Module 2: Fundamentals of Probability Theory, Decision Theory, and Information Theory

Asst. Prof. Girish Chowdhary,
Univeristy of Illinois at Urbana Champaign

February 6, 2018

Outline

Outline

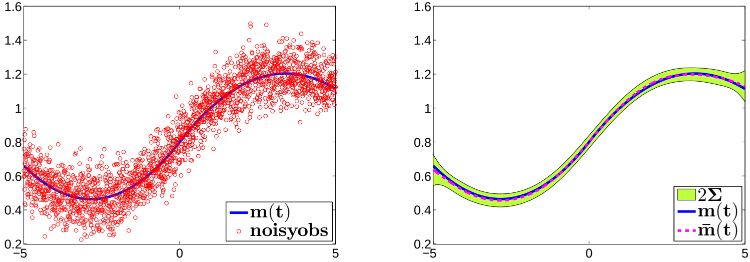
Reading:

- Chapter 1 from Russell and Norvig (skim quickly)
- Chapter 2 and 3 from Murphy (read)

- Why should we worry about probability in autonomy?
- Quick review of probability theory
- A quick primer on decision theory
- Elements of information theory
- What are generative models, how are they different from discriminative models
- What is a Kalman Filter

- Sensing the world: Perception
- Representing Knowledge: Machine learning
- Making decisions: Planning and Control
- Executing decisions and interacting with the world: control

- Finding patterns in data
- Building models from data $\rightarrow$ being able to predict patterns
- Deterministic vs Probabilistic models



(a) noisy data (b) Probablistic model

Figure:

- Consider that we are given a noisy data set $S = \{s_1, s_2, ..., s_N\}$
- Goal: Fit a curve for the given data: simple case: polynomial fit

$$y(x, w) = w_0 + w_1 x + w_2 x^2 + ... + w_M x^M = \sum_{j=0}^{M} w_j x^j$$

- can write this in vector form: $y(x, w) = [1 \ x \ x^2 \ x^3 \ ... \ x^M]^T W$, where $W \in \Re^M$ is a column vector of weights
- We can define a least squares error function:

$$E(w) = \frac{1}{2} \sum_{n=1}^{N} \{y(x_n, W) - s_n\}^2$$

- Agent can find weights W to minimize the cost function

Simple example

- Issues: what should be the dimension of W? $\rightarrow$ the complexity of our model
- Will our approach handle noise?

- Heuristic: If the value of W is too high we will get overfitting

- solution: we can avoid overfitting through regularization: penalize high values in W:

$$\tilde{E}(w) = \frac{1}{2}\sum_{n=1}^{N}\{y(x_n, W) - s_n\}^2 + \frac{1}{2}\|W\|^2$$

- This still does not solve the problem of how to choose the number of parameters M in our model

- (Bayesian) nonparametric approach: adapt the number of parameters to the data

- Gaussian Processes (GPs): distribution over functions (Rasmussen 2010)
- Bayesian Nonparametric approach which models function as correlation between points
  - Underlying structure can be inferred from data
- $p(f|X) = N(f|0, K)$, where $K_{ij} = k(x_i, x_j)$ is the kernel function

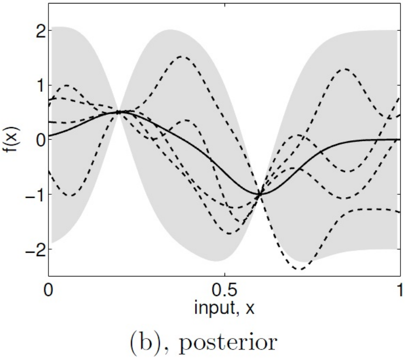


Figure: Posterior estimate given GP assumption

Outline

- $p(A)$ denotes the probability that the event A is true: Axioms of prob
  - $p(A) \geq 0$ (Probability is a positive number assigned to the event)
  - $P(C) = 1$ The probability of the certain event is 1
  - If A and B are mutually exclusive, then $p(A + B) = p(A) + p(B)$

Random variable

- (discrete) Random variable (RV): a variable that can take one of many values
- Denote the probability of event $X = x$ by $p(X = x)$ or simply $p(x)$
- **here $p(x)$ is the probability mass function**
  - $0 \geq p(x) \leq 1$ (The probability of some event happening between zero and one)
  - $\sum_{x \in X} p(x) = 1$ (Something happens)

Frequency definition

- $p(A) = \lim_{n \to \infty} \frac{n_A}{n}$, where n_A is the number of occurrences of A and n is the number of trials
- Classical definition $\Rightarrow$ For the random variable X $p(X = x_i) = c_i/N$ where N is the number of possible outcomes, c_i is the number of outcomes favorable to $X = x_i$
- e.g. even die roll: $\frac{3}{6}$
- However, the classical definition gives weired results, so the frequency definition is preferred

- For X be able to take any value x_i and RV Y be able to take any value y_i
- If c_i is the number of trials in which $X = x_i$ over a set of N trials, then frequentist definition of probability: $p(X = x_i) = c_i/N$ as $N \to \infty$
- Question: what is the joint probability that I will get "snake eyes"? i.e.$x_i = 1, y_i = 1$
- Joint probability: in the game of Craps, let X be the number of dots on a side of a dice, and Y on another

$$p(X = x_i, Y = y_i) = \frac{n_{ij}}{N}$$

- Here n_{ij} is the number of trials over which $X = x_i, Y = y_i$

- Probability of union of two events A, B, i.e. probability of **A or B**

$$\begin{aligned} p(A \vee B) &= p(A) + p(B) - P(A \wedge B) \\ &= p(A) + p(B) \;\; \text{If A and B are mutually exclusive} \end{aligned}$$

- Joint event $p(A, B)$ **Product Rule**

$$p(A, B) = p(A \wedge B) = p(A|B)p(B)$$

- Given $p(A, B)$ we define the marginal distribution over B
- Called as the Sum rule or rule of total probability

$$p(A) = \sum_b p(A|B) = \sum_b p(A|B = b)p(B = b)$$

If we are only concerned about the probability of one variable, we can *marginalize* or sum over the other variable, this leads to $p(X = x_i) = \sum_{j=1}^{L} p(X = x_i, Y = y_j)$. This leads to the sum rule

Sum rule

$$p(X) = \sum_{Y} p(X, Y) = \sum_{Y} p(X|Y = y)p(Y = y)$$

Conditional probability $p(Y = y_j|X = x_i)$

Product rule

$$p(X, Y) = p(Y|X)p(X)$$

Conditional Probability

Manipulating the Product Rule

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

Bayes Theorem

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

The denominator can be expressed as the *Total Probability* $p(Y) = \sum_{x'} p(Y = y|X = x')p(X = x')$. This is a normalization constant required to ensured that the lhs of Bayes rule over all values of Y is equal to 1 To get here, we never needed the frequency definition of probability

Example Bayes rule

Example 2.2.3.1 from Murphy

- Test sensitivity 80%, i.e. when you have cancer (y=1), test will be true (x=1): $p(x=1|y=1)=0.8$
- This is a case of a high likelihood of measuring x when the state is y
- But what is the *prior* probability of being in state y?: $\Rightarrow p(y)=0.004$
- Clearly, now $p(cancer=1|test=1)=p(y=1|x=1)\propto p(x|y)p(y)=0.8\times 0.004$
- But we must account for the total probability $p(x)$, which includes false positive $p(x=1|y=0)=0.1$

So Bayes law tells us:

$$\begin{aligned} p(y=1|x=1) &= \frac{p(x=1|y=1)p(y=1)}{p(x=1|y=1)p(y=1)+p(x=1|y=0)p(y=0)} \\ &= 0.031 \end{aligned}$$

Let y be the state, and x be a feature, then

$$p(y = c|x) = \frac{p(x|y = c)p(y = c)}{\sum_{c'} p(y = c'|\theta)p(x|y = c')}$$

- When we predict the LHS using the class conditional density (likelihood of features $p(x|y = c)$) and prior probability $p(y = c)$, we have a Generative classifier
- When we learn directly $p(y = c|x)$, i.e. the posterior, we get a discriminative classifier
- When the likelihood models are correct, Generative models will require far less data (remember Tennenbaum and friends)
- When the feature models are not correct, discriminative models can do better, at the cost of lot of data (LeCun and friends), they don't need the distribution of the features

More on generative vs discriminative modeling

- Discriminative models in general can do better on accuracy, because it might be hard to come up with class conditional probabilities
- But accuracy is not EVERYTHING, especially in autonomous decision making: how accurate do you need to be driving on the road?
- Generative modeling has a natural way of dealing with missing features (marginalize them)
- Generative models are known to do better with semi-supervised learning (Dirichlet allocations)
- But discriminative models can handle feature processing: preconditioning of data
- Recent success of deep learning is focused heavily on accuracy from unstructured data, but gets criticized for mistakes, label-sensitivity, and inability to handle missing data

Probability Density Functions (PDFs)

Concept of probability can be extended to continuous variables using the PDF

PDF

$$p(x \in (a, b)) = \int_a^b p(x)dx$$

- PDFs satisfy the rules of probability:

$$p(x) \geq 0$$

$$\int_{-\infty}^{\infty} p(x)dx = 1$$

- Sum and Product rules apply to pdfs:

$$p(x) = \int p(x, y)dy, \quad p(x, y) = p(y|x)p(x)$$

Expectation: Average value of a function

Expectation of a continuous pdf

$$\mathbb{E}[f] = \int p(x)f(x)dx$$

Notice LHS does not have x in it, why?

