Inference in Bayes Nets via Sampling

- If the Bayes net is too complex to do Variable Elimination on, or we want to answer queries that VE can't easily model.

- The Bayes net defines a joint distribution P(X1, X2, …, Xn) over its features.

- Idea is to draw atomic events from this distribution in such a way that the probability we obtain the atomic event
  <X1=d1, X2=d2, …, Xn=dn> is exactly
  P(X1=d1, X2=d2, …, Xn=dn)

Inference in Bayes Nets via Sampling

- Then if we collect a set of m samples
  <X1=d11, X2=d12, …, Xn=d1n>
  <X1=d21, X2=d32, …, Xn=d2n>
  <X1=d31, X2=d32, …, Xn=d3n>
  <X1=d41, X2=d42, …, Xn=d4n>
  <X1=d51, X2=d52, …, Xn=d5n>
  <X1=d61, X2=d62, …, Xn=d6n>
  …
  <X1=dm1, X2=dm2, …, Xn=dmn>

Inference in Bayes Nets via Sampling

- We can estimate many different probabilities by looking that the frequency among the m samples.

- e.g. to estimate P(X1=a) we count how many samples have X1=a and divide by m
- to estimate P(X1=a $\vee$ X2 = b) we count how many samples have X1=a or X2=b and divide by m.

- Notice that the 2nd query can't easily be answered with VE, even if we had the computational resources to do so.

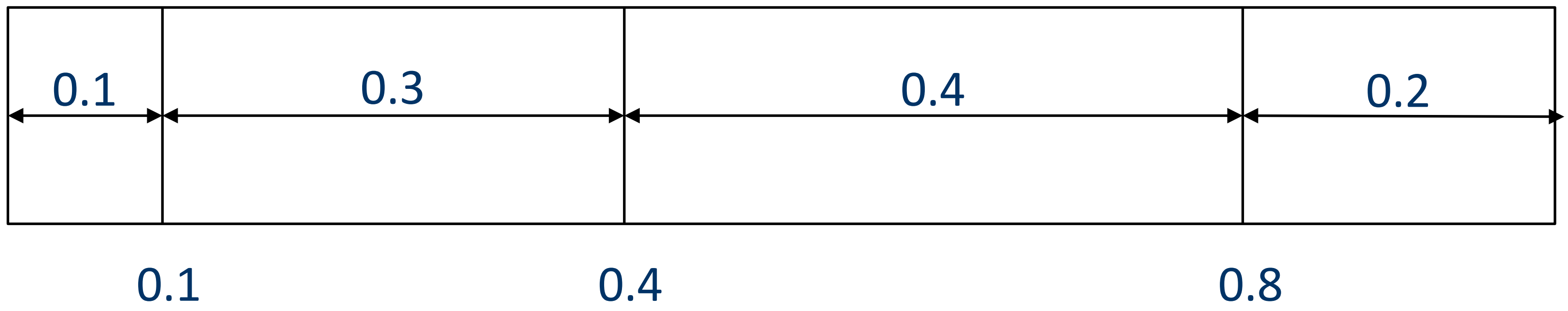
Sampling from a distribution

- In python random.uniform(0.0, 1.0) generates a random number in the range [0.0, 1.0] with each number approximately equally likely to be generated.
- Sampling in Bayes Nets requires sampling from a distribution over a variable. This is accomplished by mapping the probabilities to equal sized ranges of the [0,1] interval. Then selecting the variable’s value based on which range the uniform random number lies in.

