Bayesian Applied Methods

Student

Máster en Ciencia de Datos

Universidad Autónoma de Madrid

tu-web.es

Contents

Part I. Introduction

1. Introduction

Consider a set of variables $X_1, \dots, X_n$, we will define a model that will consider the joint probability density function of the variables

$$P(X_1, \dots, X_n)$$

There are a few forms of representing this

- Bayesian Networks, which are represented by directed graphs in which each node represents conditional probabilities, and the edges represent dependencies. (Drawing from the blackboard)
- Markov Networks. In this case, we have undirected graphs, where the edges are *factors* (tables of probability)

Bayesian networks have the *factors* in the nodes, while Markov networks have the factors in the edges.

Our goal will be to make inference about variables using the available information. Sometimes, making inference is understood as **marginalizing the joint distribution**.

$$P(A) = \sum_{B,C} P(A, B, C) = \sum_{B,C} P(A)P(B|A)P(C|B, A)$$

There are different algorithms to compute this probabilities, such as *variable elimination* or *message-passing*.

Also, there are different ways of **reasoning**, such as

- Causal reasonament, which studies causalities
- Evidencial reasonament

With all this types of reasoning, we are assuming **two main points**:

- That we have all the information about both the structure of the network and the *factors* or probabilities.

Having both the network structure and the probabilities of each of the factors, we can marginalize to obtain the joint probability distributions. However, we can consider a case where we do **not** know the structure of the network but we know the probabilities of the factors. This is also a *branch* of study, which we will not go deep in. A last case is the one where we **know** the structure of the network, but we are **lacking** parts of the table which we would like to infere. This is the case that we will focus in this course.

1.1. Example

Consider the following random variable, modeling the probability of obtaining heads or tails in a coinflip

$$v^n = \begin{cases} 1 & \text{heads} \\ 0 & \text{tails} \end{cases}$$

However, our coin might have different weights for each of the outcomes (biased coin). For instance, consider that $\theta = P(\text{heads}) = P(v^n = 1|\theta)$. Hence, $P(\text{tails}) = 1 - \theta$. In this case, our goal would be to **determine** θ.

If we tossed the coin n times, and **we knew the probability** θ, the coin tosses would be **independent**. However, if we **do not know** the probability θ, the coin tosses **would be dependent** since the **outcome** of the experiment affects θ. (Diagrams from the blackboard). In this case, the joint probability would be

$$P(\theta, v^1, \ldots, v^n) = P(\theta) \prod_{i=1}^{n} P(v^i|\theta)$$

Estimations of the parameters are sometimes done using the empirical distribution function. However, there are other methods of estimating the joint pdf, such as *maximum likelihood estimation*.

1.2. Maximum likelihood estimation

We define the **likelihood** of the data as

$$L(\theta; D) = P(D|\theta) = \prod_{j=1}^{N} P(v^j|\theta)$$

Maximum likelihood estimation determines the likelihood function and tries to find (or approximate) a maximum of it.

We can generalize our previous coin toss example. Consider that we obtained M_h heads and M_t tails in our coin toss problem. In this case, our likelihood function would be

$$L(\theta; M_h, M_t) = \theta^{M_h}(1-\theta)^{M_t}$$

The most common approach is to apply the logarithm to the likelihood function, which is a monotonous increasing function, to convert the product into sums, and then maximize the **log-likelihood**. In the previous example, our log-likelihood would be

$$\ell = \log L(\theta; M_h, M_t) = M_h \log \theta + M_t \log(1-\theta)$$

Consider the case of a three node bayesian network, with joint pdf:

$$P(A,B,Y) = P(A)P(B|A)P(Y|A,B).$$

We would like to obtain the probabilities in a table in each case. Extracting data is to observe (N) realizations of an experiment. We would like to use this data to determine $\theta_A, \theta_B, \theta_{Y|A,B}$. Recall that these θs **are not distribution parameters but computed probabilities of observations**. We estimate this parameter set:

$$\Theta = \{\theta_A, \theta_B, \theta_{Y|A,B}\}$$

considering that we have to compute θ_a for all $a \in Values(A)$, $theta_b$ for all $b \in Values(B)$ and $\theta_{y|a,b}$ for all $a \in Values(A), b \in Values(b)$ and $y \in Values(Y)$