Expectation of function of several variables can be taken wrt a variable, e.g. for $f(x, y)$

$$\mathbb{E}_x[f] = \int \int p(x, y)f(x, y)dydx$$

$\mathbb{E}_x[f]$ is a function of y

Conditional expectation

$$\mathbb{E}_x[f(x|y)] = \int p(x|y)f(x)dx$$

Variance

Variance known as the second moment

Variance

$$Var[f] = \mathbb{E}[(f(x) - \mathbb{E}[f(x)])^2]$$

Useful identity

Variance

$$Var[f] = \mathbb{E}[f(x)^2] - \mathbb{E}[f(x)]^2$$

The Gaussian distribution:

$$\mathcal{N}(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\{-\frac{1}{2\pi\sigma^2}(x-\mu)^2\}$$

hyperparameters: Mean: μ, variance σ^2, std deviation σ Figuring out Gaussian distribution hyperparameters

- Bayesian curve fitting overview in book
- Challenge: what should be the number of parameters: model selection
- Curse of dimensionality: harder and harder to classify and predict as the dimensionality of the data increases
- Solution: Try to find a reduced dimension data set that reflects most of the information (e.g. SVD, Principle Component Analysis)
- Solution: Leverage smoothness and predictability in the data to interpolate across dimensions

- Binomial: Let X be the number of heads from n coin tosses, if the probability of heads is θ then $\theta \sim \mathrm{BIN}(n, \theta)$
- Mean$=n\theta$, and variance $= n\theta(1-\theta)$
- Bernoulli distribution, special case of Binomial with $n = 1$
- Binomial: used for events with binary outcomes (e.g. coin toss)
- Multinomial: Used for events with multiple discrete outcomes (e.g. dice throw)

- Poisson distribution: $X \sim \mathrm{Poi}(\lambda)$, with arrival rate λ if

$$\mathrm{Poi}(x|\lambda) = e^{-\lambda}\frac{\lambda^x}{x!}$$

- Poisson distribution used for modeling arrival rates of events
- Gaussian distribution: $\mathcal{N}(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{1}{2\sigma^2}(x-\mu)^2}$
- mean $= \mu$, the mean is the same as the mode for Gaussian, $\sigma^2 = \mathrm{Var}[x]$
- In the limit of $\sigma^2 \to 0$, Gaussian distribution becomes the Dirac delta function
- Gaussian distribution can be sensitive to outliers, two options:
- Student t distribution
- Laplace distribution: $\mathrm{Lap}(x|\mu,b) = \frac{1}{2b}e^{-\frac{\|x-\mu\|}{b}}$
- For Laplace distribution, mean $= \mu$, mode $= \mu$, variance $= 2\sigma^2$

- Beta distribution: A highly flexible distribution that can be morphed into other distributions, and has continuous support over $[0, 1]$
- Joint probability distributions: Covariance
- Multivariate Gaussian $\mathcal{N}(x|\mu, \sigma^2) = \frac{1}{(2\pi)^{D/2}\|\Sigma\|^{1/2}} e^{-\frac{1}{(x-\mu)^T\Sigma^{-1}(x-\mu)}}$
- Multivariate Student t
- Dirichlet

- Multivariate generalization of the Beta disrtibution
- Has continuous support, over the probability simplex $S_k = \{x : 0 \leq x_k \leq 1, \sum_{k=1}^{K} x_k = 1\}$
- PDF:

$$\mathrm{Dir}(x|\alpha) = \frac{1}{B(\alpha)} \prod_{k=1}^{K} x_k^{\alpha_k - 1} \mathbb{I}(x \in S_k)$$

  where $B(\alpha)$ is the k-dimensional generalization of the Beta function (see 2.76 Murphy)
- $\alpha_0 = \sum_{k=1}^{K} \alpha_k$ controls how peaked the distribution is and α_k control where the peaks are
- $\mathbb{E}[x_k] = \frac{\alpha_k}{\alpha_0}$, mode$[x_k]$=$\frac{\alpha_k - 1}{\alpha_0 - K}$, Var$[x_k]$=$\frac{\alpha_k(\alpha_0 - \alpha_k)}{\alpha_0^2(\alpha_0 + 1)}$

Monte Carlo approximations

- Consider the transformation of a random variable: $x_{k+1} = f(x_k)$
- What is the distribution of x_k?
- The linear case (Murphy 2.6.1), the mean the covariance after the transformation can be easily expressed analytically
- Not always easy for the nonlinear case $\Rightarrow$ Monte Carlo approximation
  - First generate s samples x_i from the base distribution
  - Propagate these samples through the transformation
  - Approximate the resulting distribution using the set $\{f(x_i)\}_{i=1}^{S}$
- Example: Approximate π

Outline

- There is a huge difference between Information and Data
- How much information is received when we observe a specific value of a random variable x?
- Amount of information $\rightarrow$ *degree-of-surprise* (Bayesian view in a way)
- e.g.: coin toss: information: 500 straight heads, does this contain more information than 300 heads and 200 tails?

- There is a huge difference between Information and Data
- How much information is received when we observe a specific value of a random variable x?
- Amount of information $\rightarrow$ *degree-of-surprise* (Bayesian view in a way)
- e.g.: coin toss: information: 500 straight heads, does this contain more information than 300 heads and 200 tails?
- If we receive information about an event that was certain to happen, we have received no information

- There is a huge difference between Information and Data
- How much information is received when we observe a specific value of a random variable x?
- Amount of information $\rightarrow$ *degree-of-surprise* (Bayesian view in a way)
- e.g.: coin toss: information: 500 straight heads, does this contain more information than 300 heads and 200 tails?
- If we receive information about an event that was certain to happen, we have received no information
- Measure in the information content depends on the probability distribution of x
- Information content is captured in a monotonic function h of $p(x)$, the probability distribution of x

Why is Information Theory Useful for Decision-Making?

Problems Solved in Uncertainty Quantification by Information Theory

Problems Solved in Uncertainty Quantification by Information Theory

- **Agnostic to skew and multi-modal distributions**

Why is Information Theory Useful for Decision-Making?

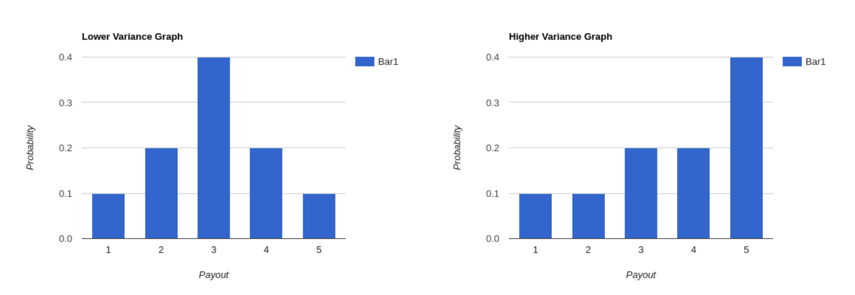
Problems Solved in Uncertainty Quantification by Information Theory

- **Agnostic to skew and multi-modal distributions**
- Scale-invariant quantification of statistical dispersion

Problems Solved in Uncertainty Quantification by Information Theory

- **Agnostic to skew and multi-modal distributions**
- Scale-invariant quantification of statistical dispersion

Introductory Example

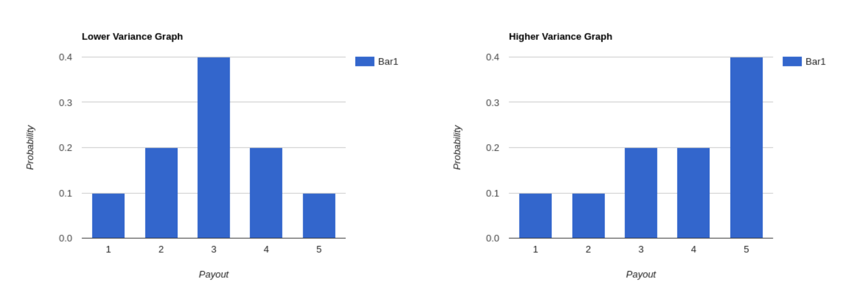


Why is Information Theory Useful for Decision-Making?