Sampling from a distribution

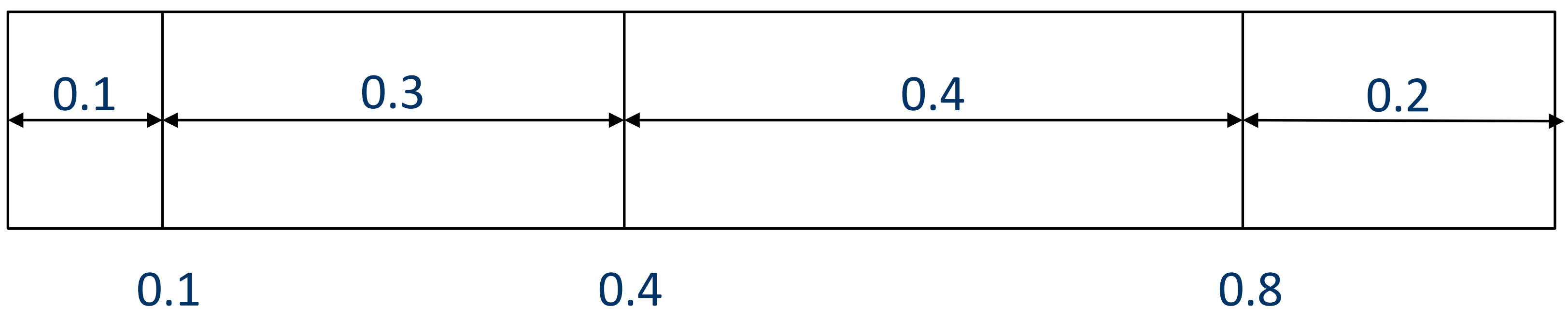
- E.g., say we have P(A|B=t, C=t) = [0.1, 0.3, 0.4, 0.2]
  That is the probability that A (given B=t, C=t) gets is first value is 0.1, the probability it gets it second value is 0.3, etc.

- We break the unit interval into 4 segments, the first being 0.1 long, the second begin 0.3 long, etc.



Sampling from a distribution

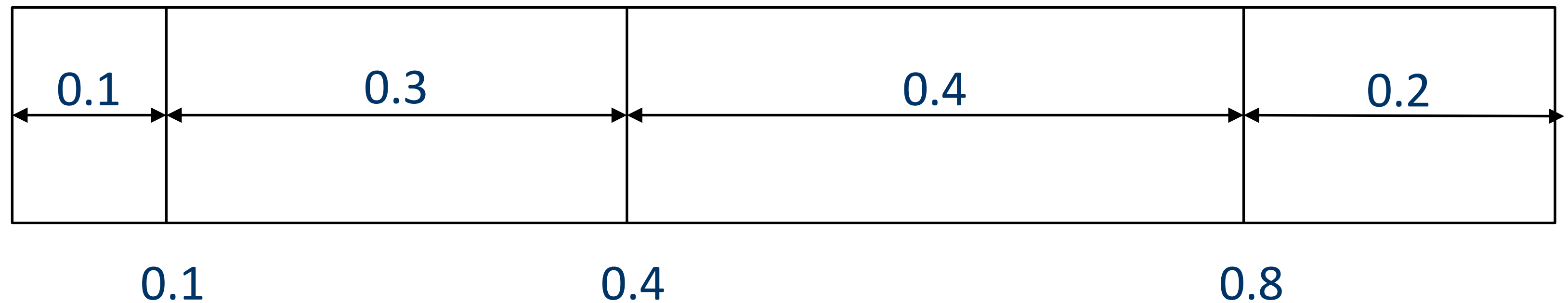
- We generate a uniformly distributed random number in the range 0.0—1.0, and if that number falls into the first range we set A to its first value, if it falls into the second range we set A to its second value, etc.
  - The boundary value (0.1, 0.4, 0.7 in our example) is considered to lie in the right range, so if the random number is 0.1 we set A to its second value, if it is 0.4 we set A to its third value etc.



