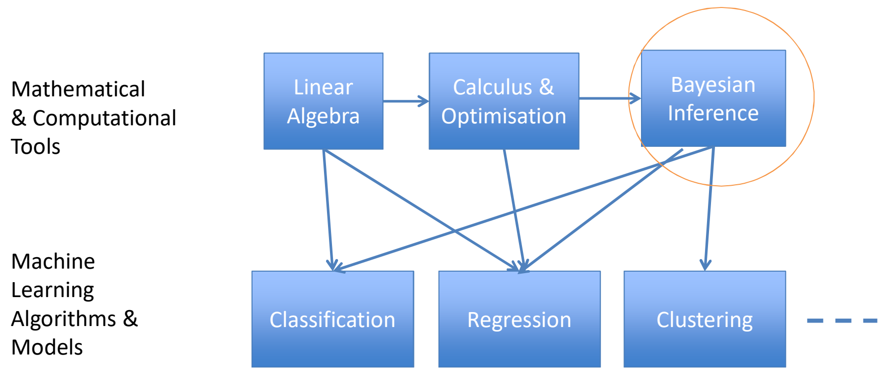
# Machine Learning
# Lecture 3 - Bayesian Inference

Dr. Ioannis Patras
EECS, QMUL 2015

Slides acknowledgments: Dr. Tim Hospedales

## Some Context



# Bayesian Inference

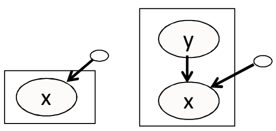
What is Bayesian inference about?

- Using Bayes' theorem to combine observations and prior belief in a rational way for inference and decision making
- Dealing rationally with unknown quantities, by summing/integrating over their values (law of total probability)

# Probabilistic Machine Learning

Canonical Problems:

- Inference: $p(y \mid x) = p(x \mid y) p(y) / p(x)$
- Marginalization: $p(x) = \int p(x, y) dy$
- ML/MAP Learning $\hat{\theta} = \mathrm{argmax}\ p(X \mid \theta)$
  – Density Estimation:
- EM Learning: $\hat{\theta} = \mathrm{argmax} \int p(X, Y \mid \theta) dY$
- Model Selection: $M = \mathrm{argmax} \int p(X, Y, \theta \mid M) p(M) dY d\theta$



# Bayesian Inference

- Agent infers the process that generated some data, *d*
- *h* is the hypothesis about this process
- *P(h)* is the probability that the agent would have ascribed to the hypothesis BEFORE seeing the data d. This is the **prior probability**
- How should the agent go about changing his beliefs in the light of the evidence provided by *d?*
- To answer this we wish to compute the **posterior probability** P(h|d).

# Bayes' theorem

Such a procedure is given by Bayes' theorem

$$P(h \mid d) = \frac{P(d \mid h)P(h)}{P(d)}$$

The denominator is given by summing over hypotheses (marginalization as we saw before)
H is the set of all hypotheses considered by the agent, the hypothesis space.

$$P(d) = \sum_{h' \in H} P(d \mid h')P(h')$$

Posterior probability

Likelihood

Prior probability

$$p(h \mid d) = \frac{p(d \mid h)\, p(h)}{\sum_{h' \in H} p(d \mid h')\, p(h')}$$

Sum over space of alternative hypotheses

# Bayesian Probabilities

- Probabilities are now thought of as degrees of belief, and not frequencies…
  - Shakespeare's plays are written by Francis Bacon
  - Signature on a cheque is genuine
- Bayesian reasoning is influenced by priors, but does not require them to do interesting things.
  - E.g., Bayesian Occam's razor (later)

## Outline

- Hypothesis comparison and likelihood ratios
- Estimation and Inference of model parameters
- Posterior predictive and model averaging
- Complexity control and Bayesian Occam's razor
- Bayesian decision theory and pattern recognition
- Some Bayesian classifiers
- Conclusions

## Compare two simple hypotheses

- A box contains two coins
- One produces heads 50% of the time
- One produces heads 90 % of the time
- You choose a coin and flip it 10 times producing
  - HHHHHHHHHH
- Which coin did you pick?
- How would you change your belief if you'd thrown HHTHTHTTHT instead?

# Compare two simple hypotheses

- To translate into a Bayesian inference problem, must specify
  - The hypothesis space H
  - The prior distribution P(h)
  - The likelihood P(d|h)
- Two coins => two natural hypotheses.
- Let θ denote probability that coin produces heads.
  - $h_o$ is the hypothesis that $\theta = 0.5$
  - $h_1$ is the hypothesis that $\theta = 0.9$
- Assuming we randomly pick a coin:, $P(h_o)=P(h_1) = 0.5$

# Compare two simple hypotheses

- Observed Data: d=HHHHHHHHHH
- Now we must specify the likelihood $P(d|\theta)$
- What is the probability of producing a sequence of coin flips containing $N_h$ heads and $N_t$ tails by a coin with heads probability θ?

# Bernoulli distribution

- Bernoulli distribution gives the probability of one trial F: {0,1}

$$P(Flip \mid \theta) = \theta^{F} (1-\theta)^{(1-F)}$$

- Coin flips are independent events:
  - Therefore probability of a sequence is product of individual event probabilities.
- Given: Heads probability θ.
  - Probability of sequence d=HHTH... with $N_h$ heads and $N_t$ tails:

$$P(d \mid \theta) = \theta^{N_H} (1-\theta)^{N_T}$$

# Likelihoods associated with $h_o$ and $h_1$

- Substitute 0.5 and 0.9 into θ to get the likelihoods of the two hypotheses.

$$P(d \mid \theta) = \theta^{N_H} (1-\theta)^{N_T}$$

- Then the priors and likelihoods can be placed into the Bayes' equation to compute the posterior probabilities of each hypothesis.

