Lecture 7

Markov Chain
Hidden Markov Models
(Markov Random Fields)

- Markov Chains
  - Gibbs Sampling
- Hidden Markov Models
  - State Estimation,
  - Prediction,
  - Smoothing,
  - Most Probable Path
- (Markov Random Fields)

Background

In dynamic systems:

State Estimation – Estimating the current state of the system given current knowledge.

Prediction – Estimating future state(s) of the system given current knowledge.

Smoothing – Estimating prior states of the system given current knowledge.

Background

Independence

$$P(A,B)=P(A)P(B)$$

Conditional Independence

$$P(A,B|C)=P(A|C)P(B|C)$$

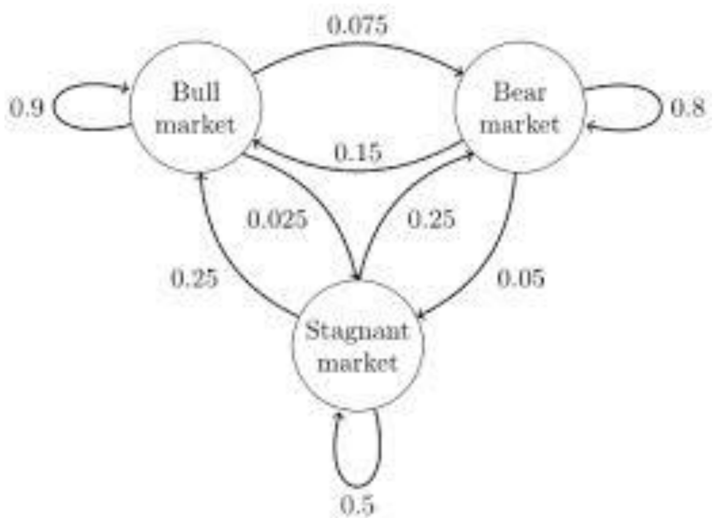
$$P(A|B,C)=P(A|C)$$

Chain Rule

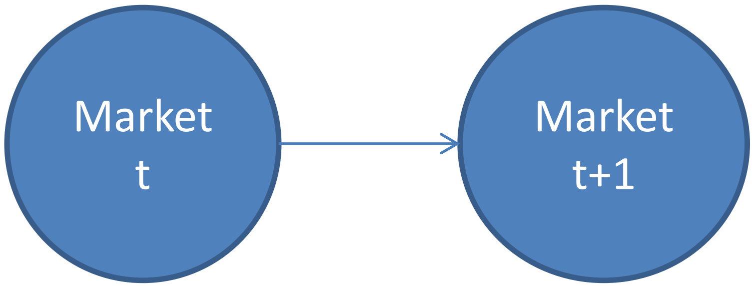
$$P(A,B,C)=P(C|A,B)P(B|A)P(A)$$

Markov Chains

- Initial state, or distribution over possible initial states.
- Transition probabilities
  - Markov Condition: State at time t+1 depends only on state at time t. (Leads to higher order MCs)
  - Ie Current state conditionally independent of all prior states except preceeding.



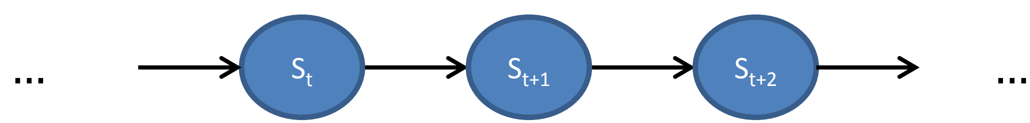
	Bull	Stagnant	Bear
Bull	.9	.025	.075
Stagnant	.25	.5	.25
Bear	.15	.05	.8



