our prior over all possible worlds. Learning means to infer about the world we live in based on the data we have!

Learning as Bayesian inference

$$P(\text{World}|\text{Data}) = \frac{P(\text{Data}|\text{World})\ P(\text{World})}{P(\text{Data})}$$

$P(\text{World})$ describes our prior over all possible worlds. Learning means to infer about the world we live in based on the data we have!

- In the context of regression, the "world" is the function $f(x)$

$$P(f|\text{Data}) = \frac{P(\text{Data}|f)\ P(f)}{P(\text{Data})}$$

$P(f)$ describes our prior over possible functions

Regression means to infer the function based on the data we have

Probability distributions

recommended reference: Bishop.: *Pattern Recognition and Machine Learning*

Bernoulli & Binomial

- We have a binary random variable $x \in \{0, 1\}$ (i.e. $\mathrm{dom}(x) = \{0, 1\}$)
  The *Bernoulli* distribution is parameterized by a single scalar μ,

$$P(x{=}1 \,|\, \mu) = \mu \;, \quad P(x{=}0 \,|\, \mu) = 1 - \mu$$
$$\mathrm{Bern}(x \,|\, \mu) = \mu^x (1-\mu)^{1-x}$$

- We have a data set of random variables $D = \{x_1, .., x_n\}$, each $x_i \in \{0, 1\}$. If each $x_i \sim \mathrm{Bern}(x_i \,|\, \mu)$ we have

$$P(D \,|\, \mu) = \textstyle\prod_{i=1}^n \mathrm{Bern}(x_i \,|\, \mu) = \prod_{i=1}^n \mu^{x_i} (1-\mu)^{1-x_i}$$

$$\underset{\mu}{\mathrm{argmax}} \;\log P(D \,|\, \mu) = \underset{\mu}{\mathrm{argmax}} \;\sum_{i=1}^n x_i \log \mu + (1-x_i) \log(1-\mu) = \frac{1}{n} \sum_{i=1}^n x_i$$

- The *Binomial distribution* is the distribution over the count $m = \sum_{i=1}^n x_i$

$$\mathrm{Bin}(m \,|\, n, \mu) = \binom{n}{m} \mu^m (1-\mu)^{n-m} \;, \quad \binom{n}{m} = \frac{n!}{(n-m)!\; m!}$$

Beta

How to express uncertainty over a Bernoulli parameter μ

- The *Beta* distribution is over the interval $[0, 1]$, typically the parameter μ of a Bernoulli:

$$\text{Beta}(\mu \mid a, b) = \frac{1}{B(a, b)}\ \mu^{a-1}(1-\mu)^{b-1}$$

with mean $\langle\mu\rangle = \frac{a}{a+b}$ and mode $\mu^* = \frac{a-1}{a+b-2}$ for $a, b > 1$

- The crucial point is:
  - Assume we are in a world with a "Bernoulli source" (e.g., binary bandit), but don't know its parameter μ
  - Assume we have a *prior* distribution $P(\mu) = \text{Beta}(\mu \mid a, b)$
  - Assume we collected some data $D = \{x_1, .., x_n\}$, $x_i \in \{0, 1\}$, with counts $a_D = \sum_i x_i$ of $[x_i\!=\!1]$ and $b_D = \sum_i (1 - x_i)$ of $[x_i\!=\!0]$
  - The posterior is

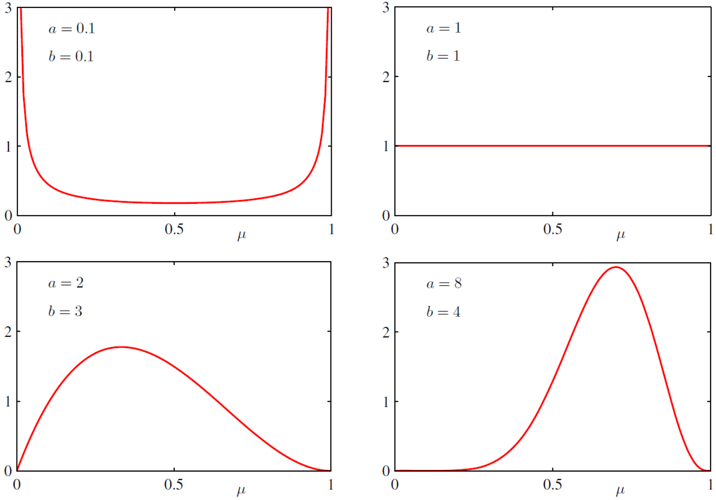
$$\begin{aligned} P(\mu \mid D) &= \frac{P(D \mid \mu)}{P(D)}\ P(\mu) \propto \text{Bin}(D \mid \mu)\ \text{Beta}(\mu \mid a, b) \\ &\propto \mu^{a_D}(1-\mu)^{b_D}\ \mu^{a-1}(1-\mu)^{b-1} = \mu^{a-1+a_D}(1-\mu)^{b-1+b_D} \\ &= \text{Beta}(\mu \mid a + a_D, b + b_D) \end{aligned}$$