Problems Solved in Uncertainty Quantification by Information Theory

- **Agnostic to skew and multi-modal distributions**
- Scale-invariant quantification of statistical dispersion

Introductory Example

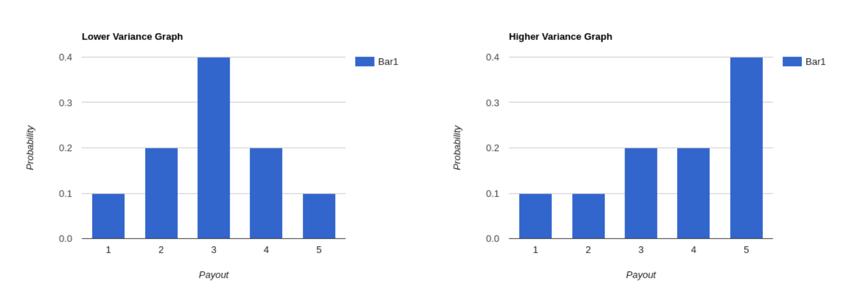


- Variance of left graph is 2.1216 $units^2$ & variance of right graph is 2.81 $units^2$

Problems Solved in Uncertainty Quantification by Information Theory

- **Agnostic to skew and multi-modal distributions**
- Scale-invariant quantification of statistical dispersion

Introductory Example



- Variance of left graph is 2.1216 *units*^{2} & variance of right graph is 2.81 *units*^{2}
- **Entropy is equal!** $\left(\frac{1}{5}\log\left(\frac{3125}{2}\right)\right)$

A Tale of Two Cryptographers

Connecting to Previous Example

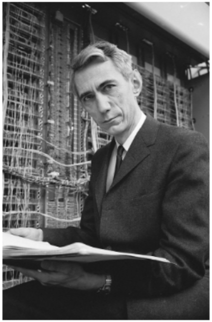
Difficulty to decode a message is agnostic to value of message.

Claude Shannon (1916-2001)

- WWII: Cryptography at Bell Labs
- *Mathematical Theory of Communication* [3]
- *Communication Theory of Secrecy Systems* [?]
- "Father of the Information Age"

Alan Turing (1912-1954)

- WWII: Cryptography at Bletchley Park
- "Father of the Artificial Intelligence"
- In 1940, used similar math to Information Theory to decipher Enigma Machine [2]



A hundred years after his birth, Claude Shannon's fingerprints are on every electronic device we own.

Photograph by Alfred Eisenstaedt / The LIFE Picture Collection / Getty

A Tale of Two Thumb Drives

A Tale of Two Thumb Drives

- Each drive is 1 Gbit, how much data can I store in **two** 1 Gbit drives?

A Tale of Two Thumb Drives

- Each drive is 1 Gbit, how much data can I store in **two** 1 Gbit drives?
- What values are storable in digital drives?

A Tale of Two Thumb Drives

- Each drive is 1 Gbit, how much data can I store in **two** 1 Gbit drives?
- What values are storable in digital drives?
- Data: Stored values

A Tale of Two Thumb Drives

- Each drive is 1 Gbit, how much data can I store in **two** 1 Gbit drives?
- What values are storable in digital drives?
- Data: Stored values
- Information: Amount of data that can be stored

A Tale of Two Thumb Drives

- Each drive is 1 Gbit, how much data can I store in **two** 1 Gbit drives?
- What values are storable in digital drives?
- Data: Stored values
- Information: Amount of data that can be stored

Dealing with Random Variables

- Data: Values generated by a random variable
- Information: Amount of values that can be generated by a random variable

More Thumb Drive Questions

More Thumb Drive Questions

- How many combinations can be stored in **one** 1 Gbit thumb drive?

More Thumb Drive Questions

- How many combinations can be stored in **one** 1 Gbit thumb drive?
  - $2^{(1 \text{ Billion})}$

More Thumb Drive Questions

- How many combinations can be stored in **one** 1 Gbit thumb drive?
  - $2^{(1 \text{ Billion})}$
- How many combinations can be stored in **two** 1 Gbit thumb drives?

More Thumb Drive Questions

- How many combinations can be stored in **one** 1 Gbit thumb drive?
  - $2^{(1 \text{ Billion})}$
- How many combinations can be stored in **two** 1 Gbit thumb drives?
  - $2^{(2 \text{ Billion})}$

More Thumb Drive Questions

- How many combinations can be stored in **one** 1 Gbit thumb drive?
  - $2^{(1 \text{ Billion})}$
- How many combinations can be stored in **two** 1 Gbit thumb drives?
  - $2^{(2 \text{ Billion})}$
- Recall that the combined storage of two 1Gbit thumb drives is 2Gbit.

More Thumb Drive Questions

- How many combinations can be stored in **one** 1 Gbit thumb drive?
  - $2^{(1 \text{ Billion})}$
- How many combinations can be stored in **two** 1 Gbit thumb drives?
  - $2^{(2 \text{ Billion})}$
- Recall that the combined storage of two 1Gbit thumb drives is 2Gbit.
  - 1Gbit + 1Gbit = 2Gbit

More Thumb Drive Questions

- How many combinations can be stored in **one** 1 Gbit thumb drive?
  - $2^{(1 \text{ Billion})}$
- How many combinations can be stored in **two** 1 Gbit thumb drives?
  - $2^{(2 \text{ Billion})}$
- Recall that the combined storage of two 1Gbit thumb drives is 2Gbit.
  - 1Gbit + 1Gbit = 2Gbit
- What function converts $2^{(1 \text{ Billion})}$ to 1Gbit?

More Thumb Drive Questions

- How many combinations can be stored in **one** 1 Gbit thumb drive?
  - $2^{(1 \text{ Billion})}$
- How many combinations can be stored in **two** 1 Gbit thumb drives?
  - $2^{(2 \text{ Billion})}$
- Recall that the combined storage of two 1Gbit thumb drives is 2Gbit.
  - 1Gbit + 1Gbit = 2Gbit
- What function converts $2^{(1 \text{ Billion})}$ to 1Gbit?
  - $\log_2(2^{(1 \text{ Billion})})$

More Thumb Drive Questions

- How many combinations can be stored in **one** 1 Gbit thumb drive?
  - $2^{(1 \text{ Billion})}$
- How many combinations can be stored in **two** 1 Gbit thumb drives?
  - $2^{(2 \text{ Billion})}$
- Recall that the combined storage of two 1Gbit thumb drives is 2Gbit.
  - 1Gbit + 1Gbit = 2Gbit
- What function converts $2^{(1 \text{ Billion})}$ to 1Gbit?
  - $\log_2(2^{(1 \text{ Billion})})$
  - $\log_2(2^{(1 \text{ Billion})}) + \log_2(2^{(1 \text{ Billion})}) = \log_2(2^{(2 \text{ Billion})})$

Information Axioms for Random Variables

Information Axioms for Random Variables

- Monotonically increasing in N

Information Axioms for Random Variables

- Monotonically increasing in N
  - $\sum_{i=1}^{N} \log_b (1/P(x_i))$ (from thumb drive example)

Information Axioms for Random Variables

- Monotonically increasing in N
  - $\sum_{i=1}^{N} \log_b (1/P(x_i))$ (from thumb drive example)
- Continuity in probability

Information Axioms for Random Variables

- Monotonically increasing in N
  - $\sum_{i=1}^{N} \log_b (1/P(x_i))$ (from thumb drive example)
- Continuity in probability
  - $\sum_{i=1}^{N} P(X = x_i) \log_b (1/P(X = x_i))$

Information Axioms for Random Variables

- Monotonically increasing in N
  - $\sum_{i=1}^{N} \log_b (1/P(x_i))$ (from thumb drive example)
- Continuity in probability
  - $\sum_{i=1}^{N} P(X = x_i) \log_b (1/P(X = x_i))$
  - $0 \log_b (1/0) = 0$

Information Axioms for Random Variables

- Monotonically increasing in N
  - $\sum_{i=1}^{N} \log_b(1/P(x_i))$ (from thumb drive example)
- Continuity in probability
  - $\sum_{i=1}^{N} P(X = x_i) \log_b(1/P(X = x_i))$
  - $0 \log_b(1/0) = 0$

Definition of Information Entropy

$$H(X) := \sum_{i=1}^{N} P(X = x_i) \log_b \left(\frac{1}{P(X = x_i)} \right)$$

Equivalent to Equation 2.1 in [1]

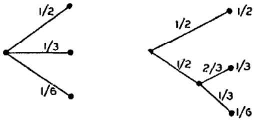


Figure: RHS involves conditional probability

Composition Theorem

Final Axiom of Information Entropy

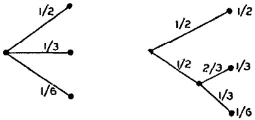


Figure: RHS involves conditional probability

Composition Theorem

- $H\left(\frac{1}{2}, \frac{1}{3}, \frac{1}{6}\right) = H\left(\frac{1}{2}, \frac{1}{2}\right) + \frac{1}{2}H\left(\frac{2}{3}, \frac{1}{3}\right)$

