1 Dependencies of Random Variables

Most of the models we've looked at thus far have assumed that our inputs $x_1, ..., x_n$ are i.i.d., which allows us to factorise the likelihood function into a product over each x_i:

$$p(x_1, .., x_n|\theta) = \prod_{i=1}^{n} p(x_i|\theta).$$

This is a very useful assumption (for example, the simple likelihood function above often makes finding the MLE for parameters θ much easier), however, its simplicity means that we severely limit the situations that we are able to tackle. To improve upon this we will discuss graphical representations of the dependencies between random variables (e.g. our inputs). First, however, we will recall the definitions of independence and conditional independence.

Let X, Y and Z be random variables.

- X is independent of $Y \Leftrightarrow X \perp Y \Leftrightarrow p(X, Y) = p(X)p(Y)$.

  We can think of this as meaning that there is no information exchange between X and Y since:

  $$X \perp Y \Leftrightarrow p(X|Y) = P(X) \Leftrightarrow p(Y|X) = P(Y).$$

- X is independent of Y given $Z \Leftrightarrow X \perp Y|Z \Leftrightarrow p(X, Y|Z) = p(X|Z)p(Y|Z)$.

  As with full independence above, we can similarly factorise the joint distribution of X, Y, Z since $X \perp Y|Z \Leftrightarrow p(X, Y, Z) \propto g_1(X, Z)g_2(Y, Z)$ for some functions g_1, g_2.

  We can also think of this as meaning that there is no information exchanged between X and Y that can't be gained from Z instead since:

  $$X \perp Y|Z \Leftrightarrow p(X|Y, Z) = P(X|Z) \Leftrightarrow p(Y|X, Z) = P(Y|Z).$$

2 Markov Networks

Consider an (undirected) graph $G = (V, E)$ in which the vertex set V contains random variables. Given three subsets of random variables $X, Y, Z \subseteq V$, if no path between X and Y exists that doesn't pass through a member of Z (that is if X and Y are completely "blocked" by Z) then we say that $X \perp Y|Z$ is represented by the graph.

For example, the graph shown in Figure 1 represents the following list of conditional independence (we also give the corresponding factorisation as in the definition of conditional independence given in the previous section):

1. $A \perp B|C \Leftrightarrow p(A, B, C) \propto g_1(A, C)g_2(B, C)$.
2. $A \perp B|C, D \Leftrightarrow p(A, B, C, D) \propto g_3(A, C, D)g_4(B, C, D)$.
3. $A \perp B, D|C \Leftrightarrow p(A, B, C, D) \propto g_5(A, C)g_6(B, C, D)$.
4. $A \perp D|C \Leftrightarrow p(A, C, D) \propto g_7(A, C)g_8(D, C)$.
5. $A \perp D|B, C \Leftrightarrow p(A, B, C, D) \propto g_9(A, B, C)g_{10}(B, C, D)$.

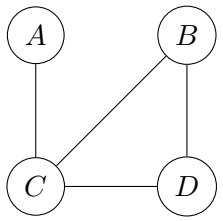


Figure 1: An example graph.

Note that each of the factorisations of $p(A, B, C, D)$ is proportional to functions of *cliques* (fully-connected subgraphs) in G (the non-clique $\{A, C, D\}$ within g_3 is proportional to functions g_7 and g_8 of the cliques $\{A, C\}$ and $\{D, C\}$ as shown in the 4th item of the list).

In fact, we can extend this idea to say that a collection of random variables X factorises over a graph $G = (X, E)$ if

$$p(X) \propto \prod_{c \in C} g_c(X^{(c)})$$

where C is the set of all cliques in G and g_c is some function defined on the subset $X^{(c)}$ of X restricted on c. We note that if one clique is contained within another clique, the function over the larger clique will absorb the function in the smaller clique, so we only need to take out product over the set of maximal cliques in G (cliques which are not part of a larger clique).

In Figure 1 we have the maximal cliques $\{A, C\}$ and $\{B, C, D\}$, so we can see that the factorisation of $p(A, B, C, D)$ over the given graph can be written as $p(A, B, C, D) \propto g_5(A, C)g_6(B, C, D)$ (as in the list given above).