Sampling from a distribution

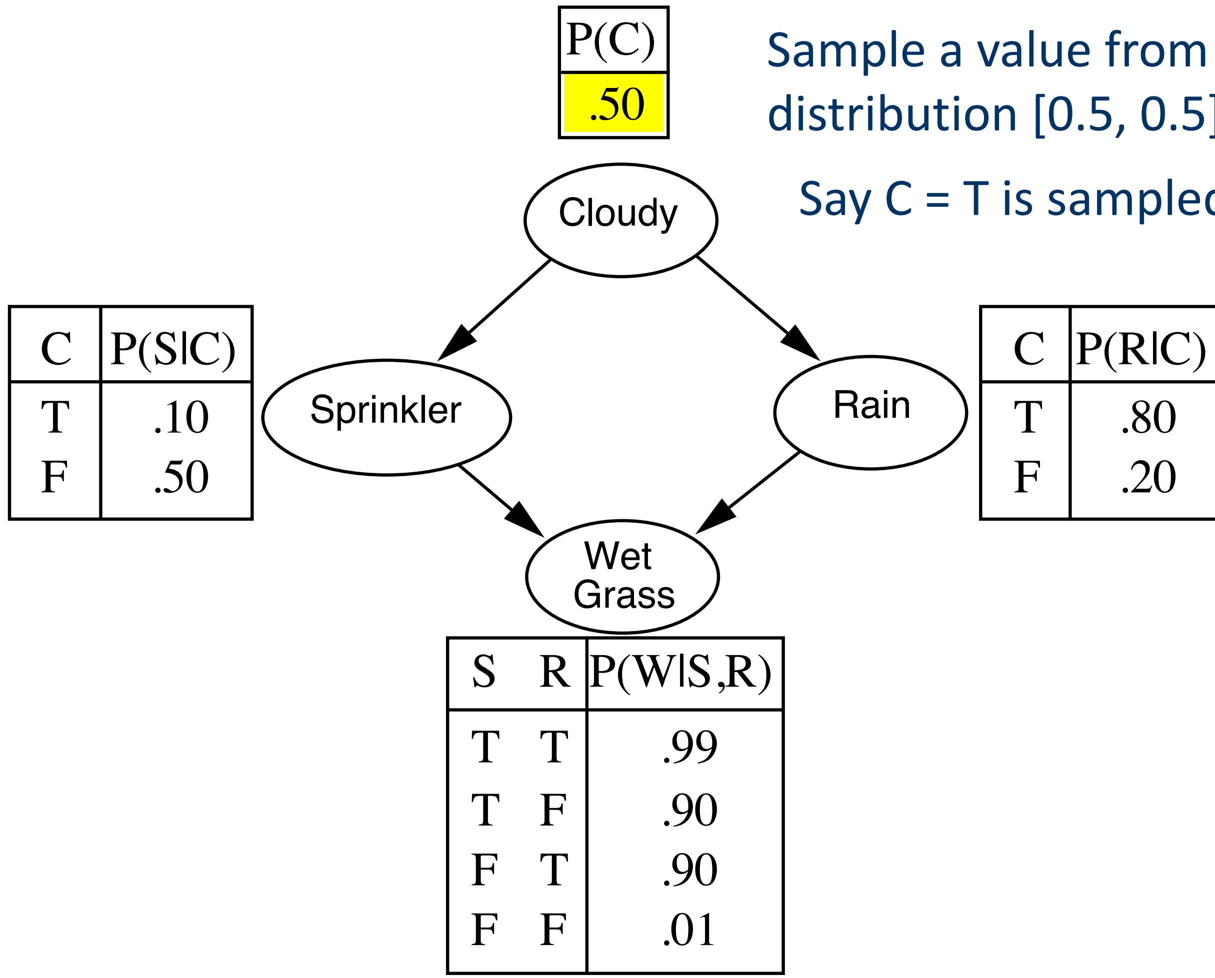
- E.g., the following uniform random numbers 0.1, 0.827, 0.326, 0.865, 0.775 generates the following random settings of the variable A

  2nd value of A, 4th value of A, 2nd value of A, 4th value of A, 3rd value of A.



Prior Sampling---Generating Samples from a Bayes net

- Start at the roots of the Bayes nets (i.e., those variables that have no parents)
- Randomly select a value for each of these variables from the distribution given by the variables CPT.
- Work your way down the net, when you have a value for all of a variable X's parents randomly select a value for X using the distribution P(X|par(X)). This distribution is in X's CPT and all of the variables in par(X) have already been set by the previous steps.
- When all variables have a value---you get one sample.

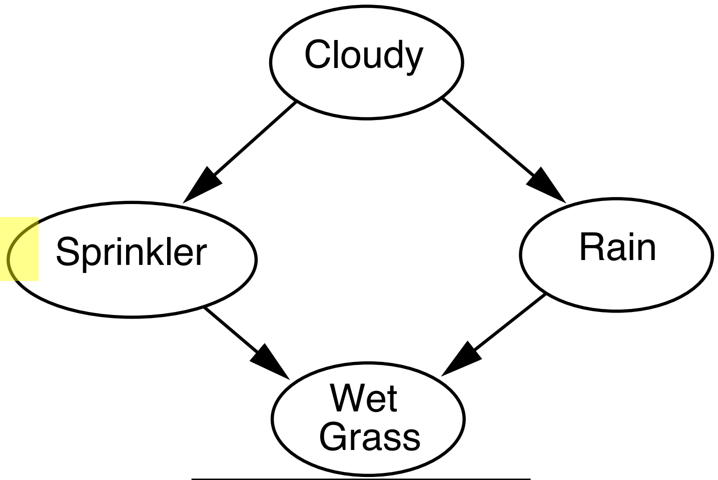


Sample a value from C from distribution [0.5, 0.5]

Say C = T is sampled.