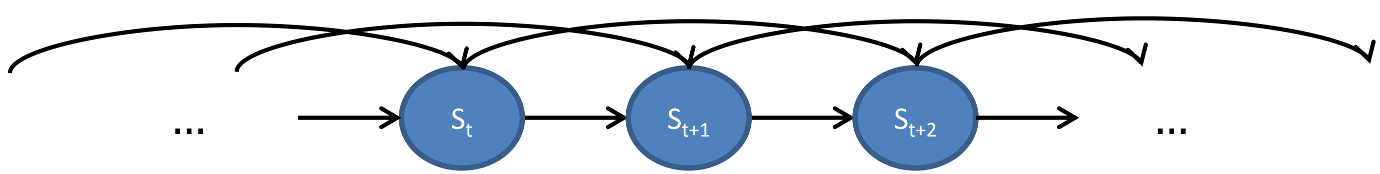
Markov Chains

- Using nodes to represent variables & conditional distributions
- Conditioned upon variables indicated by edges.

1st Order Markov Chain



2nd Order Markov Chain



Markov Chain

Transition probabilities for transition matrix T.

Simple state prediction (1st Order):

$$\boldsymbol{X}_{t+n} = {\boldsymbol{X}_t}^T \boldsymbol{T}^n$$

The eigenvector to the eigen value 1 gives the steady equilibrium distribution. (Ie ’long run’ distribution of the MC).

Sampling From a Markov Chain

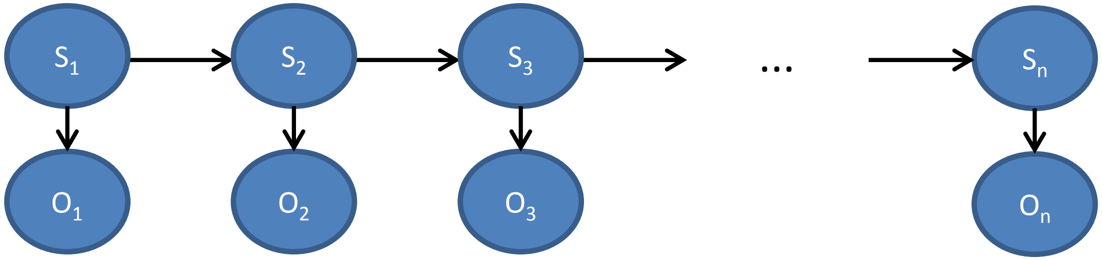
- Provides a sequence of sampled state vales corresponding to time 0 to t: $S_o, S_1, S_2, \ldots, S_t$
- Sample from the initial state distribution to find the sample value S_o
- Construct the distribution P($S_i \,|\, S_{i-1}$) using S_{i-1} and the transition matrix. Then sample from this distribution to S_i.
- Note: Samples are not independent.

Markov Chain Monte Carlo

- Generate a (1st order) MC that (in its equilibrium state) represents the target distribution.
- Proceed to generate samples from it by evolving the MC.
- As the number of samples approaches infinity, the sampled distribution approaches the actual equilibrium distribution.
  - Burn period
  - *n*th sample
- Blackboard example...

Hidden Markov Models

- State of system is hidden from us.
- Some observation related to the state is available to us.
  - Require sensor/emission probabilities, E.
  - Assume observations depend only on current state. (Conditionally indepent of all other states and observations.)



Hidden Markov Models

- Prediction: Just as in Markov Chains...

$$P(S_{t+n}|S_t\ ) = P(S_t)\boldsymbol{T}^n$$

Hidden Markov Models

Note * notation:

$$P^*(S_t|S_{t-1}) = \sum_{i=1}^{m} P(S_t|S_{t-1} = i)\,P(S_{t-1} = i)$$

Hidden Markov Models

We will make use of Bayes Rule:

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

When Y is observed, this becomes:

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

Hidden Markov Models

State Estimation, t>0:

$$P(S_t|O_{1:t}, S_0) = P(S_t|S_{t-1}, O_t)P(S_{t-1}|O_{1:t-1}, S_0)$$

Note the recursion:

$$\boldsymbol{P}(\boldsymbol{S_t}|\boldsymbol{O_{1:t}}, \boldsymbol{S_0}) = P(S_t|S_{t-1}, O_t)\boldsymbol{P}(\boldsymbol{S_{t-1}}|\boldsymbol{O_{1:t-1}}, \boldsymbol{S_0})$$

So we can proceed iteratively through, basising our estimation of S_t only on our estimation of S_{t-1} and observation O_t.