It is important to note that a joint probability distribution p factorises over a graph G if and only if p satisfies all conditional independence represented by G—the two notions are equivalent. In this case, we call the probability distribution an *undirected graphical model* or a *Markov network*.

2.1 Gaussian Markov Network

Consider the example where a Markov network has vertices representing the entries of a multivariate Gaussian random variable $\mathbf{x} \sim \mathcal{N}_{\mathbf{x}}(\mathbf{0}, \mathbf{\Sigma})$. Then defining the precision of $\mathbf{x}$ as $\mathbf{\Theta} = \mathbf{\Sigma}^{-1}$ we can see that

$$p(\mathbf{x}) \propto \exp(-\frac{1}{2}\mathbf{x}^T\mathbf{\Theta}\mathbf{x}) = \exp\left(-\frac{1}{2}\sum_{u,v}\mathbf{\Theta}^{(u,v)}x^{(u)}x^{(v)}\right) \propto \prod_{u,v:\mathbf{\Theta}^{(u,v)}\neq 0} \exp(-\mathbf{\Theta}^{(u,v)}x^{(u)}x^{(v)}).$$

That is,

$$p(\mathbf{x}) \propto \prod_{u,v:\mathbf{\Theta}^{(u,v)}\neq 0} g_{u,v}(x^{(u)}x^{(v)}),$$

meaning that $p(\mathbf{x})$ factorises over the graph G with the adjacency matrix

$$A^{(u,v)} = \begin{cases} 0 & \text{if } \mathbf{\Theta}^{(u,v)} = 0 \\ 1 & \text{if } \mathbf{\Theta}^{(u,v)} \neq 0. \end{cases}$$

From this we can see that the sparsity of $\mathbf{\Theta}$ corresponds to the sparsity of G[1]. For example, a graph G containing three edges and five nodes represents a multivariate Gaussian distribution with 11 non-zero elements in the inverse covariance matrix (5 of these are on the diagonal and the other 6 come in 3 pairs each corresponding to an edge in the graph).

We can also use this to infer the conditional independence present in a dataset by estimating a precision matrix $\mathbf{\Theta}$ and analysing its sparsity. Constructing a graph from this precision matrix when it is derived by an L_1-regularised MLE, we arrive at the *Graphical Lasso* technique.

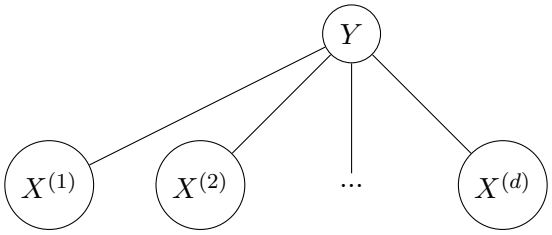
2.2 Conditional Markov Network

We can extend this analysis to conditional probability distributions by saying that $p(Y|X)$ factorises over G with vertices $X \cup Y$ if

$$p(Y|X) = \frac{1}{N(X)}\prod_{c \in C} g_c(V_c)$$

[1] Also note that A is symmetric and G is undirected, which comes about because $\mathbf{\Sigma}$ and $\mathbf{\Theta}$ are symmetric.

Figure 2: A graphical model for logistic regression.



where $N(X) := \int \prod_{c \in C} g_c(V_c) dY$ is a normalising constant.

Note that with this definition, we can cancel out factors defined on subsets of the conditioning variable X, e.g. if $p(Y|X) = \frac{1}{N(X)} g_1(Y, X) g_2(X)$ then $N(X) = \int g_1(Y, X) g_2(X) dY = g_2(X) \int g_1(Y, X) dY$ meaning that

$$p(Y|X) = \frac{g_1(Y, X) g_2(X)}{N(X)} = \frac{g_1(Y, X) g_2(X)}{g_2(X) \int g_1(Y, X) dY} = \frac{g_1(Y, X)}{\int g_1(Y, X) dY}.$$

2.3 Logistic Regression

We can use conditional Markov networks to derive logistic regression by considering the graphical model shown in Figure 2 where $Y \in \{-1, +1\}$ and $X \in \mathbb{R}^d$.