P(C)
.50

C=T



C	P(S|C)
T	.10
F	.50

C	P(R|C)
T	.80
F	.20

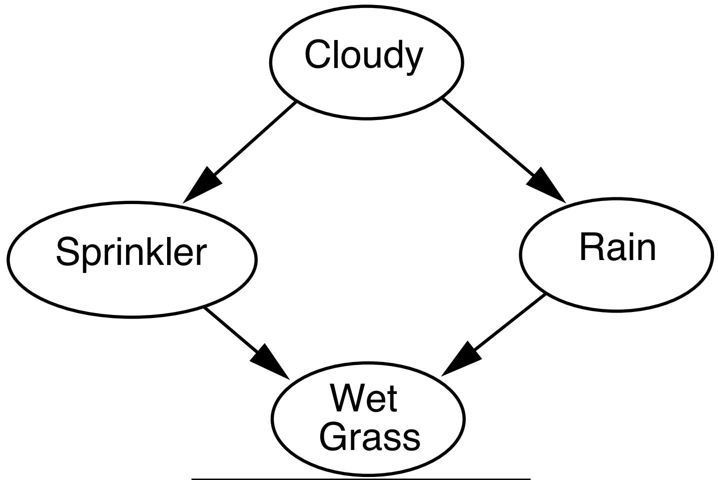
S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

Now sample a value for S from the distribution P(S|C=T) = [0.1, 0.9]

Say we sample S = F

P(C)
.50

C=T



S=F

C	P(S|C)
T	.10
F	.50

C	P(R|C)
T	.80
F	.20

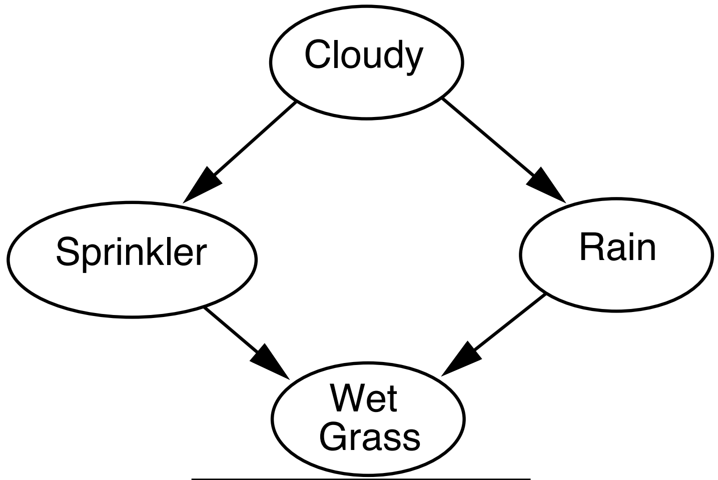
S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

Now sample a value for R from the distribution P(R|C=T) = [0.8, 0.2]

Say we sample R = T

P(C)
.50

C=T



S=F

C	P(S|C)
T	.10
F	.50

C	P(R|C)
T	.80
F	.20

R=T

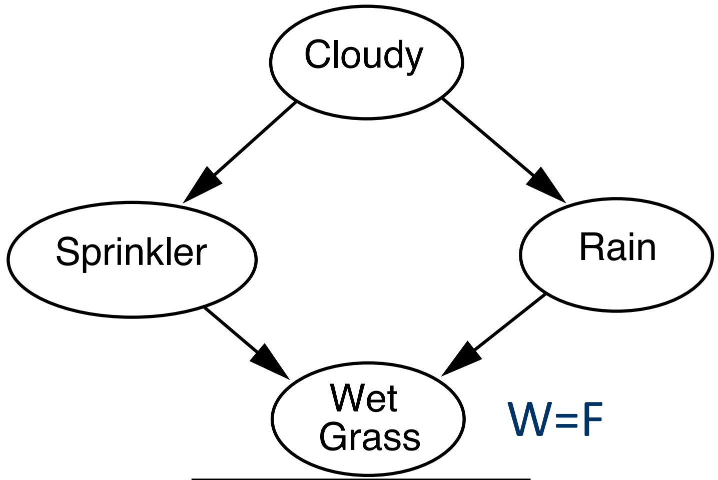
S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