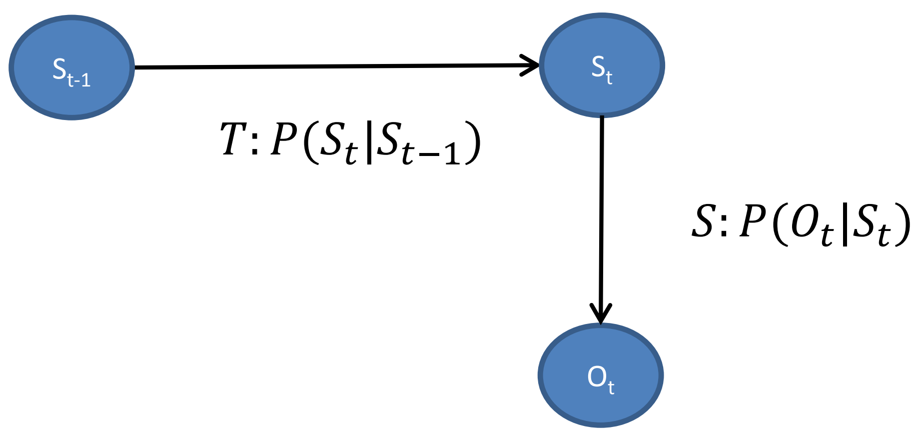
Hidden Markov Models

- State Estimation

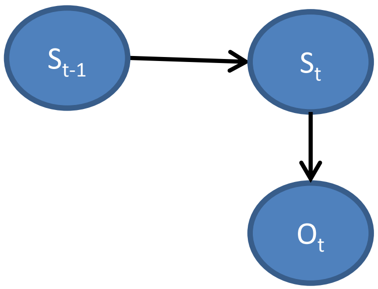
$$P(S_t|S_{t-1}, O_t) = \frac{P(O_t|S_t)P^*(S_t|S_{t-1})}{P(O_t)}$$

$$\propto P(O_t|S_t)P^*(S_t|S_{t-1})$$

Remember: The previous state estimation has <u>all</u> relevant information from the past!



Hidden Markov Models



S_{t-1}	S_t=T	S_t=F
T	.9	.1
F	.3	.7

S_t	O_t=T	O_t=F
T	.3	.7
F	.1	.9

$$P(S_t|S_{t-1}, O_t) \quad \propto P(O_t|S_t)P^*(S_t|S_{t-1})$$

- *Let our belief regarding* S_0 *be that it is 80% likely* S_0=T.
- *Let us observe* O_1=F.

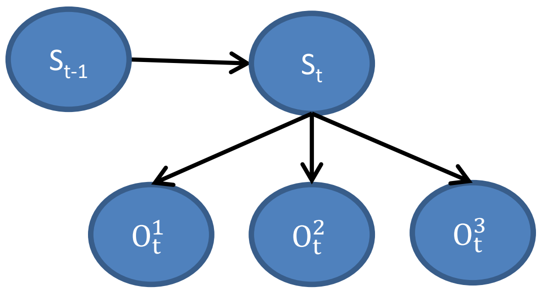
$P(S_1|S_0) =< (.8)(.9) + (.2)(.3), (.8)(.1) + (.2)(.7) > = < .78, .22 >$

$P(O_1 = F|S_1) = < .7 , .9 >$

$P(S_1|S_0, O_1 = F) \propto <(.78)(.7),(.22)(.9)>=<.546,.198>$

$P(S_1|S_0, O_1 = F) = < \frac{.546}{.546+.198}, \frac{.198}{.546+.198} > \approx <.734, .266 >$

Hidden Markov Models



S_{t-1}	S_t=T	S_t=F
T	.6	.4
F	.5	.5

S_t	O_t^1
T	$\mathcal{N}(3.5,10)$
F	$\mathcal{N}(5,5)$

S_t	O_t^2
T	$\mathcal{N}(45,100)$
F	$\mathcal{N}(55,225)$

S_t	O_t^3
T	$\mathcal{N}(0,.1)$
F	$\mathcal{N}(0,.5)$

$$P(S_t|S_{t-1},O_t) \propto \rho(O_t^1|S_t)\rho(O_t^2|S_t)\rho(O_t^3|S_t)P^*(S_t|S_{t-1})$$