This leads to the factorisation

$$p(Y|X) = \frac{\prod_{i=1}^d g_i(Y, X^{(i)})}{\sum_{Y' \in \{-1,+1\}} g_i(Y', X^{(i)})}.$$

Thus if we let $g_i(Y = y, X^{(i)} = x^{(i)}; \beta_i, \beta_0) := \exp(y(\beta_i x^{(i)} + \beta_0)$ then this factorisation becomes

$$\begin{aligned} p(y|\mathbf{x}; \boldsymbol{\beta}, \beta_0) = \frac{\prod_{i=1}^d \exp(y(\beta_i x^{(i)} + \beta_0))}{\sum_{y' \in \{-1,+1\}} \exp(y'(\beta_i x^{(i)} + \beta_0))} &= \frac{\exp(y(\langle \boldsymbol{\beta}, \mathbf{x} \rangle + d\beta_0))}{\exp(\langle \boldsymbol{\beta}, \mathbf{x} \rangle + d\beta_0) + \exp(-\langle \boldsymbol{\beta}, \mathbf{x} \rangle - d\beta_0)} \\ &= \frac{1}{\frac{\exp(\langle \boldsymbol{\beta}, \mathbf{x} \rangle + d\beta_0)}{\exp(y(\langle \boldsymbol{\beta}, \mathbf{x} \rangle + d\beta_0))} + \frac{\exp(-\langle \boldsymbol{\beta}, \mathbf{x} \rangle - d\beta_0)}{\exp(y(\langle \boldsymbol{\beta}, \mathbf{x} \rangle + d\beta_0))}} \\ &= \frac{1}{1 + \exp(-2y(\langle \boldsymbol{\beta}, \mathbf{x} \rangle + d\beta_0))} \\ &= \sigma(2y(\langle \boldsymbol{\beta}, \mathbf{x} \rangle + d\beta_0)). \end{aligned}$$

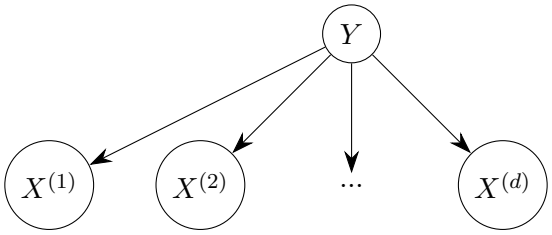
Finding the MLEs of $\boldsymbol{\beta}$ and β_0 is therefore the same logistic regression procedure as we saw in Lecture 11/Portfolio 3 since it takes the following form (note that $2y(\langle \boldsymbol{\beta}, \mathbf{x} \rangle + d\beta_0)$ is a linear function of $\mathbf{x}_i$, which is wrapped in the nonlinear sigmoid function σ):

$$\hat{\boldsymbol{\beta}}, \hat{\beta}_0 := \underset{\boldsymbol{\beta}, \beta_0}{\operatorname{argmax}} \sum_{i=1}^n \log p(y_i|\mathbf{x}_i; \boldsymbol{\beta}, \beta_0) = \underset{\boldsymbol{\beta}, \beta_0}{\operatorname{argmax}} \sum_{i=1}^n \log \sigma(2y(\langle \boldsymbol{\beta}, \mathbf{x}_i \rangle + d\beta_0)).$$

3 Bayesian Networks

So far we've only considered undirected graphs, but some dependencies (particularly causal relationships) are better represented by directed graphical models—in particular we'll be

Figure 4: A graphical model for Naïve Bayes.



considering *directed acyclic graphs* (DAGs) $G = (V, E)$ (which contain no directed cycles with E being a directed edge set) called Bayesian Networks.

Within a DAG we say that if the edge (a, b) (i.e. $a \to b$) exists, then a is the *parent* of b and b is the *child* of a. Furthermore, if there exists a directed path in G from a to b then we say that b is a *descendant* of a.

We say that a probability distribution $p(X)$ factorises over a DAG $G = (V, E)$ if

$$p(X) = \prod_{v \in V} p(X_v | X_{\text{parent}(X_v)})$$

where X_v is the random variable represented by vertex v and $X_{\text{parent}(X_v)}$ is the set of random variables represented by the parents of v.

