

# Advanced Machine Learning Subsidiary Notes

## Lecture 19: Generative Models

Adam Prügel-Bennett

January 28, 2021

## 1 Keywords

- Conditional Independence, Graphical models, LDA

## 2 Main Points

### 2.1 Overview

- We have so far considered building rather simple probabilistic models
- But what if we want to do inference on a more complicated problem, e.g.
  - We might want to write a fault diagnosis system for a car
  - Or we want to create an AI doctor
  - The model of the spread of a virus
- Here we have a vast number of random variables with complicated relationships between them
- To help design our system we can build a *graphical model* showing the causal relationships between random variables

### 2.2 Conditional Independence

- **Independence**

- Two random variable  $X$  and  $Y$  are independent if

$$\mathbb{P}[X, Y] = \mathbb{P}[X] \mathbb{P}[Y]$$

- Independence can significantly speed up calculations, e.g.

$$\mathbb{E}[X^2 Y] = \sum_{X,Y} X^2 Y \mathbb{P}[X, Y] = \left( \sum_X X^2 \mathbb{P}[X] \right) \left( \sum_Y Y \mathbb{P}[Y] \right)$$

- \* If  $X$  and  $Y$  takes  $n$  and  $m$  values then without independence the double sum  $\sum_{X,Y}$  would be over  $n \times m$  possible values
- \* With independence we can compute these sums independently so it just takes  $m+n$  additions
- When we have large systems with many independent variables then the time saving is often the difference between calculations being feasible or infeasible
- Unfortunately in most complex systems there is likely to be some dependence between random variables

- **Conditional Independence**

- A weaker notion than full independence is *conditional independence*
- We say that  $X$  and  $Y$  are conditionally independent given  $Z$  if

$$\mathbb{P}[X, Y|Z] = \mathbb{P}[X|Z] \mathbb{P}[Y|Z]$$

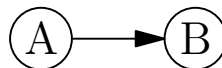
- Again this can lead to significant speed up in evaluating expectations, e.g.

$$\begin{aligned} \mathbb{E}[X^2 Y Z^3] &= \sum_{X,Y,Z} \mathbb{P}[X, Y, Z] X^2 Y Z^3 = \sum_{X,Y,Z} \mathbb{P}[X, Y|Z] \mathbb{P}[Z] X^2 Y Z^3 \\ &= \sum_Z Z^3 \left( \sum_X X^2 \mathbb{P}[X|Z] \right) \left( \sum_Y Y \mathbb{P}[Y|Z] \right) \end{aligned}$$

- \* If  $X$ ,  $Y$  and  $Z$  have  $l$ ,  $m$  and  $n$  values respectively, then, ignoring conditional independence, this expectation would require  $l \times m \times n$  additions; using conditional independence it only requires  $n \times (l + m)$  additions
- Although conditional dependence doesn't imply causality, if random variables  $X$  and  $Y$  are not directly causally related they will be conditionally independent
- This is important prior information we can build into our model

## 2.3 Graphical Models

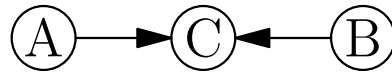
- In graphical models we represent each random variable as a node in a graph
- There are two main classes of graphical models
  - Bayesian Belief Networks
    - \* use directed graphs
    - \* we will spend most of our time discussing these
  - Markov Fields
    - \* uses undirected graphs
    - \* these are used in graphics a lot
    - \* won't really discuss these
- Each causal connection we represent as a directed edge in a Bayesian Belief Network
  - if  $A$  directly influences  $B$  we represent this as



- **Cakes**

- We consider the following example
- Abi and Ben both bake cakes and like to bring them into the coffee room
- They do this randomly without consulting with each other
- Abi will bring in cakes 20% of the time:  $\mathbb{P}[A = 1] = 0.2$
- Ben will bring in cakes 10% of the time:  $\mathbb{P}[B = 1] = 0.1$
- 90% of the time if either Abi or Ben have put cakes in the coffee room there is some left when I enter  $\mathbb{P}[C = 1|A = 1, B = 0] = \mathbb{P}[C = 1|A = 0, B = 1] = 0.9$
- If they both make cake then there is always cake left  $\mathbb{P}[C = 1|A = 1, B = 1] = 1$
- If neither Abi or Ben has made cake there is still a 5% chance someone else has put cake in the coffee room  $\mathbb{P}[C = 1|A = 0, B = 0] = 0.05$