Beta

The prior is $\mathrm{Beta}(\mu \mid a, b)$*, the posterior is* $\mathrm{Beta}(\mu \mid a + a_D, b + b_D)$

- Conclusions:
  - The semantics of a and b are counts of $[x_i = 1]$ and $[x_i = 0]$, respectively
  - The Beta distribution is conjugate to the Bernoulli (explained later)
  - With the Beta distribution we can represent beliefs (state of knowledge) about uncertain $\mu \in [0, 1]$ and know how to update this belief given data

Beta



from Bishop

Multinomial

- We have an integer random variable $x \in \{1, .., K\}$
  The probability of a single x can be parameterized by $\mu = (\mu_1, .., \mu_K)$:

$$P(x{=}k \mid \mu) = \mu_k$$

  with the constraint $\sum_{k=1}^K \mu_k = 1$ (probabilities need to be normalized)

- We have a data set of random variables $D = \{x_1, .., x_n\}$, each $x_i \in \{1, .., K\}$. If each $x_i \sim P(x_i \mid \mu)$ we have

$$P(D \mid \mu) = \prod_{i=1}^n \mu_{x_i} = \prod_{i=1}^n \prod_{k=1}^K \mu_k^{[x_i=k]} = \prod_{k=1}^K \mu_k^{m_k}$$

  where $m_k = \sum_{i=1}^n [x_i{=}k]$ is the count of $[x_i{=}k]$. The ML estimator is

$$\operatorname*{argmax}_{\mu} \; \log P(D \mid \mu) = \frac{1}{n}(m_1, .., m_K)$$

- The *Multinomial distribution* is this distribution over the counts m_k

$$\mathrm{Mult}(m_1, .., m_K \mid n, \mu) \propto \prod_{k=1}^K \mu_k^{m_k}$$

Dirichlet

How to express uncertainty over a Multinomial parameter μ

- The *Dirichlet* distribution is over the K-simplex, that is, over $\mu_1, .., \mu_K \in [0,1]$ subject to the constraint $\sum_{k=1}^K \mu_k = 1$:

$$\mathrm{Dir}(\mu \,|\, \alpha) \propto \ \prod_{k=1}^K \mu_k^{\alpha_k - 1}$$

  It is parameterized by $\alpha = (\alpha_1, .., \alpha_K)$, has mean $\langle \mu_i \rangle = \frac{\alpha_i}{\sum_j \alpha_j}$ and mode $\mu_i^* = \frac{\alpha_i - 1}{\sum_j \alpha_j - K}$ for $a_i > 1$.

- The crucial point is:
  - Assume we are in a world with a “Multinomial source” (e.g., an integer bandit), but don’t know its parameter μ
  - Assume we have a *prior* distribution $P(\mu) = \mathrm{Dir}(\mu \,|\, \alpha)$
  - Assume we collected some data $D = \{x_1, .., x_n\}$, $x_i \in \{1, .., K\}$, with counts $m_k = \sum_i [x_i = k]$
  - The posterior is

$$P(\mu \,|\, D) = \frac{P(D \,|\, \mu)}{P(D)} \ P(\mu) \propto \mathrm{Mult}(D \,|\, \mu) \ \mathrm{Dir}(\mu \,|\, a, b)$$

$$\propto \prod_{k=1}^K \mu_k^{m_k} \ \prod_{k=1}^K \mu_k^{\alpha_k - 1} = \prod_{k=1}^K \mu_k^{\alpha_k - 1 + m_k}$$