Now sample a value for W from the distribution P(W|S=F, R=T) = [0.9, 0.1]

Say we sample W = T

P(C)
.50

C=T



S=F

C	P(S|C)
T	.10
F	.50

R=T

C	P(R|C)
T	.80
F	.20

W=F

S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

Our random sample is
C=T, S=F, R=T, W=F

Rejection Sampling

- Suppose we want to estimate P(Xk = dk | X1=d1, X2=d2)
- For this query all of the samples where X1 is not equal to d1 or X2 = d2 are useless
- P(Xk = dk | X1=d1, X2=d2) =
  P(Xk = dk , X1=d1, X2=d2)/P(X1=d1, X2=d2)

- So we have to count how many samples have Xk = dk and X1=d1 and X2=d2 and then divide by the number of samples that have X1=d1 and X2=d2

- This is called rejection sampling---the samples were X1 is not equal to d1 or X2 is not equal to d2 are useless—they are rejected.

Rejection Sampling

- Problem is if X1=d1, X2=d2 has low probability we will reject almost all samples!

Likelihood Weighting

- If we want to estimate a conditional probability like P(Xk=dk | X1=d1, X2=d2) we force all samples to satisfy the condition.

- Problem—in doing so we bias the samples, so that we are no longer sampling from the Bayes net distribution.

- "Solution:" we reweigh the samples so that we undo this introduced bias.

Likelihood Weighting

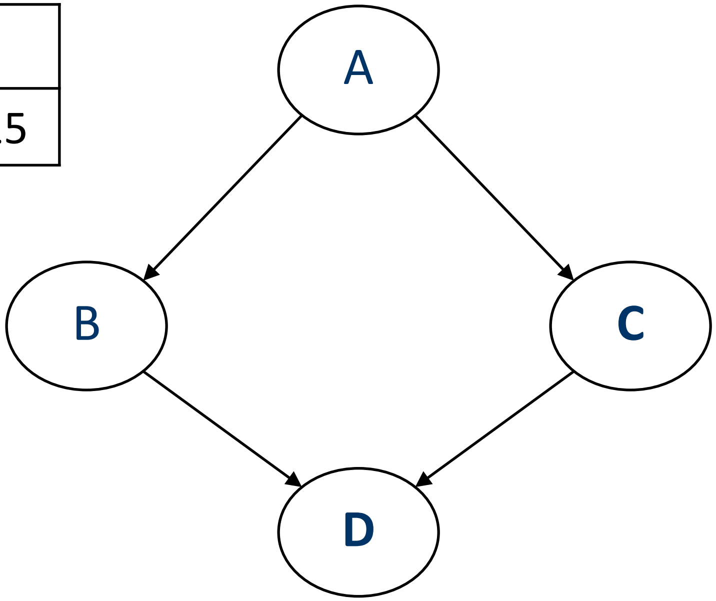
- Say the conditioning is X1=d1, X2=d2
- Set wt of the sample = 1
- start at the roots as before
  - **if the variable is fixed by the condition, we set it to the required value and multiply wt by the probability it takes on that value.**
  - Else we set it at random according to its probability
- Moving down, we set the value of a variable X after we have set the values of all of its parents.
  - Again if X is fixed by the condition we set it to the required value and multiply wt by the probability it takes on that value
  - Else we set it at random according to its probability.
- At the end we have a single sample and a weight.

Likelihood Weighting

- To estimate the probabilities we want we now take the sum of the weights of the good samples divided by the total weight of the samples.
- Pr(Xk=dk| X1=d1, X2=d2) =
  sum of weights of samples where Xk=dk/
  total sum of weights of samples

Example

P(A)	lo	med	hi
	0.25	0.5	0.25



P(B|A)	t	f
lo	0.1	0.9
med	0.4	0.6
hi	0.9	0.1

P(C|A)	t	f
lo	0.1	0.9
med	0.4	0.6
hi	0.9	0.1

P(D|B,C)	lo	med	hi
t,t	0	0.1	0.9
t,f	0.1	0.4	0.5
f,t	0.5	0.4	0.1
f,f	0.9	0.1	0

Example

- Applying the prior sampling technique and using random.uniform (see the sample.py python code linked on the website), the call to get_prior_samples(10) generates the following 10 samples:

```
1.  A = med, B = f, C = f, D = lo
2.  A = lo, B = f, C = f, D = lo
3.  A = med, B = t, C = f, D = hi
4.  A = lo, B = f, C = f, D = lo
5.  A = med, B = f, C = f, D = lo
6.  A = med, B = f, C = f, D = med
7.  A = lo, B = t, C = f, D = hi
8.  A = hi, B = t, C = t, D = med
9.  A = med, B = t, C = t, D = hi
10. A = med, B = f, C = t, D = hi
```