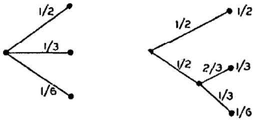


Figure: RHS involves conditional probability

Composition Theorem

- $H\left(\frac{1}{2}, \frac{1}{3}, \frac{1}{6}\right) = H\left(\frac{1}{2}, \frac{1}{2}\right) + \frac{1}{2} H\left(\frac{2}{3}, \frac{1}{3}\right)$
  - LHS $H\left(\frac{1}{2}, \frac{1}{3}, \frac{1}{6}\right) = \frac{1}{2}\log_b(2) + \frac{1}{3}\log_b(3) + \frac{1}{6}\log_b(6)$
    $= \frac{1}{6}\log_b(8) + \frac{1}{6}\log_b(9) + \frac{1}{6}\log_b(6) = \frac{1}{6}\log_b(432)$

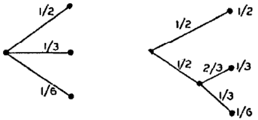
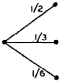


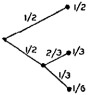
Figure: RHS involves conditional probability

Composition Theorem

- $H\left(\frac{1}{2},\frac{1}{3},\frac{1}{6}\right) = H\left(\frac{1}{2},\frac{1}{2}\right) + \frac{1}{2}H\left(\frac{2}{3},\frac{1}{3}\right)$
  - LHS $H\left(\frac{1}{2},\frac{1}{3},\frac{1}{6}\right) = \frac{1}{2}\log_b(2) + \frac{1}{3}\log_b(3) + \frac{1}{6}\log_b(6)$
    $= \frac{1}{6}\log_b(8) + \frac{1}{6}\log_b(9) + \frac{1}{6}\log_b(6) = \frac{1}{6}\log_b(432)$
  - RHS $H\left(\frac{1}{2},\frac{1}{2}\right) + \frac{1}{2}H\left(\frac{2}{3},\frac{1}{3}\right) = \frac{1}{2}\log_b(2) + \frac{1}{2}\log_b(2) + \frac{1}{2}\frac{2}{3}\log_b\left(\frac{3}{2}\right) + \frac{1}{2}\frac{1}{3}\log_b(3)$
    $= \frac{1}{6}\log_b(8) + \frac{1}{6}\log_b(8) + \frac{1}{6}\log_b\left(\frac{9}{4}\right) + \frac{1}{6}\log_b(3)$
    $= \frac{1}{6}\log_b(64) + \frac{1}{6}\log\left(\frac{27}{4}\right) = \frac{1}{6}\log_b(432)$

$$\underbrace{H\left(\frac{1}{2},\frac{1}{3},\frac{1}{6}\right)}_{\text{Joint Entropy}} = \underbrace{H\left(\frac{1}{2},\frac{1}{2}\right)}_{\text{Entropy}} + \underbrace{\frac{1}{2}\qquad\underbrace{H\left(\frac{2}{3},\frac{1}{2}\right)}_{\text{Specific Conditional Entropy}}}_{\text{Conditional Entropy}}$$

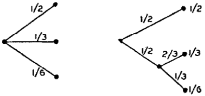




Four Entropies

Four Entropies of the Composition Theorem

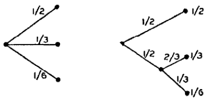
$$\underbrace{H\left(\frac{1}{2},\frac{1}{3},\frac{1}{6}\right)}_{\text{Joint Entropy}} = \underbrace{H\left(\frac{1}{2},\frac{1}{2}\right)}_{\text{Entropy}} + \underbrace{\frac{1}{2}\underbrace{H\left(\frac{2}{3},\frac{1}{2}\right)}_{\text{Specific Conditional Entropy}}}_{\text{Conditional Entropy}}$$



Four Entropies

- Information Entropy

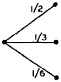
$$\underbrace{H\left(\frac{1}{2},\frac{1}{3},\frac{1}{6}\right)}_{\text{Joint Entropy}} = \underbrace{H\left(\frac{1}{2},\frac{1}{2}\right)}_{\text{Entropy}} + \underbrace{\frac{1}{2}\qquad\underbrace{H\left(\frac{2}{3},\frac{1}{2}\right)}_{\text{Specific Conditional Entropy}}}_{\text{Conditional Entropy}}$$

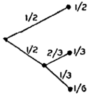


Four Entropies

- Information Entropy
  - $H(X) := -\sum_{i=1}^{N} P(X=x_i)\log_b\left(\frac{1}{P(X=x_i)}\right)$

$$\underbrace{H\left(\frac{1}{2},\frac{1}{3},\frac{1}{6}\right)}_{\text{Joint Entropy}} = \underbrace{H\left(\frac{1}{2},\frac{1}{2}\right)}_{\text{Entropy}} + \underbrace{\frac{1}{2}\underbrace{H\left(\frac{2}{3},\frac{1}{2}\right)}_{\text{Specific Conditional Entropy}}}_{\text{Conditional Entropy}}$$



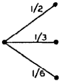


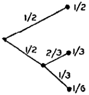
Four Entropies

- Information Entropy
  - $H(X) := -\sum_{i=1}^{N} P(X = x_i) \log_b \left(\frac{1}{P(X=x_i)}\right)$
- Specific Conditional Entropy

Four Entropies of the Composition Theorem

$$\underbrace{H\left(\frac{1}{2},\frac{1}{3},\frac{1}{6}\right)}_{\text{Joint Entropy}} = \underbrace{H\left(\frac{1}{2},\frac{1}{2}\right)}_{\text{Entropy}} + \underbrace{\frac{1}{2}\underbrace{H\left(\frac{2}{3},\frac{1}{2}\right)}_{\text{Specific Conditional Entropy}}}_{\text{Conditional Entropy}}$$

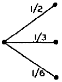


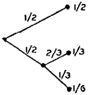


Four Entropies

- **Information Entropy**
  - $H(X) := -\sum_{i=1}^{N} P(X = x_i)\log_b\left(\frac{1}{P(X=x_i)}\right)$
- **Specific Conditional Entropy**
  - $H(X|Y = y) := \sum_{i=1}^{N} P(X = x_i|Y = y)\log_b\left(\frac{1}{P(X=x_i|Y=y)}\right)$

$$\underbrace{H\left(\frac{1}{2},\frac{1}{3},\frac{1}{6}\right)}_{\text{Joint Entropy}} = \underbrace{H\left(\frac{1}{2},\frac{1}{2}\right)}_{\text{Entropy}} + \underbrace{\frac{1}{2}\underbrace{H\left(\frac{2}{3},\frac{1}{2}\right)}_{\text{Specific Conditional Entropy}}}_{\text{Conditional Entropy}}$$

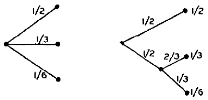




Four Entropies

- **Information Entropy**
  - $H(X) := -\sum_{i=1}^{N} P(X = x_i) \log_b \left(\frac{1}{P(X=x_i)}\right)$
- **Specific Conditional Entropy**
  - $H(X|Y = y) := \sum_{i=1}^{N} P(X = x_i|Y = y) \log_b \left(\frac{1}{P(X=x_i|Y=y)}\right)$
- **Conditional Entropy**

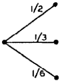
$$\underbrace{H\left(\frac{1}{2},\frac{1}{3},\frac{1}{6}\right)}_{\text{Joint Entropy}} = \underbrace{H\left(\frac{1}{2},\frac{1}{2}\right)}_{\text{Entropy}} + \underbrace{\frac{1}{2}\underbrace{H\left(\frac{2}{3},\frac{1}{2}\right)}_{\text{Specific Conditional Entropy}}}_{\text{Conditional Entropy}}$$

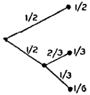


Four Entropies

- **Information Entropy**
  - $H(X) := -\sum_{i=1}^{N} P(X = x_i) \log_b \left(\frac{1}{P(X=x_i)}\right)$
- **Specific Conditional Entropy**
  - $H(X|Y = y) := \sum_{i=1}^{N} P(X = x_i|Y = y) \log_b \left(\frac{1}{P(X=x_i|Y=y)}\right)$
- **Conditional Entropy**
  - $H(X|Y) := \sum_{i=1}^{N} \sum_{j=1}^{M} P(X = x_i, Y = y_j) \log_b \left(\frac{1}{P(X=x_i|Y=y_j)}\right)$

$$\underbrace{H\left(\frac{1}{2},\frac{1}{3},\frac{1}{6}\right)}_{\text{Joint Entropy}} = \underbrace{H\left(\frac{1}{2},\frac{1}{2}\right)}_{\text{Entropy}} + \underbrace{\frac{1}{2} \qquad \underbrace{H\left(\frac{2}{3},\frac{1}{2}\right)}_{\text{Specific Conditional Entropy}}}_{\text{Conditional Entropy}}$$