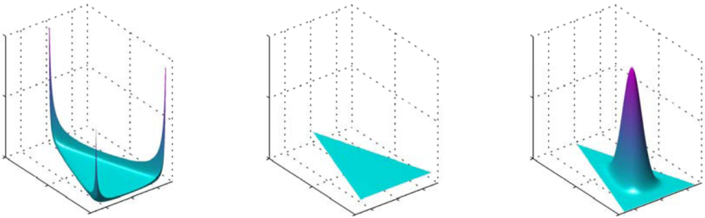
Dirichlet

The prior is $\mathrm{Dir}(\mu \mid \alpha)$*, the posterior is* $\mathrm{Dir}(\mu \mid \alpha + m)$

- Conclusions:
  - The semantics of α is the counts of $[x_i = k]$
  - The Dirichlet distribution is conjugate to the Multinomial
  - With the Dirichlet distribution we can represent beliefs (state of knowledge) about uncertain μ of an integer random variable and know how to update this belief given data

Dirichlet

Illustrations for $\alpha = (0.1, 0.1, 0.1)$, $\alpha = (1, 1, 1)$ and $\alpha = (10, 10, 10)$:



from Bishop

Motivation for Beta & Dirichlet distributions

- Bandits:
  - If we have binary [integer] bandits, the Beta [Dirichlet] distribution is a way to represent and update beliefs
  - The belief space becomes discrete: The parameter α of the prior is continuous, but the posterior updates live on a discrete "grid" (adding counts to α)
  - We can in principle do belief planning using this
- Reinforcement Learning:
  - Assume we know that the world is a finite-state MDP, but do not know its transition probability $P(s' \mid s, a)$. For each (s, a), $P(s' \mid s, a)$ is a distribution over the integer s'
  - Having a separate Dirichlet distribution for each (s, a) is a way to represent our belief about the world, that is, our belief about $P(s' \mid s, a)$
  - We can in principle do belief planning using this $\rightarrow$ *Bayesian Reinforcement Learning*
- Dirichlet distributions are also used to model texts (word distributions in text), images, or mixture distributions in general

Conjugate priors

- Assume you have data $D = \{x_1, .., x_n\}$ with likelihood

$$P(D \,|\, \theta)$$

  that depends on an uncertain parameter θ
  Assume you have a prior $P(\theta)$

- The prior $P(\theta)$ is **conjugate** to the likelihood $P(D \,|\, \theta)$ iff the posterior

$$P(\theta \,|\, D) \propto P(D \,|\, \theta) \; P(\theta)$$

  is in the *same distribution class* as the prior $P(\theta)$

- Having a conjugate prior is very convenient, because then you know how to update the belief given data

Conjugate priors

likelihood	conjugate
Binomial $\mathrm{Bin}(D \mid \mu)$	Beta $\mathrm{Beta}(\mu \mid a, b)$
Multinomial $\mathrm{Mult}(D \mid \mu)$	Dirichlet $\mathrm{Dir}(\mu \mid \alpha)$
Gauss $\mathcal{N}(x \mid \mu, \Sigma)$	Gauss $\mathcal{N}(\mu \mid \mu_0, A)$
1D Gauss $\mathcal{N}(x \mid \mu, \lambda^{-1})$	Gamma $\mathrm{Gam}(\lambda \mid a, b)$
nD Gauss $\mathcal{N}(x \mid \mu, \Lambda^{-1})$	Wishart $\mathrm{Wish}(\Lambda \mid W, \nu)$
nD Gauss $\mathcal{N}(x \mid \mu, \Lambda^{-1})$	Gauss-Wishart $\mathcal{N}(\mu \mid \mu_0, (\beta\Lambda)^{-1})\ \mathrm{Wish}(\Lambda \mid W, \nu)$

Distributions over continuous domain

- Let x be a continuous RV. The **probability density function (pdf)** $p(x) \in [0, \infty)$ defines the probability

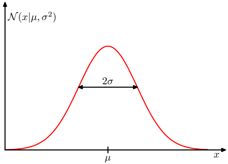
$$P(a \le x \le b) = \int_a^b p(x)\ dx \ \in [0,1]$$

The **(cumulative) probability distribution** $F(y) = P(x \le y) = \int_{-\infty}^y dx\ p(x) \in [0,1]$ is the cumulative integral with $\lim_{y \to \infty} F(y) = 1$

(In discrete domain: *probability distribution* and *probability mass function* $P(x) \in [0,1]$ are used synonymously.)