Finding the best hypothesis to explain the data….

$$h^* = \operatorname{argmax}_{h \in H} p(h \mid d)$$

$$h^* = \operatorname{argmax}_{h \in H} \frac{P(d \mid h) P(h)}{P(d)}$$

In the coin example there are two possible hypotheses….

$$P(h_1 \mid d) = \frac{P(d \mid h_1) P(h_1)}{P(d)} \qquad P(h_1 \mid d) = \frac{0.9^{N_H} (1 - 0.9)^{N_T} 0.5}{P(d)}$$

$$P(h_0 \mid d) = \frac{P(d \mid h_0) P(h_0)}{P(d)} \qquad P(h_0 \mid d) = \frac{0.5^{N_H} (1 - 0.5)^{N_T} 0.5}{P(d)}$$

With two hypotheses its easier to just consider the ratio of the posterior probabilities because p(d) is the same in both equations. This gives the form.

$$\frac{P(h_1 \mid d)}{P(h_o \mid d)} = \frac{P(d \mid h_1)}{P(d \mid h_o)} \frac{P(h_1)}{P(h_o)} \qquad \frac{P(h_1 \mid d)}{P(h_o \mid d)} = \frac{0.9^{N_H} (1 - 0.9)^{N_T}}{0.5^{N_H} (1 - 0.5)^{N_T}} \frac{0.5}{0.5}$$

For $N_h$ = 10, and $H_t$ = 0, we get posterior odds of 357:1 in favour of $h_1$ (biased coin)
For $N_h$ = 5 and $H_t$ = 5, we get posterior odds of 165:1 in favour of $h_0$ (fair coin)

## Summary

- Any time we want to evaluate or choose between two or more hypotheses explaining some data, use Bayes theorem:
  - Choose your prior
  - Fill in your likelihood
  - Divide by the total probability
    - ( May be avoidable )

$$P(h \mid d) = \frac{P(d \mid h)P(h)}{P(d)}$$

## Outline

- Hypothesis comparison and likelihood ratios
- Estimation and Inference of model parameters
- Posterior predictive and model averaging
- Complexity control and Bayesian Occam's razor
- Bayesian decision theory and pattern recognition
- Some Bayesian classifiers
- Conclusions

# Inferring Model Parameters

- So far we chose among alternative hypotheses $H=\{h_o, h_1\}$. $h_o$: θ=0.5, $h_1$: θ=0.9.
- We often want to infer the continuous value of a model parameter. E.g., θ between 0 and 1.
- Suppose we observe d=H, d=HT, or d=HHHHHHHHHH . Estimate θ
- Two ways:
  - ML/MAP: Estimate a value for θ. E.g., θ=0.7.
  - Bayesian: Infer a distribution for θ.
    - This computes a probability for every value of θ.

# ML Estimator

- **ML (Maximum Likelihood)**: choose the value of θ that maximizes the likelihood

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} P(d \mid \theta)$$

- Differentiate

$$P(d \mid \theta) = \theta^{N_H} (1-\theta)^{N_T}$$

  - With respect to θ
  - Set to 0, and solve for θ:
- Answer for coins:
  - $\theta = N_H/(N_H+N_T)$
- A particular $N_T$:$N_H$ ratio will give the same θ ignoring the total number of trials.

# Being Bayesian with Many Hypotheses

- We used Bayes theorem to choose alternative hypotheses H={$h_{o,}$ $h_1$}. $h_o$: θ=0.5, $h_1$: θ=0.9.
  - Now want to infer the continuous value of a model parameter. E.g., θ between 0 and 1.
- θ is now a **random variable** and the posterior distribution is now a **probability density.**

$$P(h \mid d) = \frac{P(d \mid h)P(h)}{P(d)} \quad \Rightarrow \quad p(\theta \mid d) = \frac{P(d \mid \theta)p(\theta)}{P(d)}$$

# From Discrete to Continuous Bayes

$$P(h \mid d) = \frac{P(d \mid h)P(h)}{P(d)} \qquad P(d) = \sum_{h' \in H} P(d \mid h')P(h')$$

$$p(h \mid d) = \frac{p(d \mid h)p(h)}{p(d)} \qquad p(d) = \int P(d \mid h')P(h')dh'$$

Posterior distribution is now a posterior density (lower case p is used for pdf)

# From Discrete to Continuous Bayes

$$p(\theta \mid d) = \frac{p(d \mid \theta)p(\theta)}{p(d)} \qquad p(d) = \int P(d \mid \theta')P(\theta')d\theta'$$

- Posterior over θ contains more information than a single point estimate.
- Probability of every value or range of values of θ.
- Alternative is three possible point estimates:

**ML (Maximum Likelihood)**: choose the value of θ that maximizes the likelihood

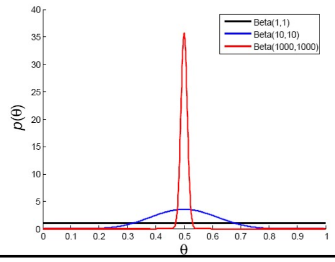
$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} P(d \mid \theta)$$

**MAP (Maximum a posteriori)**: chooses the value of θ that maximizes the posterior probability

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} P(\theta \mid d)$$

# MAP Estimation: Include prior p(θ)

- Different choices of prior p(θ) will result in different guesses of the value of θ.
- p(θ) options:
  - Uniform $\theta$
  - Non-uniform based on prior experience of coins




Convenient to represent by a beta distribution.