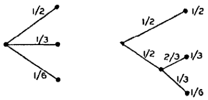


Four Entropies

- Information Entropy
  - $H(X) := -\sum_{i=1}^{N} P(X = x_i) \log_b \left(\frac{1}{P(X=x_i)}\right)$
- Specific Conditional Entropy
  - $H(X|Y = y) := \sum_{i=1}^{N} P(X = x_i|Y = y) \log_b \left(\frac{1}{P(X=x_i|Y=y)}\right)$
- Conditional Entropy
  - $H(X|Y) := \sum_{i=1}^{N} \sum_{j=1}^{M} P(X = x_i, Y = y_j) \log_b \left(\frac{1}{P(X=x_i|Y=y_j)}\right)$
- Joint Entropy

Four Entropies of the Composition Theorem

$$\underbrace{H\left(\frac{1}{2},\frac{1}{3},\frac{1}{6}\right)}_{\text{Joint Entropy}} = \underbrace{H\left(\frac{1}{2},\frac{1}{2}\right)}_{\text{Entropy}} + \underbrace{\frac{1}{2}\underbrace{H\left(\frac{2}{3},\frac{1}{2}\right)}_{\text{Specific Conditional Entropy}}}_{\text{Conditional Entropy}}$$

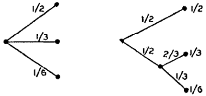


Four Entropies

- Information Entropy
  - $H(X) := -\sum_{i=1}^{N} P(X = x_i) \log_b \left(\frac{1}{P(X=x_i)}\right)$
- Specific Conditional Entropy
  - $H(X|Y = y) := \sum_{i=1}^{N} P(X = x_i|Y = y) \log_b \left(\frac{1}{P(X=x_i|Y=y)}\right)$
- Conditional Entropy
  - $H(X|Y) := \sum_{i=1}^{N} \sum_{j=1}^{M} P(X = x_i, Y = y_j) \log_b \left(\frac{1}{P(X=x_i|Y=y_j)}\right)$
- Joint Entropy
  - $H(X, Y) := \sum_{i=1}^{N} \sum_{j=1}^{M} P(X = x_i, Y = y_j) \log_b \left(\frac{1}{P(X=x_i,Y=y_j)}\right)$

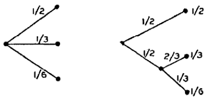
Relationships Between Four Entropies

$$\underbrace{H\left(\frac{1}{2},\frac{1}{3},\frac{1}{6}\right)}_{\text{Joint Entropy}} = \underbrace{H\left(\frac{1}{2},\frac{1}{2}\right)}_{\text{Entropy}} + \underbrace{\frac{1}{2}\qquad\underbrace{H\left(\frac{2}{3},\frac{1}{2}\right)}_{\text{Specific Conditional Entropy}}}_{\text{Conditional Entropy}}$$



Use Venn Diagrams on the Chalk Board to Visualize!

$$\underbrace{H\left(\frac{1}{2},\frac{1}{3},\frac{1}{6}\right)}_{\text{Joint Entropy}} = \underbrace{H\left(\frac{1}{2},\frac{1}{2}\right)}_{\text{Entropy}} + \underbrace{\frac{1}{2}\quad\underbrace{H\left(\frac{2}{3},\frac{1}{2}\right)}_{\text{Specific Conditional Entropy}}}_{\text{Conditional Entropy}}$$

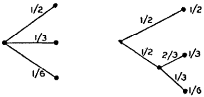


Use Venn Diagrams on the Chalk Board to Visualize!

- $H(X,Y) = H(X) + H(Y|X) = H(Y) + H(X|Y)$

Relationships Between Four Entropies

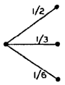
$$\underbrace{H\left(\frac{1}{2},\frac{1}{3},\frac{1}{6}\right)}_{\text{Joint Entropy}} = \underbrace{H\left(\frac{1}{2},\frac{1}{2}\right)}_{\text{Entropy}} + \underbrace{\frac{1}{2}\qquad\underbrace{H\left(\frac{2}{3},\frac{1}{2}\right)}_{\text{Specific Conditional Entropy}}}_{\text{Conditional Entropy}}$$

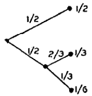


Use Venn Diagrams on the Chalk Board to Visualize!

- $H(X,Y) = H(X) + H(Y|X) = H(Y) + H(X|Y)$
- $H(X) + H(Y) \geq H(X,Y)$

$$\underbrace{H\left(\frac{1}{2},\frac{1}{3},\frac{1}{6}\right)}_{\text{Joint Entropy}} = \underbrace{H\left(\frac{1}{2},\frac{1}{2}\right)}_{\text{Entropy}} + \underbrace{\frac{1}{2}\quad\underbrace{H\left(\frac{2}{3},\frac{1}{2}\right)}_{\text{Specific Conditional Entropy}}}_{\text{Conditional Entropy}}$$

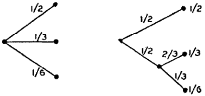




Use Venn Diagrams on the Chalk Board to Visualize!

- $H(X,Y) = H(X) + H(Y|X) = H(Y) + H(X|Y)$
- $H(X) + H(Y) \geq H(X,Y)$
- $H(X|Y) \leq H(X)$

$$\underbrace{H\left(\frac{1}{2},\frac{1}{3},\frac{1}{6}\right)}_{\text{Joint Entropy}} = \underbrace{H\left(\frac{1}{2},\frac{1}{2}\right)}_{\text{Entropy}} + \underbrace{\frac{1}{2}\underbrace{H\left(\frac{2}{3},\frac{1}{2}\right)}_{\text{Specific Conditional Entropy}}}_{\text{Conditional Entropy}}$$



Use Venn Diagrams on the Chalk Board to Visualize!

- $H(X,Y) = H(X) + H(Y|X) = H(Y) + H(X|Y)$
- $H(X) + H(Y) \geq H(X,Y)$
- $H(X|Y) \leq H(X)$
- $H(Y|X) \leq H(Y)$

Mind your P's and Q's!

$$H_\times\left(Q(X), P(X)\right) := \sum_{i=1}^{N} Q(X = x_i) \log_b \left(\frac{1}{P(X = x_i)} \right)$$

.

Mind your P's and Q's!

$$H_\times\left(Q(X), P(X)\right) := \sum_{i=1}^{N} Q(X = x_i) \log_b \left(\frac{1}{P(X = x_i)} \right)$$

- $H_\times\left(Q(X), P(X)\right) \geq H_Q(X)$ $\leftarrow$ Important!

.

Mind your P's and Q's!

$$H_\times \left(Q(X), P(X)\right) := \sum_{i=1}^{N} Q(X = x_i) \log_b \left(\frac{1}{P(X = x_i)}\right)$$

- $H_\times \left(Q(X), P(X)\right) \geq H_Q(X)$ $\leftarrow$ Important!
- In general, $H_\times \left(Q(X), P(X)\right) \neq H_\times \left(P(X), Q(X)\right)$

.

Mind your P's and Q's!

$$H_{\times}\left(Q(X), P(X)\right) := \sum_{i=1}^{N} Q(X = x_i) \log_b \left(\frac{1}{P(X = x_i)}\right)$$

- $H_{\times}\left(Q(X), P(X)\right) \geq H_Q(X)$ $\leftarrow$ Important!
- In general, $H_{\times}\left(Q(X), P(X)\right) \neq H_{\times}\left(P(X), Q(X)\right)$
- Recall conditional entropy

.

Mind your P's and Q's!

$$H_\times\left(Q(X), P(X)\right) := \sum_{i=1}^{N} Q(X = x_i) \log_b \left(\frac{1}{P(X = x_i)} \right)$$

- $H_\times\left(Q(X), P(X)\right) \geq H_Q(X)$ $\leftarrow$ Important!
- In general, $H_\times\left(Q(X), P(X)\right) \neq H_\times\left(P(X), Q(X)\right)$
- Recall conditional entropy
  - $H(X|Y) := \sum_{i=1}^{N} \sum_{j=1}^{M} P(X = x_i, Y = y_j) \log_b \left(\frac{1}{P(X=x_i|Y=y_j)} \right)$

.

Mind your P's and Q's!

$$H_\times\left(Q(X), P(X)\right) := \sum_{i=1}^{N} Q(X = x_i) \log_b \left(\frac{1}{P(X = x_i)}\right)$$

- $H_\times\left(Q(X), P(X)\right) \geq H_Q(X)$ $\leftarrow$ Important!
- In general, $H_\times\left(Q(X), P(X)\right) \neq H_\times\left(P(X), Q(X)\right)$
- Recall conditional entropy
  - $H(X|Y) := \sum_{i=1}^{N} \sum_{j=1}^{M} P(X = x_i, Y = y_j) \log_b \left(\frac{1}{P(X=x_i|Y=y_j)}\right)$
  - NOT a cross entropy! Why?

.