- Two basic examples:
Gaussian: $\mathcal{N}(x \,|\, \mu, \Sigma) = \frac{1}{|\, 2\pi\Sigma \,|^{\,1/2}}\ e^{-\frac{1}{2}(x-\mu)^\top\ \Sigma^{-1}\ (x-\mu)}$
Dirac or δ ("point particle") $\delta(x) = 0$ except at $x = 0$, $\int \delta(x)\ dx = 1$
$\delta(x) = \frac{\partial}{\partial x} H(x)$ where $H(x) = [x \ge 0] =$ Heavyside step function

Gaussian distribution



- 1-dim: $\mathcal{N}(x\,|\,\mu,\sigma^2) = \frac{1}{|\,2\pi\sigma^2\,|^{\,1/2}}\; e^{-\frac{1}{2}(x-\mu)^2/\sigma^2}$
- n-dim Gaussian in *normal form*:

$$\mathcal{N}(x\,|\,\mu,\Sigma) = \frac{1}{|\,2\pi\Sigma\,|^{\,1/2}}\;\exp\{-\frac{1}{2}(x-\mu)^\top\,\Sigma^{\text{-}1}\,(x-\mu)\}$$

with **mean** μ and **covariance** matrix Σ. In *canonical form*:

$$\mathcal{N}[x\,|\,a,A] = \frac{\exp\{-\frac{1}{2}a^\top A^{\text{-}1} a\}}{|\,2\pi A^{\text{-}1}\,|^{\,1/2}}\;\exp\{-\frac{1}{2}x^\top\,A\,x + x^\top a\} \tag{1}$$

with **precision** matrix $A = \Sigma^{\text{-}1}$ and coefficient $a = \Sigma^{\text{-}1}\mu$ (and mean $\mu = A^{\text{-}1}a$).

- Gaussian identities: see

`http://ipvs.informatik.uni-stuttgart.de/mlr/marc/notes/gaussians.pdf`

Gaussian identities

Symmetry: $\mathcal{N}(x \,|\, a, A) = \mathcal{N}(a \,|\, x, A) = \mathcal{N}(x - a \,|\, 0, A)$

Product:

$$\mathcal{N}(x \,|\, a, A)\; \mathcal{N}(x \,|\, b, B) = \mathcal{N}[x \,|\, A^{\text{-}1}a + B^{\text{-}1}b, A^{\text{-}1} + B^{\text{-}1}]\; \mathcal{N}(a \,|\, b, A + B)$$
$$\mathcal{N}[x \,|\, a, A]\; \mathcal{N}[x \,|\, b, B] = \mathcal{N}[x \,|\, a + b, A + B]\; \mathcal{N}(A^{\text{-}1}a \,|\, B^{\text{-}1}b, A^{\text{-}1} + B^{\text{-}1})$$

"Propagation":

$$\int_y \mathcal{N}(x \,|\, a + Fy, A)\; \mathcal{N}(y \,|\, b, B)\; dy = \mathcal{N}(x \,|\, a + Fb, A + FBF^\top)$$

Transformation:

$$\mathcal{N}(Fx + f \,|\, a, A) = \tfrac{1}{|\,F\,|}\; \mathcal{N}(x \,|\; F^{\text{-}1}(a - f),\; F^{\text{-}1}AF^{\text{-}\top})$$

Marginal & conditional:

$$\mathcal{N}\left(\begin{array}{c|c} x & a \\ y & b \end{array}, \begin{array}{cc} A & C \\ C^\top & B \end{array}\right) = \mathcal{N}(x \,|\, a, A) \cdot \mathcal{N}(y \,|\, b + C^\top A^{\text{-}1}(x \text{-} a),\; B - C^\top A^{\text{-}1}C)$$

More Gaussian identities: see

`http://ipvs.informatik.uni-stuttgart.de/mlr/marc/notes/gaussians.pdf`

Gaussian prior and posterior

- Assume we have data $D = \{x_1, .., x_n\}$, each $x_i \in \mathbb{R}^n$, with likelihood

$$P(D \mid \mu, \Sigma) = \textstyle\prod_i \mathcal{N}(x_i \mid \mu, \Sigma)$$

$$\operatorname*{argmax}_{\mu} \; P(D \mid \mu, \Sigma) = \frac{1}{n} \sum_{i=1}^{n} x_i$$

$$\operatorname*{argmax}_{\Sigma} \; P(D \mid \mu, \Sigma) = \frac{1}{n} \sum_{i=1}^{n} (x_i - \mu)(x_i - \mu)^\top$$