## MAP Estimator

- **MAP (Maximum a posteriori)**: chooses the value of θ that maximizes the posterior
- Differentiate
  - With respect to θ
  - Set to 0, and solve for θ:

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} P(\theta \mid d)$$

- How?
  – … We'll come back to it….

## Likelihood and prior

- Likelihood: Bernoulli($\theta$) distribution

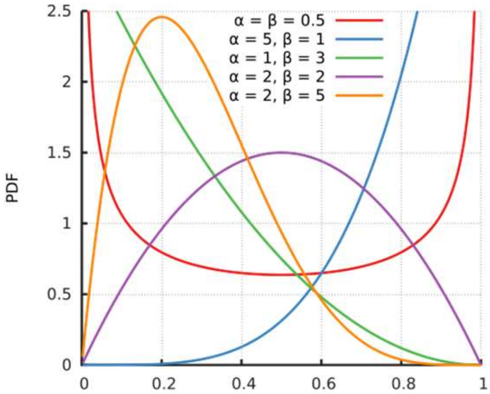
$$P(D \mid \theta) = \theta^{N_H} (1-\theta)^{N_T}$$

  – $N_H$: number of heads observed
  – $N_T$: number of tails observed
- Prior: Beta($F_H$,$F_T$) distribution

$$P(\theta) \propto \theta^{F_H-1} (1-\theta)^{F_T-1}$$

  – $F_H$: fictional observations of heads
  – $F_T$: fictional observations of tails
  – ($F_H$=$F_T$=1 makes Beta behave uniform/uninformative)

# Beta Distribution




Beta distribution has support [0,1]: The same range as Bernoulli/coin parameter.
Different Beta distributions encode different priorsfor the coin parameter.

# Bayesian Solution to coin's $\theta$

$$p(\theta \mid d) = \frac{p(d \mid \theta) p(\theta)}{p(d)}$$

$P(D \mid \theta) = \theta^{N_H} (1-\theta)^{N_T}$

$P(\theta) \propto \theta^{F_H-1} (1-\theta)^{F_T-1}$

$$p(\theta \mid d) \propto \theta^{N_H} (1-\theta)^{N_T} \times \theta^{F_H-1} (1-\theta)^{F_T-1}$$

$$p(\theta \mid d) \propto \theta^{(N_H+F_H-1)} (1-\theta)^{(N_T+F_T-1)}$$

$$= Beta(N_H + F_H, N_T + F_T)$$

Answer: Another (different) beta distribution over $\theta$!

This is a "conjugate prior" trick.

- Beta distribution is conjugate to Bernoulli.

Now we know the probability of any given value of $\theta$ (evaluate the beta distribution)
…and how confident we should be about a particular estimate.

$$p(\theta \mid d) \propto p(d \mid \theta)\, p(\theta) = \theta^{N_H+F_H-1} (1-\theta)^{N_T+F_T-1}$$

# Conjugate Prior Discussion

- This conjugate prior trick is used a lot in Bayesian inference to make it easy and fast.
  - There are tables of which distributions are conjugate
  - …. So you can find the right form of parameter prior for the likelihood of your task
- E.g., Beta-Binomial, Dirichlet-Multinomial, Gussian-Gaussian, etc.

https://en.wikipedia.org/wiki/Conjugate_prior

# Completing the MAP Estimator

- **MAP (Maximum a posteriori)**: chooses the value of θ that maximizes the posterior
- Differentiate
  - With respect to θ
  - Set to 0, and solve for θ:

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} P(\theta \mid d)$$

$$p(\theta \mid d) = Beta(N_H + F_H, N_T + F_T) \quad \theta^{(N_H + F_H - 1)}(1-\theta)^{(N_T + F_T - 1)}$$

– ML (from before) versus MAP solution:

$$\theta = N_H / (N_H + N_T) \qquad \theta = N_H + F_H - 1 / (N_H + N_T + F_H + F_T - 2)$$

| Estimate θ | ML | MAP FH=FT=1 | MAP FH=FT=10 | MAP FT=10,FH=1 |
|---|---|---|---|---|
| d=H | | | | |
| d=HT | | | | |
| d=HHH | | | | |

# Completing the MAP Estimator

- **MAP (Maximum a posteriori)**: chooses the value of θ that maximizes the posterior
- Differentiate
  - With respect to θ
  - Set to 0, and solve for θ:

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} P(\theta \mid d)$$

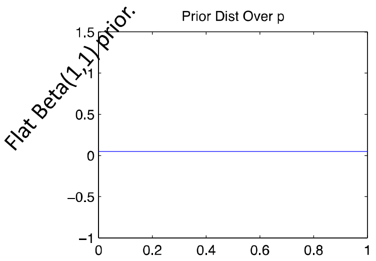
$$p(\theta \mid d) \; = \; Beta(N_H + F_H, N_T + F_T)$$
$$\theta^{(N_H + F_H - 1)} (1-\theta)^{(N_T + F_T - 1)}$$