- Differential Entropy

$$H[x] = -\int p(x) \ln p(x) dx$$

- The Gaussian distribution maximizes differential Entropy

- The differential Entropy of the Guassian:

$$H[x] = \frac{1}{2}\{1 + \ln(2\pi\sigma^2)\}$$

- Differential Entropy

$$H[x] = -\int p(x) \ln p(x) dx$$

- The Gaussian distribution maximizes differential Entropy

- The differential Entropy of the Guassian:

$$H[x] = \frac{1}{2}\{1 + \ln(2\pi\sigma^2)\}$$

- Scaling issues can be abated using *Relative Entropy*

- A way of comparing two distributions $p(x)$ and $q(x)$

Kullback Leibler Divergence

$$\mathrm{KL} = -\int p(x)\ln\{\frac{q(x)}{p(x)}\}dx$$

- note that $\mathrm{KL}(p||q) \neq \mathrm{KL}(q||p)$
- $\mathrm{KL}(p||q) \geq 0$ (Jensen's inequality in Eq 2.100 in [1])
- $\mathrm{KL}(p||q) = 0$ if and only if $p(x) = q(x)$
- **Mutual information**: Amount of information that can be obtained about one random variable by observing another
- $I[x, y] = \mathrm{KL}(p(x, y)||p(x)p(y))$
- $I[x, y] = H[x] - H[x|y] = H[y] - H[y|x]$

Table: Information Metrics

Metric	Formula
Kullback-Leibler	$D_{\mathrm{KL}}(p\|\|q) = \int p \log(\frac{p}{q}) dx$
Renyi	$D_\alpha(p\|\|q) = \frac{1}{\alpha-1} \log \int p^\alpha q^{1-\alpha} dx, \ \alpha > 1$
Chernoff	$D_c(p\|\|q) = \log \int p^\alpha q^{1-\alpha} dx$
f-divergence	$D_f(p\|\|q) = \int f(\frac{p}{q}) dq(x)$
Varational	$V(p\|\|q) = \int \|p - q\| dx$
Matusita	$D_M(p\|\|q) = \left[\int \|p^{\frac{1}{r}} - q^{\frac{1}{r}}\|^r dx\right]^{\frac{1}{r}}, \ r > 0$

$p(x)$ and $q(x)$ are two probability distributions

Outline

We want to predict $p(x|y)$, i.e. the unknown variable x given some information y relating to that variable

$$p(x|y) = \frac{p(x)p(y|x)}{\sum_x p(x)p(y|x)}$$

Before new measurement is available, we have the prior predictive distribution, also known as marginal distribution, of y

$$p(y) = \int p(y,x)dx = \int p(x)p(y|x)dx$$

After new measurements of y, we get the posterior predictive distribution of $\tilde{y}$ which is dependent on x

$$\begin{aligned} p(\tilde{y}|y) &= \int p(\tilde{y},x|y)dx \\ &= \int p(\tilde{y}|x,y)p(x|y)dx \\ &= \int p(\tilde{y}|x)p(x|y)dx \end{aligned}$$

The second equation is posterior distribution of $\tilde{y}$ conditioned on x given y and the last line is true if $\tilde{y}$ and y are independent given x

What is Kalman filter

- **The filtering problem**: Find the best estimate of the true value of a system's state given noisy measurements of some values of that system's states
- What should a good filter do?
  - Provide an accurate and un-biased estimate
  - Provide confidence in its estimate

What is Kalman filter

- **The filtering problem**: Find the best estimate of the true value of a system's state given noisy measurements of some values of that system's states
- What should a good filter do?
  - Provide an accurate and un-biased estimate
  - Provide confidence in its estimate

The Kalman Filter

The Kalman filter is an optimal estimator for estimating the states of a linear dynamical system from sensor measurements corrupted with Gaussian white noise of some of that system's states.

- **The filtering problem**: Find the best estimate of the true value of a system's state given noisy measurements of some values of that system's states
- What should a good filter do?
  - Provide an accurate and un-biased estimate
  - Provide confidence in its estimate

The Kalman Filter

The Kalman filter is an optimal estimator for estimating the states of a linear dynamical system from sensor measurements corrupted with Gaussian white noise of some of that system's states.

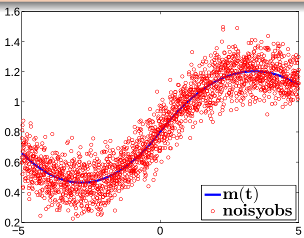


Figure: Noisy data and mean

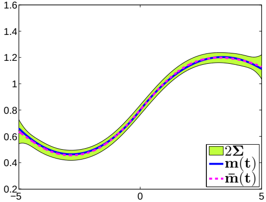
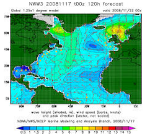
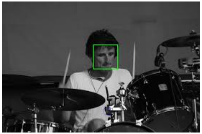


Figure: Estimate of the mean with predictive covariance

- The Kalman filter finds many many applications across pretty much all important scientific disciplines
- Its early application was on trajectory estimation on the Apollo space-craft
- Since then, it has been applied to many dynamical system state estimation problems, including: Cellphone, GPS, weather monitoring, precision agriculture, digital camera, ag-sensors.






Kalman filtering problem setup

- The Kalman filter will return a mean (average) of the quantity being estimated and provide a predictive variance on its estimate given:
  - The process and measurement noise variance is known
  - The dynamical system model is known
- The Kalman filter is guaranteed to be the optimal un-biased estimator for the following case:
  - The noise in the sensors in Gaussian
  - The dynamical system is linear
  - Sufficient number of states of the dynamical system are measured (the system is *observable*)
- If these assumptions are violated, the Kalman filter will be sub-optimal

- If the system dynamics is Linear and Time-Invariant (LTI), then the state space model is of the form

Noisy continuous-time LTI systems

$$\dot{x} = Ax + B\omega_t \quad (1)$$

$$y = Hx + v_t \quad (2)$$

$x \in \mathbb{R}^{n\times 1}$ state vector
$\omega \in \mathbb{R}^{n\times 1}$ (additive) process noise
$y \in \mathbb{R}^{l\times 1}$ sensor measurements
$v \in \mathbb{R}^{l\times 1}$ (additive) measurement noise

$A \in \mathbb{R}^{n\times n}$ state matrix
$B \in \mathbb{R}^{n\times m}$ the input matrix (here we are using it for inputting noise)
$H \in \mathbb{R}^{l\times n}$ output matrix

The assumptions on the process and measurement noise are:

- Zero mean, uncorrelated, i.e. $\omega_k \sim \mathbb{N}(0, \sigma_\omega^2)$, $v_k \sim \mathbb{N}(0, \sigma_v^2)$
- $E[(\omega_t, \omega_s)] = Q\delta(t-s)$, $E[(v_t, v_s)] = R\delta(t-s)$, where δ is the dirac delta function, s.t. $\delta(t-s) = 1$ when $t = s$.
- no cross correlation between ω_k and v_k, i.e. $cov(\omega_k, v_k) = 0$ for all k

Herein lies the "official" definition of Process and Measurement noise. What it is saying is that you can set the diagonal term as σ_ω^2 and σ_v^2

- Practically, Q matrix represents the confidence in the process model, larger the Q matrix, the less confident we are
- Practically, R matrix represents the confidence in the measurements from the correcting sensors, higher the R matrix, the less confident in the measurements we are

- If the system dynamics is Linear and Time-Invariant (LTI), then the state space model is of the form

Noisy discrete-time systems

$$x_{k+1} = \Phi_k x_k + \Gamma_k \omega_k \tag{3}$$

$$y_k = H_k x_k + v_k \tag{4}$$

$x \in \mathbb{R}^{n\times 1}$ state vector
$\omega \in \mathbb{R}^{n\times 1}$ (additive) process noise
$y \in \mathbb{R}^{l\times 1}$ sensor measurements
$v \in \mathbb{R}^{l\times 1}$ (additive) measurement noise

$\Phi_k \in \mathbb{R}^{n\times n}$ discretized state transition matrix
$\Gamma_k \in \mathbb{R}^{n\times m}$ Discretized input matrix
$H_k \in \mathbb{R}^{l\times n}$ output matrix

- The matrices Φ, H are called the system matrices and they depend on the physical parameters of the system such as mass, growth rate...
- note that these are the discrete versions of the equations: $\dot{x} = A(t)x + B(t)u; y = C(t)x$, in MATLAB the command is $c2d$
- In particular, $\Phi_k = e^{A\Delta t}$, $\Gamma_k = B(t)\Delta t$, $H_k = C(t)$, where Δt is the sampling time (dt)
- The process noise ω encodes our uncertainty in the knowledge of the dynamical evolution
- The measurement noise v encodes sensor measurement uncertainty

Q_d is the discrete time version of Q, remember, $E[(\omega_t, \omega_s)] = Q_k \delta(t - s)$
TO get Q_d we integrate the dynamics in the continuous time case:

$$x(k+1) = \Phi_k x(k) + \int_{t_k}^{t_k + \Delta t} e^{A(t_{k+1} - \lambda)} B(\lambda) \omega(\lambda) d\lambda \tag{5}$$