- *Let our belief regarding* S_0 *be that it is 50% likely* S_0=T.
- *Let us observe* O_1^1=6.103, O_1^2=54.7 and O_1^3=.154

$$P(S_1|S_0) = <(.5)(.6)+(.5)(.5),(.5)(.4)+(.5)(.5)> = <.55,.45>$$
$$P(O_1^1 = 6.103|S_1) \approx <.089,.158>$$
$$P(O_1^2 = 54.7|S_1) \approx <.025,.027>$$
$$P(O_1^3 = .154|S_1) \approx <1.120,.551>$$
$$P(S_1|S_0,O_1^1 = 6.103, O_1^2 = 54.7, O_1^3 = .154) \propto <(.55)(.089)(.025)(1.120),(.45)(.158)(.027)(.551)> \approx <.00137,.00106>$$
$$P(S_1|S_0,O_1^1 = 6.103, O_1^2 = 54.7, O_1^3 = .154) \approx <\frac{.00137}{.00137+.00106},\frac{.00106}{.00137+.00106}> \approx <.564,.436>$$

Hidden Markov Models: Lab B

- State Estimation
  - Given an initial state, transition and sensor probabilities, we can iteratively calculate the distribution at each subsequent state.
  - We can do this online.

Note that for Lab B

- Vector of 3 observations (as in last example)
- Sparse transition matrix (many impossible transitions). Assume random walk with possibility of staying still.
- Presumably uniform initial state.
- NOT real time.

Hidden Markov Models

Smoothing: The Forward-Backward Algorithm

$$P(S_{s\leq t}|O_{0:t}, S_0)$$
$$= P(S_{s\leq t}|O_{0:s}, S_0)P(S_{s\leq t}|O_{s+1:t})$$

The Forward Algorithm:

- We have seen how, given an initial state, transition and sensor probabilities, we can iteratively calculate $P(S_s|O_{0:s}, S_0)$ for $1 \leq s \leq t$.

The Backward Algorithm:

- Starting at t, we can iteratively calculate:

$$P(S_s|O_{s+1:t})$$

Hidden Markov Models

$$
\begin{aligned}
\boldsymbol{P(S_s|O_{s+1:t})} &= P(S_s|O_{s+2:t}, O_{s+1}) \\
&= P(S_s|S_{s+1})P(S_{s+1}|O_{s+1:t}) \\
&= P(S_s|S_{s+1})P(S_{s+1}|O_{s+2:t}, O_{s+1}) \\
&= P(S_s|S_{s+1})P(O_{s+1}|S_{s+1})P(S_{s+1}|O_{s+2:t}) \\
&\propto \ P(S_s)P(S_{s+1}|S_s)P(O_{s+1}|S_{s+1})\boldsymbol{P(S_{s+1}|O_{s+2:t})}
\end{aligned}
$$

– Note the recursion.

– We have a base case since:

$$P(O_{t+1:t}|S_t) = P(\emptyset|S_t) = 1$$

– We actually iterate backwards from t.

Hidden Markov Models

Forward backward algorithm for all states.

Forward chain:

$$f_0 = S_0$$
$$f_i = f_{i-1} T O_i$$

Backward chain:

$$b_t = 1$$
$$b_i = O_{i+1} T b_{i+1}$$

Combination:

$$P(S_i) \propto f_i b_i$$

Blackboard example...