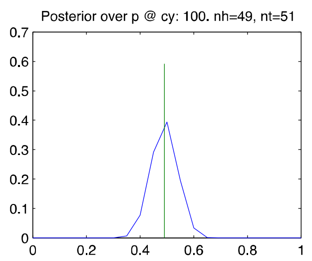
– ML (from before) versus MAP solution:

$$\theta = N_H / (N_H + N_T) \qquad \theta = N_H + F_H - 1/(N_H + N_T + F_H + F_T - 2)$$

| Estimate θ | ML | MAP FH=FT=1 | MAP FH=FT=10 | MAP FT=10,FH=1 |
|---|---|---|---|---|
| d=H | 1 | 1 | 0.53 | 0.1 |
| d=HT | 0.5 | 0.5 | 0.5 | 0.09 |
| d=HHH | 1 | 1 | 0.57 | 0.25 |

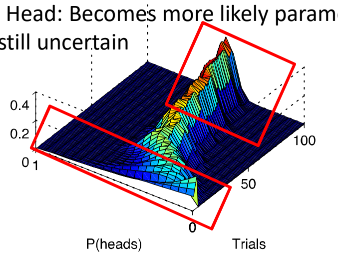
# Illustration of Bayesian Coin Learning




Post surface over time

First Head: Becomes more likely parameter is closer to 1.
But still uncertain



## Bayes versus ML/MAP discussion

- Q: Why do we want to infer the distribution of a quantity rather than pick it's best value?
  - A1. Because we want to know our confidence of the answer
    - And thus decide if to: take action / go out and collect more data / etc
  - A2. Because downstream decisions can be improved by considering every possible value.

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} P(\theta \mid d)$$

versus

$$p(\theta \mid d) = \frac{P(d \mid \theta) P(\theta)}{P(d)}$$

## Summary

- Bayes theorem can be used to evaluate the probability of continuous quantities
  - … Such as model parameters like coin bias
- This requires some care of choice of distributions in order to be exact and efficient
  - ( Otherwise it can also be approximated numerically )
- Conventional ML/MAP estimates pick the maximum of the likelihood or the posterior

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} P(\theta \mid d)$$

$$p(\theta \mid d) = \frac{P(d \mid \theta) P(\theta)}{P(d)}$$

# Outline

- Hypothesis comparison and likelihood ratios
- Estimation and Inference of model parameters
- Posterior predictive and model averaging
- Complexity control and Bayesian Occam's razor
- Bayesian decision theory and pattern recognition
- Some Bayesian classifiers
- Conclusions

# Posterior Predictive Distribution aka Marginal Likelihood

- A common question to ask is of the form:
  - What's the probability of some new data d given old data D.
  - E.g., what is the probability of d=H, given that so far we observed D=HTHH
- We want to know: $p(d = \mathrm{H} \mid D)$
  - Simple solution:
    - Pick the best hypothesis about the model..
    - E.g., Estimate $\theta_{est}$ given D using ML/MAP.
  - Then compute likelihood $p(d|\theta_{est})$

# Posterior Predictive Distribution

- What is the prob. of d=H, given observed D=HTHH and prior encoded by "fake observations" F={$F_H$,$F_T$}
  - Depends on $\theta$ which we don't know
- Bayesian solution: Integrate out the parameter:

$$p(d=\mathrm{H} \mid D, F) = \int_0^1 \mathrm{P}(\mathrm{d}=\mathrm{H} \mid \mathrm{q})\, \mathrm{P}(\mathrm{q} \mid \mathrm{D}, \mathrm{F})\, \mathrm{dq}$$

Aka "hypothesis averaging"

$$P(d = H \mid \theta) = \theta^{H=1}(1-\theta)^{H=0}$$

$$p(\theta \mid D) \propto p(D \mid \theta)\, p(\theta) = \theta^{N_H+F_H-1}(1-\theta)^{N_T+F_T-1}$$

$$P(d = \mathrm{H} \mid D, F_H, F_T) = \frac{(N_H+F_H)}{(N_H+F_H+N_T+F_T)}$$

# Posterior Predictive Distribution

- What is the prob. of d=H, given that we observed D=HTHH
  - Depends on $\theta$ which we don't know
- Bayesian solution: Integrate out the parameter.
  - For bernoulli (coin) distributions

$$p(d=\mathrm{H} \mid D, F) = \int_0^1 \mathrm{P}(\mathrm{d}=\mathrm{H} \mid \theta)\, \mathrm{P}(\theta \mid \mathrm{D}, \mathrm{F})\, \mathrm{d}\theta$$

$$= \frac{(N_H+F_H)}{(N_H+F_H+N_T+F_T)}$$

- Takes into account the infinite set of possible $\theta$, and the posterior probability of each
  - Unlike ML/MAP

# Example: coin fresh from bank

- e.g., *F* ={1000 heads, 1000 tails} ~ strong expectation that any new coin will be fair
  - After seeing 4 heads, 6 tails, *P*(H) on next flip = 1004 / (1004+1006) = 49.95%
- Compare: *F* ={3 heads, 3 tails} ~ weak expectation that any new coin will be fair
  - After seeing 4 heads, 6 tails, *P*(H) on next flip = 7 / (7+9) = 43.75%
- Either large F or D make prediction more confident

# Other Distributions

- There procedure is analogous for other distributions….
  - Bernoulli (coin): Integrate out one coin parameter.
  - Multinomial (dice): Integrate out the 6d dice bias.
  - Gaussian: Integrate out the mean and variance.