So we have

$$\omega_k = \int_{t_k}^{t_k + \Delta t} e^{A(t_{k+1} - \lambda)} B(\lambda) \omega(\lambda) d\lambda \tag{6}$$

The exact solution is (assuming white noise process, and computing $Q_{d_k} = cov(\omega_k)$)

$$Q_{d_k} = \int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, s) B(s) Q(s) B^T(s) \Phi(t_{k+1}, s)^T ds \tag{7}$$

A good approximation is: $Q_d = BQB^T \Delta T$ this is only good as long as the eigenvalue norm satisfies $\|A\Delta t\|_F << 1$

- Let $x = [x(1), x(2), ..., x(n)] \in \mathbb{R}^n$, with $x(i)$ the i^{th} component of x, then the covariance matrix $P \in \mathbb{R}^{n \times n}$ is defined as:

$$\begin{aligned}\mathrm{COV}(x) &\triangleq \mathbf{E}[(x-\bar{x})(x-\bar{x})^T] \\ &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} (x-\bar{x})(x-\bar{x})^T dx(1)dx(2)\cdots dx(n) \qquad \triangleq P\end{aligned}$$

- The i^{th} diagonal elements of P are the variance of $x(i)$ given by σ_i^2
- The off-diagonals are the cross-correlations, given by $\sigma_i \sigma_j$
- The covariance matrix is symmetric and positive definite, it can always be diagonalized

Additive process and measurement noise

The zero-mean additive Gaussian white noise assumption

$\omega_k \sim \mathbb{N}(0, Q_k)$: measurement noise, encodes the uncertainty in the sensors

$v_k \sim \mathbb{N}(0, R_k)$: The process noise, encodes our uncertainty in modeling the process

- Here $Q_k \in \mathbb{R}^{n\times n}$ and $R_k \in \mathbb{R}^{l\times l}$ are positive definite matrices, encoding the process and measurement noise covariances
- Typically sufficient to pick diagonal matrices with positive entries
- The measurement noise R_k (typically stationary: R) is typically provided in the sensor specification sheets, its the variance of the sensor
- Q_k (or when stationary: Q) is a little more difficult to find, typically this is the variable that needs to be *tuned*

$$
\begin{aligned}
E[x_{k+1}] &= E[\Phi_k x_k + \omega_k] && (8)\\
E[x_{k+1}] &= \Phi_k E[x_k] + 0 && (9)
\end{aligned}
$$

Let $\mu_k = E[x_k]$ Covariance propagation

$$
P_k = E[(x - \mu_k)(x - \mu_k)^T] \tag{10}
$$

Hence

$$
\begin{aligned}
P_{k+1} &= E[(x_{k+1} - \mu_{k+1})(x_{k+1} - \mu_{k+1})^T] && (11)\\
&= E[\Phi_k(x_k - \mu_k + \omega_k)(\Phi_k(x_k - \mu_k + \omega_k))^T] && (12)\\
&= E[\Phi_k(x_k - \mu_k)(x_k - \mu_k)^T \Phi_k^T + \omega_k \omega_k^T + \Phi_k(x_k - \mu_k)\omega_k^T && (13)\\
&+ \omega_k(x_k - \mu_k)^T \Phi_k^T]\\
P_{k+1} &= \Phi_k P_k \Phi_k^T + Q_k && (14)
\end{aligned}
$$

where $E[w_k w_k^T] = Q_k$ and noting that $E[\omega_k] = 0$

$$\hat{x}_{k+1} = \Phi_k x_k + L_k(y_k - H_k \hat{x}_k^-) \tag{15}$$

Let $e_k^+ = x_k - \hat{x}_k^+$

$$e_k^+ = x_k - \hat{x}_k^- - L_k(y_k - H_k \hat{x}_k^-) \tag{16}$$

$$= x_k - \hat{x}_k^- - L_k(H_k e_k^- + v_k) \tag{17}$$

$$= (I - L_k H_k) e_k^- - L_k v_k \tag{18}$$

From the above and utilizing the predictive error covariance matrix, we get

$$P_k^+ = (I - L_k H_k) P_k^- (I - L_k H_k)^T + L_k R_k L_k^T \tag{19}$$

To compute the optimal gain L_k we minimize $trace(P_k^+)$ wrt L_k. To do this, solve $\frac{\partial trace(P_k^+)}{\partial L_k} = 0$ for L_k

- Well what does optimal mean?
- What if L_k was chosen to minimize the error variance
- The error variance is contained in the diagonal of P_k^+
- So, minimize trace of P_k^+

$$\begin{aligned} tr(P^+) &= tr(P^-) - tr(LHP^-) - tr(P^- H^T L^T) + tr(L(HP^- H^T + R)L^T) \\ &= tr(P^-) - 2tr(LHP^-) + tr(L(HP^- H^T + R)L^T) \end{aligned}$$

For a minimum, need $\frac{\partial P^+}{\partial L} = 0$

Kalman Gain

$$\frac{\partial}{\partial L} tr(P^+) = -2tr((HP^-)^T) + 2tr(L(HP^-H^T + R))$$

Setting $\frac{\partial P^+}{\partial L} = 0$ we get the Kalman gain

Kalman gain

$$L_{kalman} = P^- H^T (HP^- H^T + R)^{-1}$$

- The Kalman filter has two steps:
- Prediction step:
    - In this step we predict forward the state of the system using our model and Q
    - This is our best guess of what the system state would look like
    - But it will deviate from the true state if the system evolves in a different manner than we expecte
- Correction step: To ensure that our predictions do not drift for too long, the KF utilizes the idea of feedback corrections
    - In this step we correct our predicted state using feedback between predicted measurement and the actual sensor measurement
    - The correction brings our prediction back on track, without having to have information about all the states, or doing it all the time
- Together the predict-correct framework leads to a robust state estimation technique

Kalman filtering algorithm mathematical specifics

Initialize $x_0 \sim \mathbb{N}(0, P_0)$

Prediction step

$$x_k^- = \Phi_k x_{k-1}$$
$$P_k^- = \Phi_k P_{k-1} \Phi_k^T + Q_k$$

Correction step

$$e_k = y_k - H_k x_k^-$$
$$S_k = H_k P_k^- H_k^T + R_k$$
$$L_k = P_k^- H_k^T S_k^{-1}$$
$$x_k^+ = x_k^- + L_k e_k$$
$$P_k^+ = P_k^- - L_k S_k L_k^T$$

If we assume that A, C, Q, R are continuous-time counterparts:

KF algorithm

Prediction step Initialize $\hat{x}^-(t) = \hat{x}(t = k), P(t) = P_k(t = k)$

$$\dot{\hat{x}}^- = A\hat{x}$$

$$\dot{P}^- = AP + PA^T + Q$$

Correction step

$$P_k^- = P(t = k) \tag{20}$$

$$L_k = P_k^- C^T (C P_k^- C^T + R)^{-1} \tag{21}$$

$$x_k = x_k^- + L(y_k - C x_k^-) \tag{22}$$

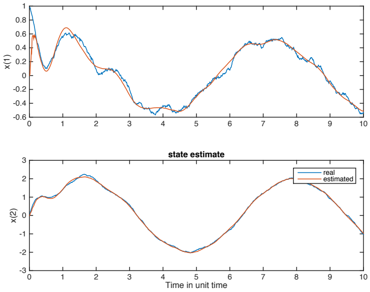
$$P_k = [I - L_k C] P^-(k) \tag{23}$$

- Note that P_k does not depend on x_k, this is a direct consequence of the linearity and Gaussian noise assumption
- This means we can pre-compute K_k off line by iteratively solving (1) and (2) until they converge

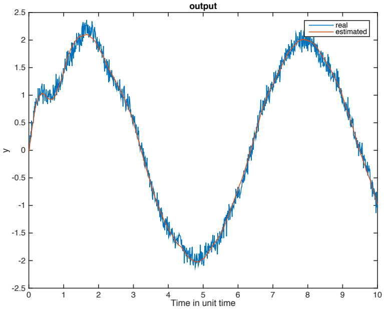
Outline

$$A = \begin{bmatrix} -1 & -5 \\ 6 & -1 \end{bmatrix}$$
$$C = \begin{bmatrix} 0 & 1 \end{bmatrix}$$

- Matlab code `KF_simple.m`
- The measurement noise variance is 0.1 for both states
- The process noise variance is 0.01 for both states



(a) States



(b) Output

What if the noise is non-Gaussian or the system is non-linear?