Let us compute the likelihood of this Θ:

$$\begin{aligned} L(\Theta, D) &= P(D|\Theta) \\ &= \prod_{j=1}^{N} P(a[j], b[j], y[j]|\Theta) \\ &= \prod_{j=1}^{N} P(a[j]|\Theta)P(b[j]|\Theta)P(y[j]|a[j], b[j], \Theta) \\ &= \prod_{j=1}^{N} P(a[j]|\Theta_A)P(b[j]|\Theta_B)P(y[j]|a[j], b[j], \Theta_{Y|A,B}) \\ &= \prod_{j=1}^{N} L(\theta_A, D)L(\theta_B, D)L(\theta_{Y|A,B}, D) \end{aligned}$$

So we have expressed the likelihood of the parameters Θ as the product of the likelihood of the individual parameters θ_i. We can extend this to a more teneral case.

Proposition.- Let $X_1, \ldots, X_K$ be random variables with a bayesian network dependence. Let D be a sample. Let $\tilde{U} = \mathrm{Par}_g(X_i)$. Then,

$$\begin{aligned} L(\Theta, D) &= \prod_{j=1}^{N} P(\tilde{x}[j]|\Theta) \\ &= \prod_{j=1}^{N}\prod_{i=1}^{K} P(x[j]|\tilde{u}_i[j], \Theta_i) \\ &= \prod_{i=1}^{K} L(\Theta_i|D) \end{aligned}$$

Proposition.- The MLE of the general case of a bayesian network is given by:

$$\Theta_{x|\tilde{u}} = \frac{M[X = x, \tilde{U} = \tilde{u}]}{M[\tilde{U} = \tilde{u}]}$$

where M is the counting function.

Maximum likelihood estimation has a few limitations:

1. Fragmentation: The number of values to estimate de CDP table increases exponentially with the number of parents $|\tilde{U}|$. When the number of observations is low, the probability estimation is very poor and some possible values may even not appear. A common solution is to represent the data in a smaller number of states.
2. Overfitting

Example: spam detection using KNN.

INCLUDE IMAGE!!!

Consider a sample of $1M$ examples. If we had $K = 30$ nodes, we would have to stimate 2^{30} parameters. This is clearly not factible.

We want to solve this using **bayesian estimation**.

2. The bayesian approach

Until now, we have been dealing with the likelihood of our data $L(\Theta; D)$. In the bayesian approach, we try to compute the **posterior probability** using the Bayes' theorem:

$$P(\Theta|D) = \frac{P(D|\Theta)P(\Theta)}{P(D)}$$

where $P(\Theta)$ is the prior probability of our parameter, $P(D)$ is the proabability of the data, and $P(D|\Theta)$ is the likelihood.

To achieve this, we **need a good estimation of the prior probability** $P(\Theta)$. We will know study a few distributions that can be used as prior distributions:

2.1. Beta and Dirichlet distributions

This is the univariate version of the Dirichlet distirbution. It is given by:

$$f(x; \alpha, \beta) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha, \beta)}$$

where

$$B(\alpha, \beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)} = \int_0^1 x^{\alpha-1}(1-x)^{\beta-1}$$

is essentially a normalizing factor.

We generalize this formula to obtain the Dirichlet distribution. (Complete from jupyter notebook)

Why is this distirbution interesting?

Definition.- If the posterior distribution $P(\theta|D)$ is in the same probability distribution family as the prior probability distirbution $P(\theta)$, then the prior and the posterior are called **conjugate distributions**, and the prior is called a **conjugate prior** for the likelihood function $P(D|\Theta)$.