## Summary: Posterior Predictive / Marginal Likelihood

- To predict the next observation in a sequence of (IID) data
- ML/MAP approximation: Estimate the model parameters and then make the prediction
- Bayesian solution: Integrate out the model parameters.

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} P(\theta \mid D)$$
$$p(d \mid D) \approx p(d \mid \hat{\theta})$$

$$p(\theta \mid D) = \frac{P(D \mid \theta) P(\theta)}{P(D)}$$
$$p(d \mid D) = \int p(d \mid \theta) p(\theta \mid D) d\theta$$

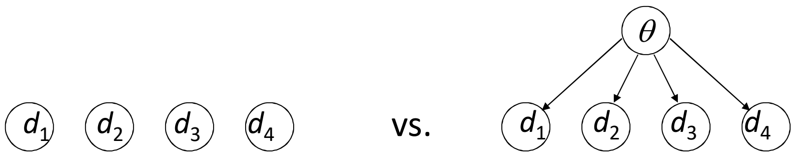
## Outline

- Hypothesis comparison and likelihood ratios
- Estimation and Inference of model parameters
- Posterior predictive and model averaging
- Complexity control and Bayesian Occam's razor
- Bayesian decision theory and pattern recognition
- Some Bayesian classifiers
- Conclusions

# Back to Model Comparison

- Sometimes we want to compare two models/hypotheses of different "complexities"
  - Here Bayesian and non-Bayesian approaches can give **very** different answers!
- Non-Bayesian must resort to "Occam's Razor" heuristics.
- Bayesian approach gives the right answer automatically! ☺
  - "Bayesian Occam's Razor"

# Comparing Simple and Complex Hypothesis (Model Selection)




Fair coin, $P(\mathrm{H}) = 0.5$

$P(\mathrm{H}) = \theta$

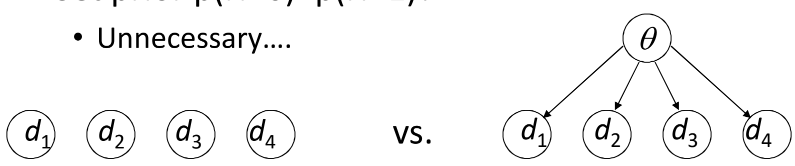
- Which provides a better account of the data:
  - H0: A fair coin P(H)=0.5
  - H1: $P(\mathrm{H}) = \theta$?

## Comparing simple and complex hypotheses: the need for Occam's razor

- $P(\mathtt{H}) = \theta$ is more complex than $P(\mathtt{H}) = 0.5$:
  - $P(\mathtt{H}) = 0.5$ is a special case of $P(\mathtt{H}) = \theta$
  - for any observed sequence $D$, we can choose $\theta$ such that $D$ is more probable than if $P(\mathtt{H}) = 0.5$
  - H0 has zero free parameters.
  - H1 has a free parameter $\theta$.

## ML/MAP gets model comparison wrong

- ML/MAP strategy for comparing p(H=1) p(H=0)
  - Estimate $\theta_{est}$ from data using ML/MAP
  - Likelihood ratio p(d|H=1, $\theta_{est}$)/p(d|H=0) to decide
- H=1 will always win!
- How to fix?
  - Set prior p(H=0)>p(H=1)?
    - Unnecessary….




Fair coin, $P(\mathtt{H}) = 0.5$ $P(\mathtt{H}) = \theta$

# Bayesian Model Comparison

$$\frac{P(h_1/D)}{P(h_0/D)} = \frac{P(D/h_1)}{P(D/h_0)} \times \frac{P(h_1)}{P(h_0)}$$

$$P(D \mid h_0) = (1/2)^n (1-1/2)^{N-n} = 1/2^N$$

$$P(D \mid h_1) = \int_0^1 P(D \mid \theta, h_1) p(\theta \mid h_1) d\theta$$

- Don't select the unknown parameter, integrate it.
- This is "marginal likelihood": Learned in previous section!
  - ~ "The probability that *randomly selected* parameters would generate the data"


# Bayesian Model Comparison

$$\frac{P(h_1/D)}{P(h_0/D)} = \frac{P(D/h_1)}{P(D/h_0)} \times \frac{P(h_1)}{P(h_0)}$$

$$P(D \mid h_0) = (1/2)^n (1-1/2)^{N-n} = 1/2^N$$

$$P(D \mid h_1) = \int_0^1 P(D \mid \theta, h_1) p(\theta \mid h_1) d\theta$$

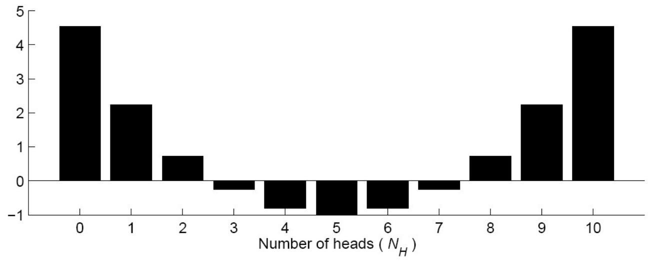

Averaging over all possible values of θ penalizes "overfitted" hypotheses because only a small range fits data well


- Don't select the unknown parameter, integrate it.
- This is "marginal likelihood": Learned in previous section!
  - ~ "The probability that *randomly selected* parameters would generate the data"

# Bayesian Model Comparison

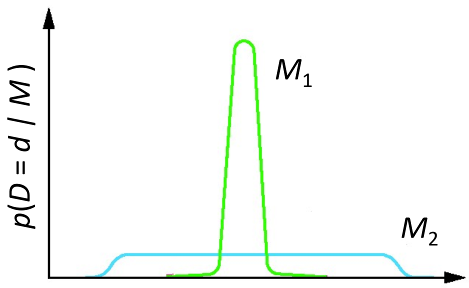
$$\log \frac{P(D \mid h_1)}{P(D \mid h_0)}$$

Log posterior odds in favour of $h_1$




...Why does this work?