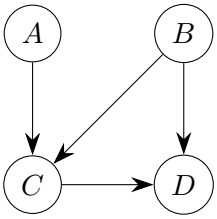


Figure 3: An example DAG.

Thus the DAG in Figure 3 represents the factorisation

$$p(A, B, C, D) = p(A)p(B)p(C|A, B)(D|C, B).$$

Much like with Markov Networks, in Bayesian Networks there is an equivalency between factorisations of a probability distribution and conditional independence of random variables represented by the graphical model. In particular, a probability distribution $p(X)$ factorises over a DAG G if and only if it satisfies all conditional independence represented in G as:

$$\forall v \in V : X_v \perp X_{\text{non-descendants}(X_v)} | X_{\text{parent}(X_v)}$$

In Figure 3 this corresponds to $A \perp B$ and $D \perp A|B, C$.

3.1 Naïve Bayes

Just as we found that the Markov Network in Figure 2 leads to a form of logistic regression when finding $p(Y|X)$, by adding directions to the edges (as shown in Figure 4) the conditional probability $p(Y|X)$ takes the same form as in Naïve Bayes:

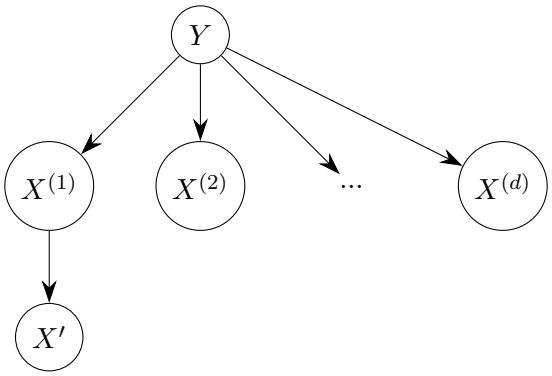
$$p(Y|X) = \frac{p(Y) \prod_{i=1}^{d} p(X^{(i)}|Y)}{p(X)}.$$

Logistic regression and Naïve Bayes both predict an output class $\hat{Y} := \text{argmax}_Y \, p(Y|X)$ using almost the same graphical representation, except that Naïve Bayes factorises $p(Y|X)$ (calculating it up to a constant factor) using conditional probability of edges in a DAG whereas logistic regression calculates $p(Y|X)$ exactly using factorisations based on cliques of an undirected graph (whose only cliques are actually just pairs of neighbouring vertices Y and $X^{(i)}$).

(In Appendix A we extend the example in Figure 4 to explore how we might decide on which input features to use in a classification task.)

A Redundant Features

Figure 5: A graphical model for Naïve Bayes with a redundant input feature.



In this appendix we see that graphical models can help us to identify redundant features when performing classification. Observe that Figure 5 represents the factorisation

$$p(Y, X, X') = p(X'|X^{(1)})P(Y)\prod_{i=1}^{d} p(X^{(i)}|Y).$$

When predicting $\hat{Y} := \operatorname{argmax}_Y p(Y|X)$ we then see that

$$\begin{aligned}\hat{Y} := \operatorname*{argmax}_Y p(Y|X, X') = \operatorname*{argmax}_Y \frac{p(Y, X, X')}{p(X, X')} &= \operatorname*{argmax}_Y \frac{p(X'|X^{(1)})p(Y)\prod_{i=1}^{d} p(X^{(i)}|Y)}{p(X, X')} \\ &= \operatorname*{argmax}_Y p(Y)\prod_{i=1}^{d} p(X^{(i)}|Y).\end{aligned}$$

That is, we can ignore the X' feature as it will not change the prediction $\hat{Y}$ (given that we do include $X^{(1)}$) since

$$\frac{p(X'|X^{(1)})p(Y)\prod_{i=1}^{d} p(X^{(i)}|Y)}{p(X, X')} \propto p(Y)\prod_{i=1}^{d} p(X^{(i)}|Y),$$

therefore its inclusion will only serve to increase computation time.