- Assume we are initially uncertain about μ (but know Σ). We can express this uncertainty using again a Gaussian $\mathcal{N}[\mu \mid a, A]$. Given data we have

$$\begin{aligned} P(\mu \mid D) &\propto P(D \mid \mu, \Sigma) \; P(\mu) = \textstyle\prod_i \mathcal{N}(x_i \mid \mu, \Sigma) \; \mathcal{N}[\mu \mid a, A] \\ &= \textstyle\prod_i \mathcal{N}[\mu \mid \Sigma^{-1} x_i, \Sigma^{-1}] \; \mathcal{N}[\mu \mid a, A] \propto \mathcal{N}[\mu \mid \Sigma^{-1} \sum_i x_i, \; n\Sigma^{-1} + A] \end{aligned}$$

  Note: in the limit $A \to 0$ (uninformative prior) this becomes

$$P(\mu \mid D) = \mathcal{N}(\mu \mid \frac{1}{n} \sum_i x_i, \frac{1}{n}\Sigma)$$

  which is consistent with the Maximum Likelihood estimator

Motivation for Gaussian distributions

- Gaussian Bandits
- Control theory, Stochastic Optimal Control
- State estimation, sensor processing, Gaussian filtering (Kalman filtering)
- Machine Learning
- etc

Particle Approximation of a Distribution

- We approximate a distribution $p(x)$ over a continuous domain $\mathbb{R}^n$
- A particle distribution $q(x)$ is a weighed set $\mathcal{S} = \{(x^i, w^i)\}_{i=1}^N$ of N particles
  - each particle has a "location" $x^i \in \mathbb{R}^n$ and a weight $w^i \in \mathbb{R}$
  - weights are normalized, $\sum_i w^i = 1$

$$q(x) := \sum_{i=1}^N w^i \; \delta(x - x^i)$$

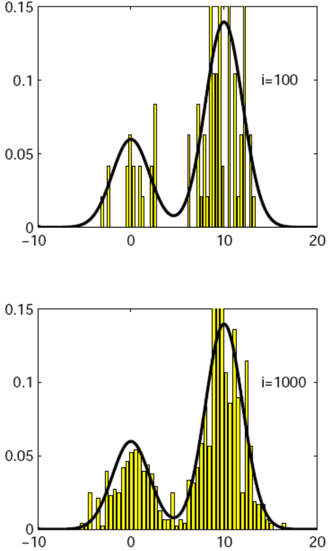
  where $\delta(x - x^i)$ is the δ-distribution.
- Given weighted particles, we can estimate for any (smooth) f:

$$\langle f(x) \rangle_p = \int_x f(x) p(x) dx \;\approx\; \textstyle\sum_{i=1}^N w^i f(x^i)$$

  See *An Introduction to MCMC for Machine Learning*
  `www.cs.ubc.ca/~nando/papers/mlintro.pdf`

Particle Approximation of a Distribution

Histogram of a particle representation:



Motivation for particle distributions

- Numeric representation of “difficult” distributions
  - Very general and versatile
  - But often needs many samples
- Distributions over games (action sequences), sample based planning, MCTS
- State estimation, particle filters
- etc

Utilities & Decision Theory

- Given a space of events Ω (e.g., outcomes of a trial, a game, etc) the utility is a function
$$U:\ \Omega \to \mathbb{R}$$
- The utility represents preferences as a single scalar – which is not always obvious   (cf. multi-objective optimization)
- *Decision Theory* making decisions (that determine $p(x)$) that maximize expected utility
$$\mathsf{E}\{U\}_p = \int_x U(x)\ p(x)$$
- Concave utility functions imply risk aversion (and convex, risk-taking)

Entropy

- The neg-log $(-\log p(x))$ of a distribution reflects something like "error":
  - neg-log of a Guassian $\leftrightarrow$ squared error
  - neg-log likelihood $\leftrightarrow$ prediction error

- The $(-\log p(x))$ is the "optimal" coding length you should assign to a symbol x. This will minimize the expected length of an encoding

$$H(p) = \int_x p(x)[-\log p(x)]$$

- The **entropy** $H(p) = \mathsf{E}_{p(x)}\{-\log p(x)\}$ of a distribution p is a measure of uncertainty, or lack-of-information, we have about x