- We note that  $\mathbb{P}[C = 0|A, B] = 1 - \mathbb{P}[C = 1|A, B]$  as  $\sum_{C \in \{0,1\}} \mathbb{P}[C|A, B] = 1$
- We can draw the causal relationships as



- This allows us to break down the joint probability as

$$\begin{aligned} \mathbb{P}[A, B, C] &\stackrel{(1)}{=} \mathbb{P}[C, B|A] \mathbb{P}[A] \\ &\stackrel{(2)}{=} \mathbb{P}[C|A, B] \mathbb{P}[B|A] \mathbb{P}[A] \stackrel{(3)}{=} \mathbb{P}[C|A, B] \mathbb{P}[B] \mathbb{P}[A] \end{aligned}$$

- (1) Using the definition of conditional probability (this is always true)
- (2) Using the definition of conditional probability again (this is always true)
- (3) Using the fact that  $B$  and  $A$  are independent (there is no arrow between them in the graphical representation) so  $\mathbb{P}[B|A] = \mathbb{P}[B]$ 
  - \* From the graphical representation we can immediately write down a simple form for this joint distribution
- We can use this decomposition to help us compute various probabilities
- (To compute probabilities we use the fact that the expectation of an indicator function  $\llbracket \text{predicate} \rrbracket$  is equal to the probability  $\mathbb{P}[\text{predicate}] = \mathbb{E}[\llbracket \text{predicate} \rrbracket]$ 
  - \* The indicator function  $\llbracket \text{predicate} \rrbracket$  equals 1 if the predicate is true and 0 otherwise)
- Let's compute the probability there are cakes

$$\mathbb{P}[C = 1] = \sum_{A, B, C \in \{0,1\}} \llbracket C = 1 \rrbracket \mathbb{P}[A, B, C] = \sum_{A, B \in \{0,1\}} \mathbb{P}[C = 1|A, B] \mathbb{P}[A] \mathbb{P}[B] = 0.303$$

- \* See Section 3.1 for details of the calculation (this is an exercise that should really help)
- \* Here we exhaustively sum over all variables
- Let us consider what happens when we observe a random variable
  - \* In graphical models we often shade observed random variables

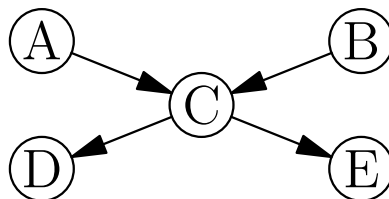


- \* Let's compute quantities conditioned on an observation of  $C$

$$\mathbb{P}[A, B|C] = \frac{\mathbb{P}[A, B, C]}{\mathbb{P}[C]}$$

- \* Thus  $\mathbb{P}[A, B|C = 1] = \mathbb{P}[A, B, C = 1] / \mathbb{P}[C = 1]$
- \* Using this we can compute
 
$$\mathbb{P}[A = 1, B = 1|C = 1] = 0.066, \quad \mathbb{P}[A = 1|C = 1] = 0.630, \quad \mathbb{P}[B = 1|C = 1] = 0.317$$
- \* We note that  $\mathbb{P}[A = 1, B = 1|C = 1] \neq \mathbb{P}[A = 1|C = 1] \mathbb{P}[B = 1|C = 1]$
- \* That is once we observe  $C$  then  $A$  and  $B$  are no longer independent
- We can extend our model further
  - \* We suppose that Dave likes cakes so if there is a cake in the coffee room there is a 80% chance that I will see him eating a cake:  $\mathbb{P}[D = 1|C = 1] = 0.8$
  - \* Even if there are no cakes in the coffee room there is a 10% chance that Dave has bought his own cake:  $\mathbb{P}[D = 1|C = 0] = 0.1$
  - \* Eli also likes cakes: there is a 60% chance that I will see her eating cakes if there are cakes in the coffee room:  $\mathbb{P}[E = 1|C = 1] = 0.6$