Example

- If we want to compute P(C=t | D = med) we must reject samples 1, 2, 3, 4, 5, 7, 9, and 10 as none of these agree with the condition D=med

```
1.  A = med, B = f, C = f, D = lo
2.  A = lo, B = f, C = f, D = lo
3.  A = med, B = t, C = f, D = hi
4.  A = lo, B = f, C = f, D = lo
5.  A = med, B = f, C = f, D = lo
6.  A = med, B = f, C = f, D = med
7.  A = lo, B = t, C = f, D = hi
8.  A = hi, B = t, C = t, D = med
9.  A = med, B = t, C = t, D = hi
10. A = med, B = f, C = t, D = hi
```

Example

- This leaves us with only 3 samples

```
6.  A = med, B = f, C = f, D = med
8.  A = hi, B = t, C = t, D = med
```

- One of these has C=t so our estimate of P(C=t|D=med) = ½
- We do not have much confidence in this estimate since the number of non-rejected samples is so small (only 2).

- Using sample.py to generate 10000 samples we get 1980 of them left are rejection (about 20%), and we estimate P(C=t|D=med) = .47 and this estimate is pretty good.

Example

- Using likelihood weighting we can generate samples as follows (again sample.py gives the code)

Say we have as evidence C=t

1. set wt = 1.0
2. Sample a value for A. A is not set in evidence so we sample a value for it from the distribution P(A). Perhaps we get A=hi
3. Sample a value for B. B is not set in the evidence so we sample a value for it from the distribution P(B|A=hi). Perhaps we get B=t
4. Sample a value for C. **C is in evidence** so we must set C=t and we set wt = wt*P(C=t|A=hi)=1.0*0.9—probability of the evidence given the values of the parents already set in the sample.
5. Finally we sample a value for D. D is not set in evidence so we sample a value for it from the distribution P(D|B=t, C=t). Perhaps we get D=hi

Example

- So our final sample is
  A=hi, B=t, C=t, D=hi with wt=0.9

- If we generate 10 samples (using the function get_likelihood_samples(10) from sample.py) we obtain:

```
1. wt = 0.4, A = med, B = f, C = t, D = lo
2. wt = 0.4, A = med, B = f, C = t, D = med
3. wt = 0.4, A = med, B = f, C = t, D = med
4. wt = 0.4, A = med, B = t, C = t, D = hi
5. wt = 0.1, A = lo, B = f, C = t, D = lo
6. wt = 0.4, A = med, B = t, C = t, D = med
7. wt = 0.4, A = med, B = f, C = t, D = hi
8. wt = 0.1, A = lo, B = f, C = t, D = lo
9. wt = 0.4, A = med, B = f, C = t, D = med
10. wt = 0.4, A = med, B = t, C = t, D = med
```

Example

- From these samples we estimate P(D=med|C=t) to be the weights of samples 2, 3, 6, 9 and 10 divided by the total weight of samples

```
0.4 + 0.4 + 0.4 + 0.4 + 0.4/0.4 + 0.4 + 0.4 + 0.4 + 0.1
                            +0.4 + 0.4 + 0.1 + 0.4 + 0.4
= 0.35
```

```
1. wt = 0.4, A = med, B = f, C = t, D = lo
2. wt = 0.4, A = med, B = f, C = t, D = med
3. wt = 0.4, A = med, B = f, C = t, D = med
4. wt = 0.4, A = med, B = t, C = t, D = hi
5. wt = 0.1, A = lo, B = f, C = t, D = lo
6. wt = 0.4, A = med, B = t, C = t, D = med
7. wt = 0.4, A = med, B = f, C = t, D = hi
8. wt = 0.1, A = lo, B = f, C = t, D = lo
9. wt = 0.4, A = med, B = f, C = t, D = med
10. wt = 0.4, A = med, B = t, C = t, D = med
```

Example

- A larger sample of 10000 samples gives the estimate P(D=med|C=t) = 0.205 which is a pretty good estimate.