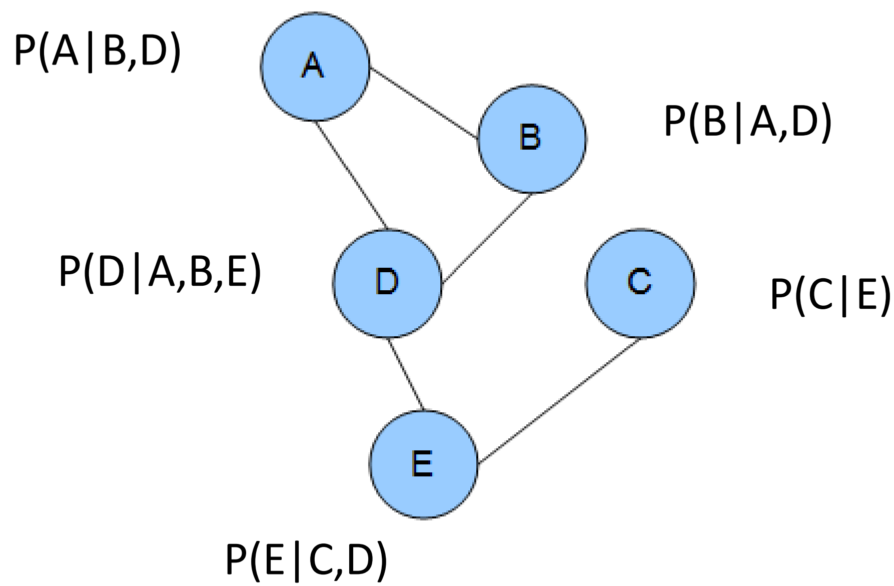
Hidden Markov Models

- Most Probable Path: Viterbi Algorithm
  - Calculate the path probabilities as in the Dynamic Programming for Path Finding.
    - Difference: Multiplicative instead of additive accumulation function.
    - Does not find probability of most probable path unless normalization (over all paths) occurs at each step.
  - Using log probabilities useful.
  - Blackboard example...

Markov Random Fields

An undirected network of models, each specifying the conditional distribution for a variable given its neighbors.

Note graph convention: Nodes represent variables. Conditional distributions associated with each node. Edges into nodes indicate which variables are conditioned upon.



Markov Random Fields

- Inference:

Given the states we know, to find out the states we do not know, we sample...

- Gibbs Sampler
  - Divide domain into known and unknown variables.
  - Assign unknown variables a random value.
  - We iterate through unknown variables, calculating a new value given the values assigned to their neighbors.
  - After each iteration, we record a sample.
  - We estimate distributions of interest from these samples.

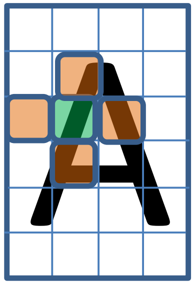
Example on blackboard.

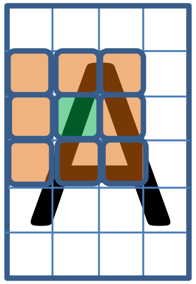
Markov Random Fields

- Example: Letter recognition
  1. Gather lots of examples of writen letters of the relevant alphabet (and scale them to a normal size).
  2. Divide the letters into segments.
  3. Pick a characteristic of the image found in these segments: Eg number of curves, or number of vertices (meeting points of curves).

Markov Random Fields

- We notice that the probability distribution for a given characteristic, for a given letter, is dependent on its neighboring segments.
- The variables in the model are discrete. The distributions are all conditional multinomials.





Markov Random Fields

Training:

Train a model <u>for each letter</u> using Bayesian methods:

- Use Dirichlet/count statistics to estimate the true distributions from the training data for each letter.
- This will give you models for expected distribution of particular letters.

Example on blackboard.

Markov Random Fields

Classifying:

Given a new letter-image, we:

- Divide it into segments and classify each segment by the chosen characteristic (eg number of curves).
- Calculate the probability of this set of characteristic values for the segments for each of our letters.
- Normalizing these values give us the probability of the letter-image being a given letter.

Example on blackboard.

Categorical Distributions

Categorical distributions use *n* parameters to specify the probability distribution of a *n*-valued random variable. Each parameter, *i*, gives the probability of the variable taking the *i*th value. The degrees of freedom of such a distribution is *n*-1, since we have the constraint:

$$\sum_{j=1}^{n} P(X = x_j)$$

Result:	Win	Draw	Loss
Prob.	.6	.3	.1

Here is a three valued categorical distribution representing the result of a match:

Conditional Categorical Distributions

Conditional categorical distributions P(Y|**X**) give a categorical distribution for each possible value of the discrete variables being conditioned upon.