- \* But she never buys herself cakes  $\mathbb{P}[E = 1|C = 0] = 0$
- \* We can depict the dependencies of the this large model



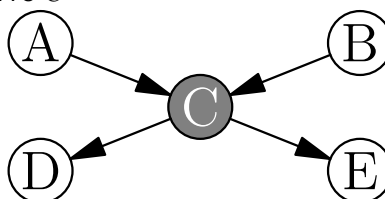
$$\begin{aligned}\mathbb{P}[A, B, C, D, E] &= \mathbb{P}[C, D, E|A, B] \mathbb{P}[B] \mathbb{P}[A] \\ &= \mathbb{P}[D|C] \mathbb{P}[E|C] \mathbb{P}[C|A, B] \mathbb{P}[B] \mathbb{P}[A]\end{aligned}$$

- where we have used the conditional independence
- note that  $D$  and  $E$  are conditionally independent of  $A$  and  $B$  given  $C$
- That is, these probabilities will depend on events  $A$  and  $B$ , but once I know there are cakes in the coffee room it doesn't matter who put them there
- We can compute probabilities for this larger system

$$\mathbb{P}[D = 1] = 0.3121, \quad \mathbb{P}[E = 1] = 0.1818, \quad \mathbb{P}[D = 1, E = 1] = 0.14544$$

so  $\mathbb{P}[D, E] \neq \mathbb{P}[D] \mathbb{P}[E]$

- $D$  and  $E$  are not independent variables as they coupled through  $C$
- However when we observe  $C$



then  $\mathbb{P}[D, E|C] \stackrel{(u)}{=} \mathbb{P}[D|C] \mathbb{P}[E|C]$

- E.g.

$$\mathbb{P}[D = 1|C = 1] = 0.8 \quad \mathbb{P}[E = 1|C = 1] = 0.6 \quad \mathbb{P}[D = 1, E = 1|C = 1] = 0.48$$

## 2.4 Latent Dirichlet Allocation

- Most probabilistic models can be represented as a graphical model
- There are times when this isn't particularly useful
- But it can just help us to understand what is going on
- We consider an example of this called **Latent Dirichlet Allocation**
  - This is sometimes known as LDA, but should not be confused with *linear discriminant analysis*
- LDA is used to model topics in a set of documents (or *corpus*)
- We want to identify a set of topics
- The topics are associated with particular words
- The documents will be associated with a small number of topics
- To model this we build a *generative model*

- This is natural to build
- Although it seems the wrong way around—we don't want to build a corpus of documents
- But Bayes's rule allows us to invert this

- Let us start with some definitions

- We consider generating a corpus of documents

$$\mathcal{C} = \{d_i | i = 1, 2, \dots, |\mathcal{C}|\}$$

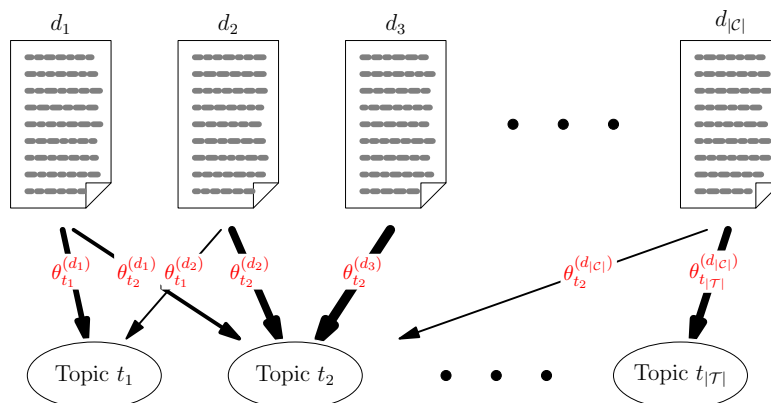
- Each document consists of a set of words