Let us see this in an example using the beta distribution in the biased coin example:

$$P(\Theta|D) = \frac{\theta^{M_h}(1-\theta)^{M_t} \cdot \theta^{\alpha-1}(1-\theta)^{\beta-1}}{P(D)} \frac{1}{B(\alpha,\beta)} = \frac{1}{P(D)B(\alpha,\beta)}\theta^{M_h+\alpha-1}(1-\theta)^{M_t+\beta-1}$$

Let us compute the probabiltiy of the data

$$\begin{aligned} P(D) &= \int_0^1 P(D|\theta)P(\theta)d\theta \\ &= \frac{1}{B(\alpha,\beta)} \int_0^1 \theta^{M_h+\alpha-1}(1-\theta)^{M_t+\beta-1}d\theta \\ &= \frac{1}{B(\alpha,\beta)} \frac{\Gamma(M_h+\alpha)\Gamma(M_t+\beta)}{\Gamma(M_h+M_t+\alpha+\beta)} \end{aligned}$$

We use it back in the previous expression

$$\begin{aligned} \frac{1}{P(D)B(\alpha,\beta)}\theta^{M_h+\alpha-1}(1-\theta)^{M_t+\beta-1} &= \frac{\Gamma(M_h+\alpha)\Gamma(M_t+\beta)}{\Gamma(M_h+M_t+\alpha+\beta)}\theta^{M_h+\alpha-1}(1-\theta)^{M_t+\beta-1} \\ &= B(\theta; M_h+\alpha, M_t+\beta) \end{aligned}$$

3. Introduction

Consider a set of variables $X_1, \ldots, X_n$, we will define a model that will consider the joint probability density function of the variables

$$P(X_1, \ldots, X_n)$$

There are a few forms of representing this

- Bayesian Networks, which are represented by directed graphs in which each node represents conditional probabilities, and the edges represent dependencies. (Drawing from the blackboard)
- Markov Networks. In this case, we have undirected graphs, where the edges are *factors* (tables of probability)

Bayesian networks have the *factors* in the nodes, while Markov networks have the factors in the edges.

Our goal will be to make inference about variables using the available information. Sometimes, making inference is understood as **marginalizing the joint distribution**.

$$P(A) = \sum_{B,C} P(A,B,C) = \sum_{B,C} P(A)P(B|A)P(C|B,A)$$

There are different algorithms to compute this probabilities, such as *variable elimination* or *message-passing*.

Also, there are different ways of **reasoning**, such as

- Causal reasonament, which studies causalities
- Evidencial reasonament

With all this types of reasoning, we are assuming **two main points**:

- That we have all the information about both the structure of the network and the *factors* or probabilities.

Having both the network structure and the probabilities of each of the factors, we can marginalize to obtain the joint probability distributions. However, we can consider a case where we do **not** know the structure of the network but we know the probabilities of the factors. This is also a *branch* of study, which we will not go deep in. A last case is the one where we **know** the structure of the network, but we are **lacking** parts of the table which we would like to infere. This is the case that we will focus in this course.

3.1. Example

Consider the following random variable, modeling the probability of obtaining heads or tails in a coinflip

$$v^n = \begin{cases} 1 & \text{heads} \\ 0 & \text{tails} \end{cases}$$

However, our coin might have different weights for each of the outcomes (biased coin). For instance, consider that $\theta = P(\text{heads}) = P(v^n = 1|\theta)$. Hence, $P(\text{tails}) = 1 - \theta$. In this case, our goal would be to **determine** θ.

If we tossed the coin n times, and **we knew the probability** θ, the coin tosses would be **independent**. However, if we **do not know** the probability θ, the coin tosses **would be dependent** since the **outcome** of the experiment affects θ. (Diagrams from the blackboard). In this case, the joint probability would be

$$P(\theta, v^1, \dots, v^n) = P(\theta) \prod_{i=1}^{n} P(v^i|\theta)$$

Estimations of the parameters are sometimes done using the empirical distribution function. However, there are other methods of estimating the joint pdf, such as *maximum likelihood estimation*.

3.2. Maximum likelihood estimation

We define the **likelihood** of the data as

$$L(\theta; D) = P(D|\theta) = \prod_{j=1}^{N} P(v^j|\theta)$$

Maximum likelihood estimation determines the likelihood function and tries to find (or approximate) a maximum of it.