Here is a conditional categorical distribution representing the distribution over the result of a match given the values taken by the location and weather variables:

Location	Weather	Win	Draw	Loss
Home	**Raining**	.2	.7	.1
Home	**Normal**	.8	.15	.05
Home	**Hot**	.6	.2	.1
Away	**Raining**	.1	.8	.1
Away	**Normal**	.5	.4	.1
Away	**Hot**	.2	.6	.2

This is a classifier that gives a distribution for an output variable Y given input variables **X**.

Maximum Likelihood & Count Parameters

Take discrete variable $X: \{x_1, x_2, \dots, x_n\}$, distributed $cat(p_1, p_2, \dots, p_n)$.
Let us track the number of times that we have seen X take particular values with the count parameters:

$$\{c_1, c_2, \dots, c_n\}$$

The *maximum likelihood* value of the parameters $p_1, p_2, \dots, p_n$ is the value that makes the observations most probable. It is:

$$p_i = \frac{c_i}{\sum_{j=1}^{n} c_j}$$

- The ratio of the count parameters gives us our ML estimation for the distribution parameters.
- As the count parameters increase, new observations will alter the ML estimation of the distribution parameters less and less.

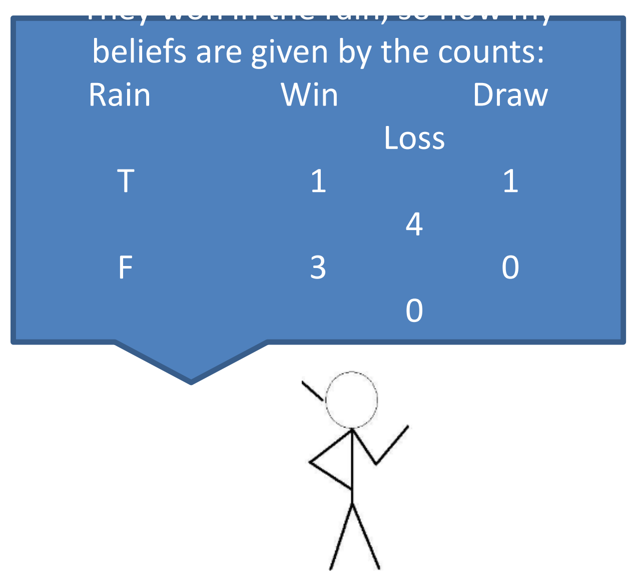
Count Parameters & Adaption

Using counts makes it easy to adapt our parameter estimates: We simply add to the counts as observations occur and adjust accordingly.

We can adapt to soft evidence too: If we hear that another coin toss has occurred from someone who cannot remember the result for sure, but is 75% sure that it was heads, we would have the counts <2.75,1.25>.