# Bayesian Occam's Razor

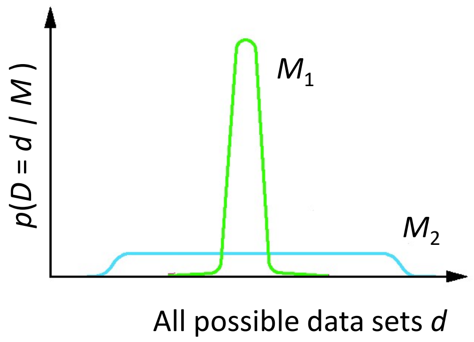



All possible data sets *d*

For any model *M*, $\sum_{\text{all } d \in D} p(D = d \mid M) = 1$

Law of *"conservation of belief"*: A model that can predict many possible data sets must assign each of them low probability.

# Bayesian Occam's Razor




Only predicts a few datasets.
(E.g., not D=HHHHH)
Must give those few higher probability.

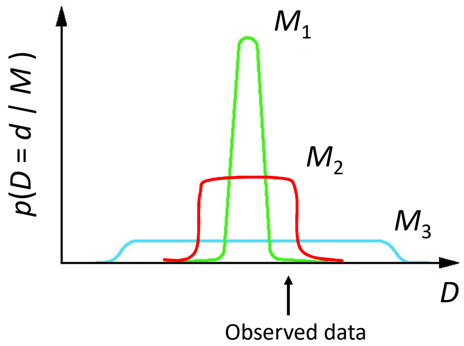
H0: A fair coin P(H)=0.5
H1: *P*(H) = $\theta$ ?

Can predict any dataset.
(By varying $\theta$)
Must give each of them low probability.

For any model *M*, $\sum_{\text{all } d \in D} p(D = d \mid M) = 1$

Law of *"conservation of belief"*: A model that can predict many possible data sets must assign each of them low probability.

$$\sum_{\text{all } d \in D} p(D = d \mid M) = 1$$




$M_1$: A model that is *too simple* is unlikely to generate the data.
$M_3$: A model that is *too complex* can generate many possible data sets, so it is unlikely to generate this particular data set at random.

## Summary

- Comparing hypothesis of differing complexity:
- ML/MAP:
  - Likelihood ratio, after estimating the parameters of each model to compare
  - => Wrong answer.
  - You may see heuristics to ameliorate this
- Bayes:
  - Likelihood ratio, integrating out the parameters of each model to compare
  - => Optimal answer.
  - One of the most powerful capabilities of Bayesian over non-Bayesian models

## Outline

- Hypothesis comparison and likelihood ratios
- Estimation and Inference of model parameters
- Posterior predictive and model averaging
- Complexity control and Bayesian Occam's razor
- Bayesian decision theory and pattern recognition
- Some Bayesian classifiers
- Conclusions

## Bayesian Decision Theory: Inference applied to pattern recognition

- Use what we've learned about Bayesian inference to do some pattern recognition machine learning.
  – Consider classifying fish: Sea bass and Salmon
- True fish type is a random variable.
- Define w as the type of fish we observe:
  – W = $w_1$ for seabass,
  – w = $w_2$ for salmon.
  – $P(w_1)$ is the *a priori probability* of bass.
  – $P(w_2)$ is the *a priori probability* of salmon.

## Bayesian Decision Theory: Inference applied to pattern recognition

- Prior probabilities reflect our knowledge of how likely each type of fish will appear before we actually see it.
  – How to choose $P(w_1)$ and $P(w_2)$?
  - Set $P(w_1) = P(w_2)$ if equiprobable (*uniform priors*).
  – May use different values depending on the fishing area, time of the year, etc.
- Assume there are no other types of fish
  – $P(w_1) + P(w_2) = 1$ (exclusivity and exhaustivity).

# Making a Decision

- If only prior information….?

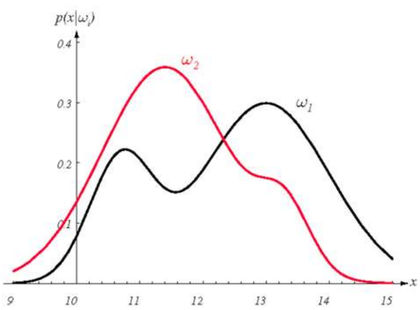
$$\text{Decide} \begin{cases} w_1 & \text{if } P(w_1) > P(w_2) \\ w_2 & \text{otherwise} \end{cases}$$

- We can compute the probability of error of this decision:

$$P(error) = \min\{P(w_1), P(w_2)\}$$

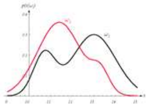
# Exploiting Measurements

- Suppose we also have a lightness measurement x.
  - x is a continuous random variable.
  - Likelihood p(x|w) is probability of measurement x given fish type w
  - P(x|w1) and p(x|w2) describe the difference in lightness between bass and salmo