We can generalize our previous coin toss example. Consider that we obtained M_h heads and M_t tails in our coin toss problem. In this case, our likelihood function would be

$$L(\theta; M_h, M_t) = \theta^{M_h}(1-\theta)^{M_t}$$

The most common approach is to apply the logarithm to the likelihood function, which is a monotonous increasing function, to convert the product into sums, and then maximize the **log-likelihood**. In the previous example, our log-likelihood would be

$$\ell = \log L(\theta; M_h, M_t) = M_h \log\theta + M_t \log(1-\theta)$$

Consider the case of a three node bayesian network, with joint pdf:

$$P(A, B, Y) = P(A)P(B|A)P(Y|A, B).$$

We would like to obtain the probabilities in a table in each case. Extracting data is to observe (N) realizations of an experiment. We would like to use this data to determine $\theta_A, \theta_B, \theta_{Y|A,B}$. Recall that these θs **are not distribution parameters but computed probabilities of observations**. We estimate this parameter set:

$$\Theta = \{\theta_A, \theta_B, \theta_{Y|A,B}\}$$

considering that we have to compute θ_a for all $a \in Values(A)$, $theta_b$ for all $b \in Values(B)$ and $\theta_{y|a,b}$ for all $a \in Values(A)$, $b \in Values(b)$ and $y \in Values(Y)$

Let us compute the likelihood of this Θ:

$$\begin{aligned}
L(\Theta, D) &= P(D|\Theta) \\
&= \prod_{j=1}^{N} P(a[j], b[j], y[j]|\Theta) \\
&= \prod_{j=1}^{N} P(a[j]|\Theta)P(b[j]|\Theta)P(y[j]|a[j], b[j], \Theta) \\
&= \prod_{j=1}^{N} P(a[j]|\Theta_A)P(b[j]|\Theta_B)P(y[j]|a[j], b[j], \Theta_{Y|A,B}) \\
&= \prod_{j=1}^{N} L(\theta_A, D)L(\theta_B, D)L(\theta_{Y|A,B}, D)
\end{aligned}$$

So we have expressed the likelihood of the parameters Θ as the product of the likelihood of the individual parameters θ_i. We can extend this to a more teneral case.

Proposition.- Let $X_1, \ldots, X_K$ be random variables with a bayesian network dependence. Let D be a sample. Let $\tilde{U} = \mathrm{Par}_g(X_i)$. Then,

$$\begin{aligned} L(\Theta, D) &= \prod_{j=1}^{N} P(\tilde{x}[j] | \Theta) \\ &= \prod_{j=1}^{N} \prod_{i=1}^{K} P(x[j] | \tilde{u}_i[j], \Theta_i) \\ &= \prod_{i=1}^{K} L(\Theta_i | D) \end{aligned}$$

Proposition.- The MLE of the general case of a bayesian network is given by:

$$\Theta_{x|\tilde{u}} = \frac{M[X = x, \tilde{U} = \tilde{u}]}{M[\tilde{U} = \tilde{u}]}$$

where M is the counting function.

Maximum likelihood estimation has a few limitations:

1. Fragmentation: The number of values to estimate de CDP table increases exponentially with the number of parents $|\tilde{U}|$. When the number of observations is low, the probability estimation is very poor and some possible values may even not appear. A common solution is to represent the data in a smaller number of states.
2. Overfitting

Example: spam detection using KNN.

INCLUDE IMAGE!!!

Consider a sample of $1M$ examples. If we had $K = 30$ nodes, we would have to stimate 2^{30} parameters. This is clearly not factible.

We want to solve this using **bayesian estimation**.

4. The bayesian approach

Until now, we have been dealing with the likelihood of our data $L(\Theta; D)$. In the bayesian approach, we try to compute the **posterior probability** using the Bayes' theorem:

$$P(\Theta | D) = \frac{P(D|\Theta) P(\Theta)}{P(D)}$$

where $P(\Theta)$ is the prior probability of our parameter, $P(D)$ is the proabability of the data, and $P(D|\Theta)$ is the likelihood.

To achieve this, we **need a good estimation of the prior probability** $P(\Theta)$. We will know study a few distributions that can be used as prior distributions:

4.1. Beta and Dirichlet distributions

This is the univariate version of the Dirichlet distirbution. It is given by:

$$f(x;\alpha,\beta) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha,\beta)}$$

where

$$B(\alpha,\beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)} = \int_0^1 x^{\alpha-1}(1-x)^{\beta-1}$$

is essentially a normalizing factor.