$$d = (w_1^{(d)}, w_2^{(d)}, \dots, w_{N_d}^{(d)})$$

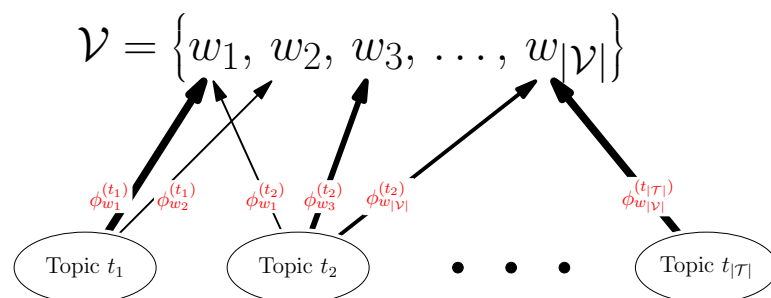
- We assume there is a set of topics

$$\mathcal{T} = \{t_1, t_2, \dots, t_{|\mathcal{T}|}\}$$

- We associate a probability,  $\theta_t^{(d)}$ , that a word in document  $d$  relates to a topic  $t$



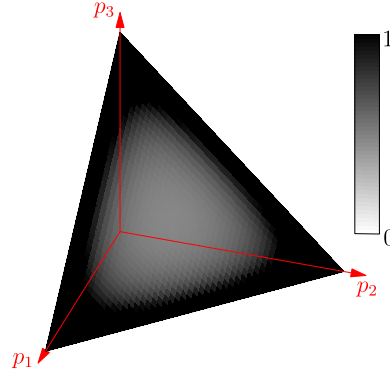
- We associate a probability  $\phi_w^{(t)}$  that a word,  $w$ , is related to a topic  $t$



- Most documents are predominantly about a few topics and most topic have a small number of words associated to them
- We can generate probability vectors  $\theta^{(d)}$  and  $\phi^{(t)}$  from a Dirichlet distribution

$$\text{Dir}(\mathbf{p}|\boldsymbol{\alpha}) = \Gamma\left(\sum_i \alpha_i\right) \prod_{i=1}^n \frac{p_i^{\alpha_i-1}}{\Gamma(\alpha_i)}$$

- $\theta^{(d)} \sim \text{Dir}(\alpha \mathbf{1})$  and  $\phi^{(t)} \sim \text{Dir}(\beta \mathbf{1})$
- By choosing a Dirichlet distribution with a small components,  $\alpha_i$ , we ensure that have most of its probability density lies around the edges



- By drawing  $\theta^{(d)}$  and  $\phi^{(t)}$  from a Dirichlet distribution with small parameters  $\alpha$  and  $\beta$  we ensure that most components are very small with a few large components
- To generate a document we choose a topic for each word and a word for each topic
- We use the categorical distribution
  - \* if  $\mathbf{p}$  is a vector of non-negative values that sum to 1 then  $\text{Cat}(i|\mathbf{p}) = p_i$
  - \* That is if  $I \sim \text{Cat}(\mathbf{p})$  then  $I$  will be an integer,  $i$  with probability  $p_i$
- Thus for word  $i$  of document  $d$  we first choose a topic  $\tau_i^{(d)} \sim \text{Cat}(\theta^{(d)})$  and then we choose a word  $w_i^{(d)} \sim \text{Cat}(\phi^{\tau_i^{(d)}})$ 
  - \* It is a slightly crazy model in that words are randomly chosen from the topics of the document with no ordering
- We could represent this by a rather ugly graphical model (see Figure 1)