## Posterior Probabilities for Decisions

- Now we can use Bayes formula to build a fish recognizer by combining prior and posterior probabilities



$$P(w_j|x) = \frac{p(x|w_j)P(w_j)}{p(x)}$$

$$p(x) = \sum_{j=1}^{2} p(x|w_j)P(w_j).$$

## Posterior Probabilities for Decisions

- Prior-only decision was

$$\text{Decide} \begin{cases} w_1 & \text{if } P(w_1) > P(w_2) \\ w_2 & \text{otherwise} \end{cases}$$

- Now with observation:

$$\text{Decide} \begin{cases} w_1 & \text{if } P(w_1|x) > P(w_2|x) \\ w_2 & \text{otherwise} \end{cases} \qquad \begin{cases} w_1 & \text{if } \frac{p(x|w_1)}{p(x|w_2)} > \frac{P(w_2)}{P(w_1)} \\ w_2 & \text{otherwise} \end{cases}$$

- Error probability:

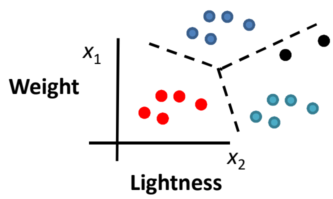
$$P(error|x) = \min\{P(w_1|x), P(w_2|x)\}.$$

## Generalizing to more Realistic Cases

- More than one feature: scalar *x* -> vector **x**
  - E.g.,
  - Gaussian likelihoods => multivariate Gaussian
  - Bernoulli likelihoods => Multinomial likelihoods
- Multiple states/categories: straightforward
  - Pick the most probable of many hypothesis, rather than of only two hypotheses.

## Discriminant Functions

- A useful way of representing classifiers:
  - Discriminant functions: $g_i(x)$ where the classifier assigns x to class $w_i$ if: $g_i(\mathbf{x}) > g_j(\mathbf{x}) \quad \forall j \neq i.$
- Discriminant functions divide the space x into decision regions separated by decision boundaries



# Discriminant Functions and Bayes

- Discriminant functions can arise from Bayes rule:
  - Most probable hypothesis:

$$P(w_j|x) = \frac{p(x|w_j)P(w_j)}{p(x)} \Rightarrow w_i \text{ if } \quad g_i(\mathbf{x}) > g_j(\mathbf{x}) \quad \forall j \neq i.$$

where

$$g_i(\mathbf{x}) = \ln p(\mathbf{x}|w_i) + \ln P(w_i).$$

Aside: Taking logs is a common trick in ML.
Log(x) is a monotonic function of x.
So for "pick the largest" it doesn't matter if log(x) or x.

# Discriminant Functions and Bayes

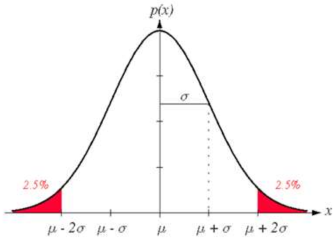
- Discriminant functions can arise from Bayes rule:

$$P(w_j|x) = \frac{p(x|w_j)P(w_j)}{p(x)} \Rightarrow w_i \text{ if } \quad g_i(\mathbf{x}) > g_j(\mathbf{x}) \quad \forall j \neq i.$$

$$g_i(\mathbf{x}) = \ln p(\mathbf{x}|w_i) + \ln P(w_i).$$

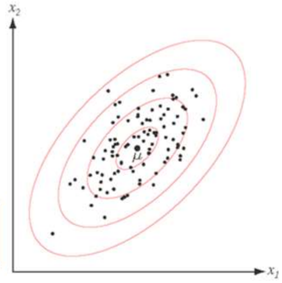
- Let's look at the specific discriminant functions that arise for common likelihoods *p(x|w)*
  - Gaussian for continuous data
  - Binomial/multinomial for discrete data

# Gaussian Densities




For $x \in \mathbb{R}$:

$$p(x) = N(\mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right]$$




For $\mathbf{x} \in \mathbb{R}^d$:

$$p(\mathbf{x}) = N(\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right]$$

# Discriminant Functions for Gaussians

$$P(w_j|x) = \frac{p(x|w_j)P(w_j)}{p(x)}$$

$w_i$ if $g_i(\mathbf{x}) > g_j(\mathbf{x}) \quad \forall j \neq i.$

- General Bayes classifier: $g_i(\mathbf{x}) = \ln p(\mathbf{x}|w_i) + \ln P(w_i).$
- Gaussian classifier:

For $p(\mathbf{x}|w_i) = N(\boldsymbol{\mu_i}, \boldsymbol{\Sigma_i})$

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu_i})^T \boldsymbol{\Sigma_i}^{-1}(\mathbf{x}-\boldsymbol{\mu_i}) - \frac{d}{2}\ln 2\pi - \frac{1}{2}\ln|\boldsymbol{\Sigma_i}| + \ln P(w_i).$$

# Discriminant Functions for Gaussians

$$g_i(\mathbf{x}) = \ln p(\mathbf{x}|w_i) + \ln P(w_i). \qquad \text{For } p(\mathbf{x}|w_i) = N(\boldsymbol{\mu_i}, \boldsymbol{\Sigma_i})$$

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu_i})^T \boldsymbol{\Sigma}_i^{-1}(\mathbf{x}-\boldsymbol{\mu_i}) - \frac{d}{2}\ln 2\pi - \frac{1}{2}\ln|\boldsymbol{\Sigma_i}| + \ln P(w_i).$$

- A classifier's decision boundary is the line it believes separates the classes.
  - The line where $p(w_1|x)=p(w_2|x)$
  - or equivalently where $g_1(x)=g_2(x)$

- How does it look for this Gaussian classifier?