Kullback-Leibler divergence

- Assume you use a "wrong" distribution $q(x)$ to decide on the coding length of symbols drawn from $p(x)$. The expected length of a encoding is

$$\int_x p(x)[-\log q(x)] \geq H(p)$$

- The difference

$$D\big(p \,\|\, q\big) = \int_x p(x) \log \frac{p(x)}{q(x)} \geq 0$$

  is called Kullback-Leibler divergence

  Proof of inequality, using the Jenson inequality:

$$-\int_x p(x) \log \frac{q(x)}{p(x)} \geq -\log \int_x p(x) \frac{q(x)}{p(x)} = 0$$

Monte Carlo methods

- Generally, a Monte Carlo method is a method to generate a set of (potentially weighted) samples that approximate a distribution $p(x)$.
In the unweighted case, the samples should be i.i.d. $x_i \sim p(x)$
In the general (also weighted) case, we want particles that allow to estimate expectations of anything that depends on x, e.g. $f(x)$:

$$\lim_{N\to\infty} \langle f(x) \rangle_q = \lim_{N\to\infty} \sum_{i=1}^{N} w^i f(x^i) = \int_x f(x)\, p(x)\, dx = \langle f(x) \rangle_p$$

In this view, Monte Carlo methods approximate an integral.
- Motivation: $p(x)$ itself is too complicated to express analytically or compute $\langle f(x) \rangle_p$ directly

- Example: What is the probability that a solitair would come out successful? (Original story by Stan Ulam.) Instead of trying to analytically compute this, generate many random solitairs and count.
- Naming: The method developed in the 40ies, where computers became faster. Fermi, Ulam and von Neumann initiated the idea. von Neumann called it "Monte Carlo" as a code name.

Rejection Sampling

- How can we generate i.i.d. samples $x_i \sim p(x)$?
- Assumptions:
  - We can sample $x \sim q(x)$ from a simpler distribution $q(x)$ (e.g., uniform), called **proposal distribution**
  - We can numerically evaluate $p(x)$ for a specific x (even if we don't have an analytic expression of $p(x)$)
  - There exists M such that $\forall_x : p(x) \leq Mq(x)$ (which implies q has larger or equal support as p)

- Rejection Sampling:
  - Sample a candiate $x \sim q(x)$
  - With probability $\frac{p(x)}{Mq(x)}$ accept x and add to $\mathcal{S}$; otherwise reject
  - Repeat until $|\mathcal{S}| = n$
- This generates an unweighted sample set $\mathcal{S}$ to approximate $p(x)$

Importance sampling

- Assumptions:
  - We can sample $x \sim q(x)$ from a simpler distribution $q(x)$ (e.g., uniform)
  - We can numerically evaluate $p(x)$ for a specific x (even if we don't have an analytic expression of $p(x)$)
- Importance Sampling:
  - Sample a candiate $x \sim q(x)$
  - Add the weighted sample $(x, \frac{p(x)}{q(x)})$ to $\mathcal{S}$
  - Repeat n times
- This generates an weighted sample set $\mathcal{S}$ to approximate $p(x)$
  The weights $w_i = \frac{p(x_i)}{q(x_i)}$ are called **importance weights**
- Crucial for efficiency: a good choice of the proposal $q(x)$

Applications

- MCTS can be viewed as estimating a distribution over games (action sequences) conditional to win
- Inference in graphical models (models involving many depending random variables)

Some more continuous distributions*

Gaussian $\mathcal{N}(x \mid a, A) = \frac{1}{\mid 2\pi A \mid^{1/2}}\ e^{-\frac{1}{2}(x-a)^\top\ A^{-1}\ (x-a}$

Dirac or δ $\delta(x) = \frac{\partial}{\partial x} H(x)$

Student's t $p(x;\nu) \propto [1 + \frac{x^2}{\nu}]^{-\frac{\nu+1}{2}}$
(=Gaussian for $\nu \to \infty$, otherwise heavy tails)

Exponential $p(x;\lambda) = [x \ge 0]\ \lambda e^{-\lambda x}$
(distribution over single event time)

Laplace $p(x;\mu,b) = \frac{1}{2b} e^{-\mid x-\mu \mid /b}$
("double exponential")

Chi-squared $p(x;k) \propto [x \ge 0]\ x^{k/2-1} e^{-x/2}$

Gamma $p(x;k,\theta) \propto [x \ge 0]\ x^{k-1} e^{-x/\theta}$