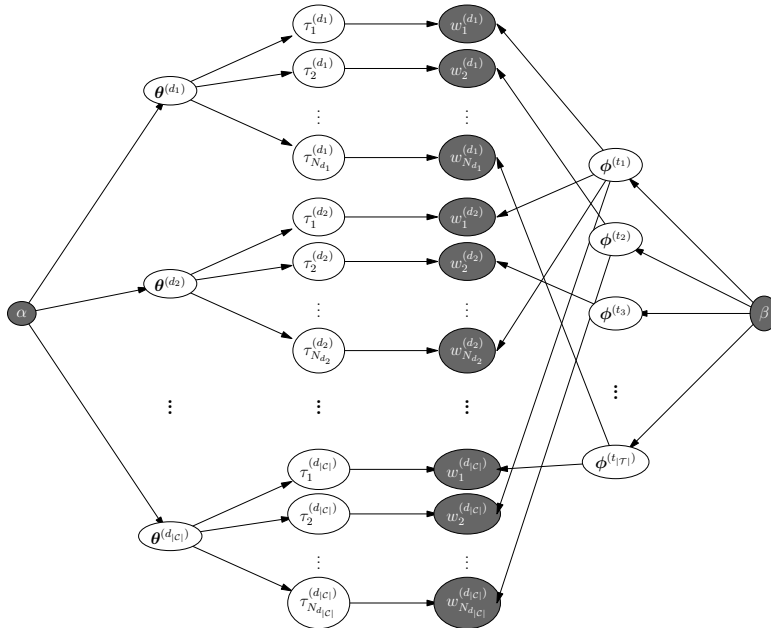
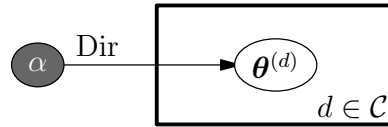


Figure 1: Graphical Model for Latent Dirichlet Allocation

- To make graphical models more manageable people have invented a graphical means of showing repeats
- For example, to illustrate that we have a probability vector  $\theta^{(d)}$  drawn from a Dirichlet distribution with parameter  $\alpha$  for each document  $d$  in our corpus  $\mathcal{C}$  we can use a **plate diagram**



- Using a plate diagram we can represent the LDA as shown in Figure 2

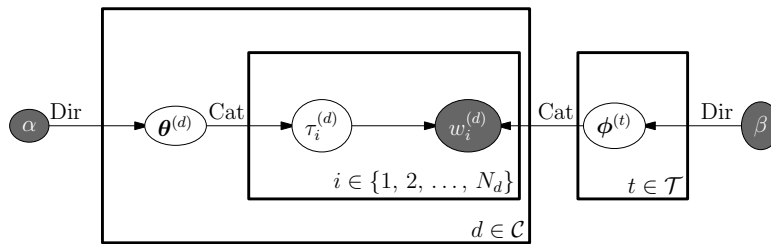


Figure 2: Graphical Model for Latent Dirichlet Allocation

- It takes a bit of time to decode this
  - \* We have a probability vector  $\theta^{(d)}$  for every document in our corpus
    - this tells us the distribution of topics in the document
    - $\theta^{(d)}$  is drawn from a Dirichlet distribution with parameters  $\alpha = (\alpha, \alpha, \alpha, \dots, \alpha)$
  - \* We have a probability  $\phi^{(\tau)}$  for every topic
    - this tells us the distribution of words associated with a topic
    - $\phi^{(\tau)}$  is drawn from a Dirichlet distribution with parameters  $\beta = (\beta, \beta, \beta, \dots, \beta)$
  - \* For each document,  $d$ , and each word,  $w_i^{(d)}$  in the document we have
    - a topic  $\tau_i^{(d)}$  drawn from  $\theta^{(d)}$
    - the words  $w_i^{(d)}$  are drawn from  $\phi^{(\tau_i^{(d)})}$
    - that is it depends both on the topic  $\tau_i^{(d)}$  and on the distributions of words associated with that topic
  - \* In practice I am usually given the documents with words (the words are observed)
  - \* I have shaded what is usually taken to be observed (for  $\alpha$  and  $\beta$  we usually just choose these from the start—we could learn them so they would not be observed)
- The graphical model helps us write down the joint distribution
- We define matrices to denote all the variables