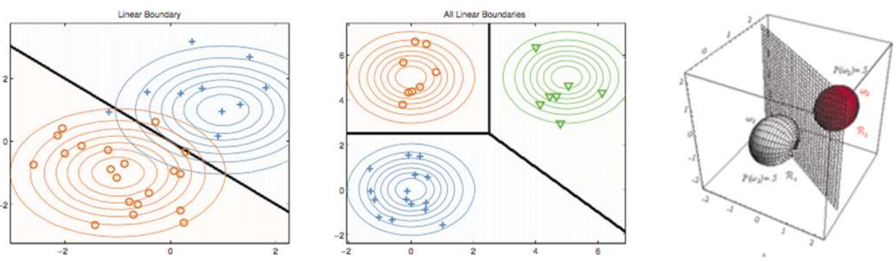
# Discriminant Functions for Gaussians

$$g_i(\mathbf{x}) = \ln p(\mathbf{x}|w_i) + \ln P(w_i). \qquad \text{For } p(\mathbf{x}|w_i) = N(\boldsymbol{\mu_i}, \boldsymbol{\Sigma_i})$$

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu_i})^T \boldsymbol{\Sigma}_i^{-1}(\mathbf{x}-\boldsymbol{\mu_i}) - \frac{d}{2}\ln 2\pi - \frac{1}{2}\ln|\boldsymbol{\Sigma_i}| + \ln P(w_i).$$

Decision boundary depends on the Gaussian covariance.

- Same covariance $\Sigma_i = \Sigma$. Classifer is linear



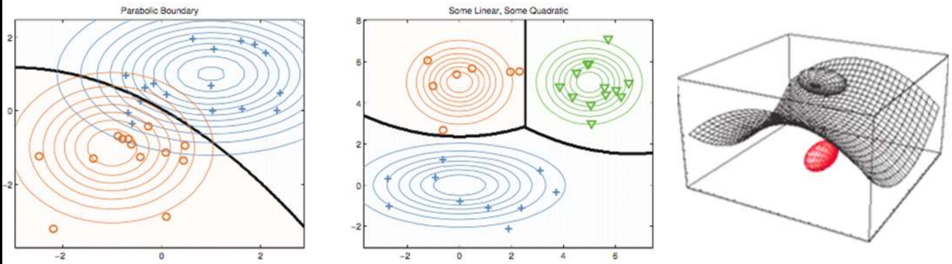
## Discriminant Functions for Gaussians

$g_i(\mathbf{x}) = \ln p(\mathbf{x}|w_i) + \ln P(w_i).$ For $p(\mathbf{x}|w_i) = N(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i) - \frac{d}{2}\ln 2\pi - \frac{1}{2}\ln|\boldsymbol{\Sigma}_i| + \ln P(w_i).$$

Decision boundary depends on the Gaussian covariance.

- General covariance: Classifier is hyperquadratic




## Summary

- We can build classifiers for pattern recognition using Bayes theorem.
  - If we know the different likelihood for different categories of data => Bayes gives us a classifier.
- The contour where the two classes' posteriors are equal is the decision boundary.
- Next: How to get the likelihoods.

## Outline

- Hypothesis comparison and likelihood ratios
- Estimation and Inference of model parameters
- Posterior predictive and model averaging
- Complexity control and Bayesian Occam's razor
- Bayesian decision theory and pattern recognition
- Some Bayesian classifiers
- Conclusions

## Probabilistic Machine Learning

Classic Problems:

- Inference $p(y \mid x) = p(x \mid y)p(y) / p(x)$
- Marginal Likelihood $p(x) = \int p(x, y)dy$
- ML Learning, $\hat{\theta} = \text{argmax}\ p(X \mid \theta)$
  - Density Estimation
- EM Learning $\hat{\theta} = \text{argmax} \int p(X, Y \mid \theta)dY$
- Model Selection $M = \text{argmax} \int p(X, Y, \theta \mid M)p(M)dYd\theta$

## Conclusions

- So far we looked at exact methods for Bayesian inference
  - Analytical posteriors (E.g., Bernoulli-Beta)
  - Integration to find marginal likelihoods
- For many practical problems this is intractable and we resort to approximate methods:
- Deterministic approximations
  - Variational methods
- Stochastic approximations
  - Markov chain monte carlo (MCMC)

## Conclusions

- We saw the key Bayesian computations: Bayes theorem, conjugate priors for parameter inference, and marginalization.
  - Bayes theorem: Leads to hypothesis comparison, and classifiers
  - With conjugate priors, allows model parameter inference (and not just estimation or selection)
  - With marginalization, allows predictive distributions, and model comparison across complexities (Bayesian Occam's razor)

# Learning Outcomes

- You should:
  - Be able to use Bayes theorem
  - Appreciate the limitations of ML versus MAP for parameter estimation
  - Appreciate the benefits of Bayesian parameter inference over estimation
  - Understand the significance of marginal likelihood/posterior predictive distributions in Bayesian inference
  - Understand how Bayesian Occam's allows hypothesis comparison across complexities
  - Know how to build a simple classifier from Gaussian/Bernoulli likelihoods and Bayes theorem