Count Parameters & Conditional Distributions

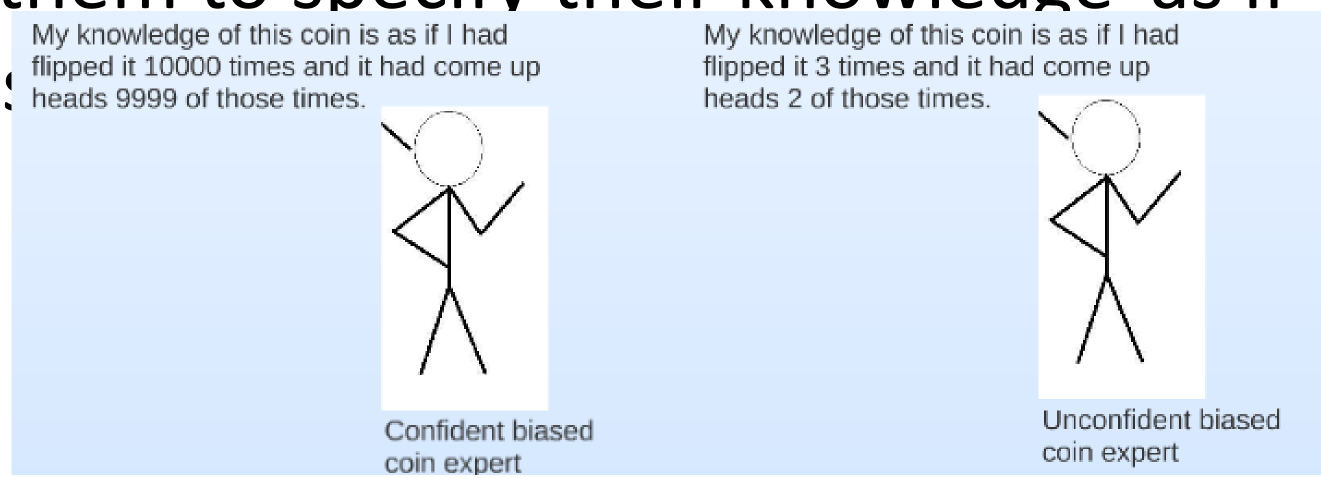
For conditional distributions, we keep counts under each set of possible conditions of the variables being conditioned upon.



Pseudo-Observations and Expert Knowledge

Easy to encode an expert’s knowledge about probabilities <u>and</u> their confidence in their estimation:

Get them to specify their knowledge ’as if’ they had s



Count Parameters and Dirichlet Distributions

The count parameters can be interpreted as the parameters of a Dirichlet distribution over our belief regarding the correct value of the parameters in the categorical distribution.

$$Dir(c_1, c_2 \dots c_n) = \frac{\Gamma(\sum_{i=1}^{n} c_i)}{\prod_{i=i}^{n} \Gamma(c_i)} \prod_{i=1}^{n} x_i^{c_i - 1},$$

$$With\ support\colon\ x_1, \dots, x_{n-1}\ where\ x_i \in [0,1]\ and\ \sum_{i=1}^{n-1} x_i < 1$$

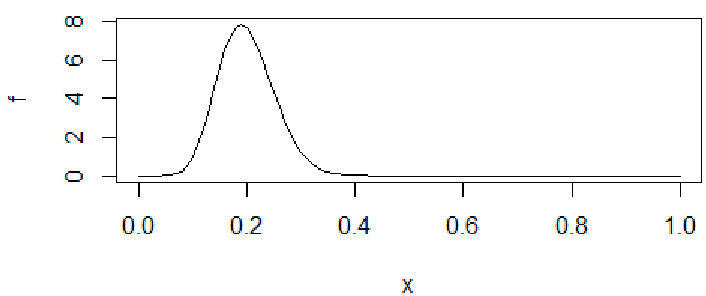
Note: x_n is implicit from the constraint.

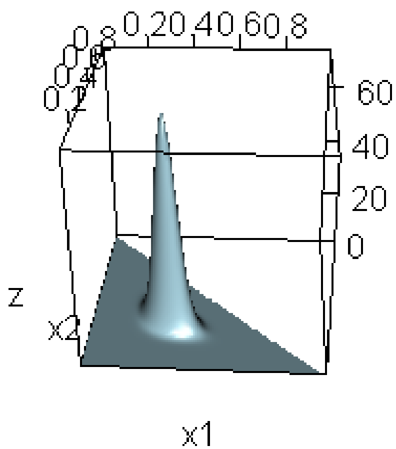
Just understand the relationship between the shape and the parameters! See examples: abn::dir.plot(…).

Count Parameters and Dirichlet Distributions

We can obtain confidence estimates for the parameters of our categorical distribution given our observations and prior beliefs (pseduo-obsevations) count.

Working with Dirichlets is beyond the scope of this course.





Ignorance & Conservativism

A common choise is to model ignorance ’as if’ we had seen all values occur once. This is because otherwise we would jump to certainty after a single observation! (Why?)

Dirichlet distributions accord with this convention: Dirichlet distributions of all ones are uniform over possible parameters.