$$\begin{aligned}
\mathbf{W} &= (\mathbf{w}^{(d)} | d \in \mathcal{C}) \quad \text{with} \quad \mathbf{w}^{(d)} = (w_1^{(d)}, w_2^{(d)}, \dots, w_{N_d}^{(d)}), \quad \text{and} \quad w_i^{(d)} \in \mathcal{V} \\
\mathbf{T} &= (\tau_i^{(d)} | d \in \mathcal{C} \wedge i \in \{1, 2, \dots, N_d\}) \quad \text{with} \quad \tau_i^{(d)} \in \mathcal{T} \\
\Theta &= (\theta^{(d)} | d \in \mathcal{C}) \quad \text{with} \quad \theta^{(d)} = (\theta_t^{(d)} | t \in \mathcal{T}) \in \Lambda^{|\mathcal{T}|} \\
\Phi &= (\phi^{(t)} | t \in \mathcal{T}) \quad \text{with} \quad \phi^{(t)} = (\phi_w^{(t)} | w \in \mathcal{V}) \in \Lambda^{|\mathcal{V}|}
\end{aligned}$$

- Then the joint distribution is given by

$$\mathbb{P}[\mathbf{W}, \mathbf{T}, \boldsymbol{\Theta}, \boldsymbol{\Phi} | \alpha, \beta] = \left( \prod_{t \in \mathcal{T}} \text{Dir}(\boldsymbol{\phi}^{(t)} | \beta \mathbf{1}) \right) \left( \prod_{d \in \mathcal{C}} \text{Dir}(\boldsymbol{\theta}^{(d)} | \alpha \mathbf{1}) \prod_{i=1}^{N_d} \text{Cat}(\tau_i^{(d)} | \boldsymbol{\theta}^{(d)}) \text{Cat}(w_i^{(d)} | \boldsymbol{\phi}^{(\tau_i^{(d)})}) \right)$$

- It is now a technical exercise to compute the quantities of interest
- For example  $f(\boldsymbol{\Theta}, \boldsymbol{\Phi} | \mathbf{W}, \alpha, \beta)$  will tell us about the words associated with the topics that are present in the corpus and the topics associated with each document
- Note that we would marginalise out  $\mathbf{T}$
- There are different techniques for computing these probabilities, e.g. using MCMC or variational approximations

### 3 Exercise

#### 3.1 Cakes

- Write a program to compute the probability of various events concerning cakes
- To compute all the probabilities (sometimes inefficiently) we can sum over all values our variables can take
- I have done this somewhat inefficiently in the answers

### 4 Answers

#### 4.1 Cakes

- I am using asymptote which I usually use for drawing diagrams, but its a language with C syntax
- Port this to a language of your choice

```

real pcGab(int a, int b, int c) { // P(C|A,B)
  real p;
  if (a==1 && b==1)
    p = 1;
  else if (a==1 || b==1)
    p = 0.95;
  else
    p = 0.05;
  if (c==1)
    return p;
  else
    return 1-p;
}

real pa(int a) { // P(A)
  return (a==1)? 0.2:0.8;
}

real pb(int b) { // P(B)
  return (b==1)? 0.1:0.9;
}