- If the system dynamics are non-linear but sufficiently smooth, then we can try local linearization
- This leads to the Extended Kalman Filter (Linearize process and measurement noise at every time-step)
- If the dynamics are not sufficiently smooth, or the noise is non-Gaussian, we can utilize Particle Filters (Compute first/second moment of a set of particles)
- The idea here is to create a cloud of particles, transform them through the dynamics and noise, and then re-compute the mean and variance at the other side
- A smart way of doing this is known as the Unscented Kalman Filter (Julier and Uhlmann, 1997)

A very Bayesian derivation of the Kalman Filter

Discriminative approach:

$$P_k^+ = (I - L_k)P_k^-(I - L_k)^T + L_k R_k L_k^T$$

The optimal Kalman Gain is found by solving: $\min_{L_K} trace(P_k^+)$

$$L_k = P_k^- H_k^T (H_k P_k^- H_k^T + R)^{-1}$$

Bayesian approach: Define the Chapman Kolmogrov equation and utilize Bayes theorem We will use Simo Särkkä's excellent presentation on Bayesian derivation of the Kalman Filter:
`http://www.lce.hut.fi/~ssarkka/course_k2011/pdf/handout3.pdf`

Decision Theory

- Target variable t, input variable x, the joint probability distribution $p(x, t)$ provides a complete summary of the uncertainty
- determining $p(x, t)$ from data requires: perception, inference, and knowledge representation
- Decision theory is concerned with making decisions using a prediction of t,
- Take actions based on what values t is likely to take

Example

• How to assign exploration vehicles, and target-vehicles when the distribution of the underlying environment is unknown • Similar problems in operations, or logistics?

- Goal: make the “best” decision in face of uncertainty
- Decisions based on classification: avoiding misclassification
- Decisions based on “utility”, or reward or loss, maximizing expected reward
- The reject option: involving the “Oracle”

Minimizing misclassification

- Consider a case in which targets are of two classes C_1 and C_2
- decision needs to be made on whether the action vehicle should be deployed
- Rule: Deploy action vehicle if the target is of class C_2
- probability of a class given data x using Bayes rule:

$$p(C_k|\mathrm{x}) = \frac{p(\mathrm{x}|C_k)p(C_k)}{p(\mathrm{x})}$$

- Our goal: Minimize the chance of assigning x to the wrong class
- Approach, choose the class having higher posterior probability

- Simple goal: Make as many few misclassifications as possible
- Divide the input space into decision regions R_k, such that all points in R_k are assigned to class C_k
- Boundary between the regions is the decision surface
- We can compute the probability of making a mistake: happens when we misclassify

Probability of making a mistake (risk)

$$
\begin{aligned}
p(\text{mistake}) &= p(x \in R_1, C_2) + p(x \in R_2, C_1) && (24) \\
&= \int_{R_1} p(x, C_2)dx + \int_{R_2} p(x, C_1)dx && (25)
\end{aligned}
$$

Minimizing misclassification

- We can decide how to assign x to minimize misclassification rate
- To minimize $p(\mathrm{mistake})$: assign x to a class C_1 if $p(x, c_1) > p(x, c_2)$
- but from the product rule $p(x, C_k) = p(C_k|x)p(x)$
- Minimum probability of making a mistake is obtained when each value of x is assigned to the class for which $p(C_k, x)$ (the posterior probability) is the largest

Minimizing Expected Loss

- What if our objective is beyond just minimizing the misclassification
- Our goal is to avoid mistakes due to decisions we took based on the information we have
- It may be more natural to find decision rules that directly try to minimize (maximize) the penalty (reward) of making the wrong (right) choice
- How to make decision wrt to the *consequence* of the decision
- One approach: Loss function L, also referred to as a cost function
- The same as the negative of the reward function, or the utility function

- For our example, let loss $L_{k,j}$ be the loss experienced due to assigning target to class k when it is of type j (j could be equal to k)
- We can assign the values to each combinations through a loss function or a loss matrix (e.g. page 41 Bishop)
- Optimal solution minimizes the loss, but the actual loss function depends on the true class, which we don't know
- Approach: can try to optimize *on the average*, minimize the average loss

Expected Loss

$$\mathbb{E}[L] = \sum_k \sum_j \int_{R_j} L_{kj} p(\mathrm{x}, C_k) dx$$

- Choose region R_j to minimize expected loss

- For each x we should minimize $\sum_k L_{kj} p(x, C_k)$

- from product rule, $p(x, C_k) = p(C_k|x)p(x)$ we get the expected loss minimizing decision rule is the one that assigns each new x to the class j for which the quantity
$$\sum_k L_{kj} p(C_k|x)$$
is minimized

- The reject option: Defer the decisions on things that the machine cannot easily distinguish between to an "Oracle"
- e.g. to a human expert

Reading assignment: How to Grow A Mind, Joshua B. Tenenbaum, et al., Science 331, 1279 (2011);
Also watch Josh's Posner lecture at NIPS:
http://videolectures.net/nips2010_tenenbaum_hgm/

- Murphy's Chapter 3, and Josh's thesis
- How do people learn new concepts?, How do children learn concepts?
- Simple example: Concept learning: does a particular item belong to a category?
- Humans make strong generalization from just few examples

- Number game: Assume we are concerned only with integers between 0 and 100
- I tell you that one (positive) example of the concept is: “16”, what is the concept?

- Number game: Assume we are concerned only with integers between 0 and 100
- I tell you that one (positive) example of the concept is: “16”, what is the concept?
- other positive examples are 8, 2, 64
- What do you think the concept is now?

- Number game: Assume we are concerned only with integers between 0 and 100
- I tell you that one (positive) example of the concept is: "16", what is the concept?
- other positive examples are 8, 2, 64
- What do you think the concept is now?
- We are narrowing down the hypothesis from a hypothesis space $\mathcal{H}$
- Why is powers of two more likely than even numbers or numbers between 1 and 70?

- The Occam's razor: avoiding suspicious coincidence
- The likelihood ratio: humans prefer models that are more likely to be true
- **Strong Sampling Assumption**: Examples are drawn uniformly from the extension of a concept (hypothesis h), e.g. numbers ending with 3
- The probability of independently sampling N items from a hypothesis h is:

$$p(D|h) = [\frac{1}{\mathrm{size}(h)}]^N \tag{26}$$

- The model favors the simples (smallest) hypothesis consistent with the data $\Rightarrow$ **Occam's razor**

- Prior: If $D = \{16, 8, 2, 64\}$, what is the hypothesis?

- Prior: If $D = \{16, 8, 2, 64\}$, what is the hypothesis?
- Why is it hard to guess (it has lower likelihood than powers of two): powers of two except 32? $\Rightarrow$ conceptually unnatural
- This is the influence of the prior: but, the prior can be very subjective

- Posterior: Just the likelihood times the prior
- So "powers of two except 32" has low posterior support, despite having high likelihood due to low prior
- Odd numbers has low posterior support, despite having high prior due to low likelihood
- the **aha** moment comes when the learner's likelihood dominates the posterior
- The low prior on unnatural concept prevents overlearning

Maximum A-Posteriori Estimates

- When we have *enough* data, the posterior p(h—D) becomes peaked on a single concept ⇒ the MAP estimate

$$p(h|D) \rightarrow \delta_{\hat{h}^{\mathrm{MAP}}}(h) \tag{27}$$

- $\hat{h}^{\mathrm{MAP}}$ is the posterior mode, and δ is the Dirac measure.
- the MAP estimate can be written as

$$\hat{h}^{\mathrm{MAP}} = \arg\max_h p(D|h)(h) = \arg\max_h [\log p(D|h) + logp(h)] \tag{28}$$

- Since the likelihood depends exponentially on N, and prior is constant, the MAP estimate converges to the Maximum Likelihood Estimate (MLE):

$$\hat{h}^{\mathrm{mle}} = \arg\max_h p(D|h) = \arg\max_h logp(D|h) \tag{29}$$

- I.e. when we have enough data, the data overwhelms the prior!
- If the true hypothesis is in our hypo space, then MAP or MLE will converge to it
- Hence, Bayesian estimators are consistent in the limit
- Generative Modeling: Learning a predictive distribution of the posterior
- I.e. Learn the model to predict the distribution of the observed data, not the specific observations per-say

- Murphy's chapter 3 then goes on to provide some examples of using Bayesian inference on toy problems:
- The coin toss problem with a Beta-Binomial model: Where the likelihood is bionomial, and the posterior is Beta
- The nice property of the choice of Beta as a prior here is that the posterior has the same form as the prior (Beta) $\Rightarrow$ Conjugate Prior
- Hyperparameters: The parameters of the prior, these are learned from data

- Murphy then illustrates the use of the Dirichlet-Multinomial model for the dice throw example
- Here the likelihood is multinomial
- The Conjugate prior here is Dirichlet, since it has support over the K-dimensional probability simplex defined by the six possible outcomes
- So the posterior will also be Dirichlet

Logistics

- Required reading for this week:
- Chapter 2 of Bishop
- Chapters 2 and 3 from Murphy
- Optional readings: Shannon's seminal paper, Kelly's classic paper

References I

[1] Thomas M. Cover and Joy A. Thomas. *Elements of information theory*. Hoboken, NJ: Wiley-Interscience, 2006.

[2] Irving J Good. Studies in the history of probability and statistics. xxxvii am turing's statistical work in world war ii. *Biometrika*, pages 393–396, 1979.

[3] Claude Elwood Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423, 623–656, 2001.