```



```

}

real pd(int d, int c) { // P(D|C)
    real p = (c==1)? 0.8:0.1;
    return (d==1)? p:1-p;
}

real pe(int e, int c) { // P(E|C)
    real p = (c==1)? 0.6:0;
    return (e==1)? p:1-p;
}

typedef real func(int, int, int, int, int); // define signature of general function

real expect(func f) { // compute expectations exhaustively
    real sum = 0;
    for (int a=0; a<=1; ++a) {
        for (int b=0; b<=1; ++b) {
            for (int c=0; c<=1; ++c) {
                for (int d=0; d<=1; ++d) {
                    for (int e=0; e<=1; ++e) {
                        sum += f(a,b,c,d,e)*pcGab(a,b,c)*pa(a)*pb(b)*pd(d,c)*pe(e,c);
                    }
                }
            }
        }
    }
    return sum;
}

/* Define functions to find expectations */
/* These are all indicator funtions so I end up with probabilities */

real f(int a, int b, int c, int d, int e) {return 1;}
real fa(int a, int b, int c, int d, int e) {return a;}
real fb(int a, int b, int c, int d, int e) {return b;}
real fab(int a, int b, int c, int d, int e) {return a*b;}
real fc(int a, int b, int c, int d, int e) {return c;}
real fac(int a, int b, int c, int d, int e) {return a*c;}
real fbc(int a, int b, int c, int d, int e) {return b*c;}
real fabc(int a, int b, int c, int d, int e) {return a*b*c;}
real fd(int a, int b, int c, int d, int e) {return d;}
real fe(int a, int b, int c, int d, int e) {return e;}
real fde(int a, int b, int c, int d, int e) {return d*e;}
real fcd(int a, int b, int c, int d, int e) {return c*d;}
real fce(int a, int b, int c, int d, int e) {return c*e;}

real fcde(int a, int b, int c, int d, int e) {return c*d*e;}

```

```

write("Check joint probability is normalised: ", expect(f));
write("P(A=1) = ", expect(fa));
write("P(B=1) = ", expect(fb));
write("P(A=1)*P(B=1) = ", expect(fa)*expect(fb));
write("P(A=1,B=1) = ", expect(fab));
write("Note P(A=1,B=1) = P(A=1)*P(B=1)");
write("-");

real Pc = expect(fc);
write("P(C=1) = ", Pc);
real PaGc = expect(fac)/Pc;
real PbGc = expect(fbc)/Pc;
real PabGc = expect(fabc)/Pc;
write("P(A=1|C=1) = ", PaGc);
write("P(B=1|C=1) = ", PbGc);
write("P(A=1|C=1)*P(B=1|C=1) = ", PaGc*PbGc);
write("P(A=1,B=1|C=1) = ", PabGc);
write("Note: P(A=1,B=1|C=1) != P(A=1|C=1)*P(B=1|C=1)");
write("-");

write("P(D=1) = ", expect(fd));
write("P(E=1) = ", expect(fe));
write("P(D=1)*P(E=1) = ", expect(fd)*expect(fe));
write("P(D=1,E=1) = ", expect(fde));
write("Note: P(D=1,E=1) != P(D=1)*P(E=1)");
write("-");

write("P(D=1|C=1) = ", expect(fcd)/Pc);
write("P(E=1|C=1) = ", expect(fce)/Pc);
write("P(D=1|C=1)*P(E=1|C=1) = ", expect(fcd)/Pc*expect(fce)/Pc);
write("P(D=1,E=1|C=1) = ", expect(fcde)/Pc);
write("Note: P(D=1,E=1|C=1) != P(D=1|C=1)*P(E=1|C=1)");

```

## 4.2 Result from program

```

Check joint probability is normalised: 1
P(A=1) = 0.2
P(B=1) = 0.1
P(A=1)*P(B=1) = 0.02
P(A=1,B=1) = 0.02
Note P(A=1,B=1) = P(A=1)*P(B=1)
-
P(C=1) = 0.303
P(A=1|C=1) = 0.63036303630363
P(B=1|C=1) = 0.316831683168317
P(A=1|C=1)*P(B=1|C=1) = 0.19971898179917
P(A=1,B=1|C=1) = 0.066006600660066
Note: P(A=1,B=1|C=1) != P(A=1|C=1)*P(B=1|C=1)
-
P(D=1) = 0.3121
P(E=1) = 0.1818
P(D=1)*P(E=1) = 0.05673978
P(D=1,E=1) = 0.14544
Note: P(D=1,E=1) != P(D=1)*P(E=1)

```

-

$$P(D=1|C=1) = 0.8$$

$$P(E=1|C=1) = 0.6$$

$$P(D=1|C=1)*P(E=1|C=1) = 0.48$$

$$P(D=1, E=1|C=1) = 0.48$$

$$\text{Note: } P(D=1, E=1|C=1) = P(D=1|C=1)*P(E=1|C=1)$$