We generalize this formula to obtain the Dirichlet distribution. (Complete from jupyter notebook)

Why is this distirbution interesting?

Definition.- If the posterior distribution $P(\theta|D)$ is in the same probability distribution family as the prior probability distirbution $P(\theta)$, then the prior and the posterior are called **conjugate distributions**, and the prior is called a **conjugate prior** for the likelihood function $P(D|\Theta)$.

Let us see this in an example using the beta distribution in the biased coin example:

$$P(\Theta|D) = \frac{\theta^{M_h}(1-\theta)^{M_t}\cdot\theta^{\alpha-1}(1-\theta)^{\beta-1}}{P(D)}\frac{1}{B(\alpha,\beta)} = \frac{1}{P(D)B(\alpha,\beta)}\theta^{M_h+\alpha-1}(1-\theta)^{M_t+\beta-1}$$

Let us compute the probabiltiy of the data

$$\begin{aligned}
P(D) &= \int_0^1 P(D|\theta)P(\theta)d\theta \\
&= \frac{1}{B(\alpha,\beta)}\int_0^1 \theta^{M_h+\alpha-1}(1-\theta)^{M_t+\beta-1}d\theta \\
&= \frac{1}{B(\alpha,\beta)}\frac{\Gamma(M_h+\alpha)\Gamma(M_t+\beta)}{\Gamma(M_h+M_t+\alpha+\beta)}
\end{aligned}$$

We use it back in the previous expression

$$\begin{aligned}
\frac{1}{P(D)B(\alpha,\beta)}\theta^{M_h+\alpha-1}(1-\theta)^{M_t+\beta-1} &= \frac{\Gamma(M_h+\alpha)\Gamma(M_t+\beta)}{\Gamma(M_h+M_t+\alpha+\beta)}\theta^{M_h+\alpha-1}(1-\theta)^{M_t+\beta-1} \\
&= B(\theta;M_h+\alpha,M_t+\beta)
\end{aligned}$$

4.2. Prediction using Bayesian Networks

Consider that we have observed n random variables $v[1],\dots,v[n]$, we would like to predict the following value $v[n+1]$. This can be expressed as:

$$P(v[n+1]=1|v[1],\dots,v[n]) = \int_0^1 P(v[n+1]=1|\theta)P(\theta|v[1],\dots,v[n])$$

In the biased coin toss, we recall that $P(v[n+1]|\theta)=\theta$ and $P(\theta|v[1],\dots,v[n]) = P(\theta|D)\sim\beta(\theta;M_h+\alpha,m_t+\beta)$. Thus, the last expression follows:

$$\begin{aligned}\int_0^1 P(v[n+1]=1|\theta)P(\theta|v[1],\dots,v[n]) &= \int_0^1 \theta\frac{\Gamma(M_h+M_t+\alpha+\beta)}{\Gamma(M_h+\alpha)\Gamma(M_t+\beta)}\theta^{M_h+\alpha-1}(1-\theta)^{M_t+\beta-1}d\theta\\ &= \frac{\Gamma(M_h+M_t+\alpha+\beta)}{\Gamma(M_h+\alpha)\Gamma(M_t+\beta)}\int_0^1\theta^{M_h+\alpha-1}(1-\theta)^{M_t+\beta-1}d\theta\\ &= \cdots\\ &= \frac{M_h+\alpha}{M_h+M_t+\alpha+\beta}\end{aligned}$$

The conclusion is that in this example, we only need to know the previous values to estimate the following example. All this calculations are only a proof of the expectation of a beta random variable for a particular case.

4.3. Naive Bayes

Consider the INSERTAR DIBUJO . Consider that X_i are gaussian distributions, that is

$$P(X_i|C)=\mathcal{N}(\mu,\sigma^2)$$

In this case, we would have the following SECOND DIAGRAM where $X_i = \{X_i[1],\dots,x_i[n]\}$. Clearly, the parameters to seek in this case are $\theta_{X|C}=\mu_i,\sigma_i^2$. Let us compute the likelihood, assuming independence:

$$L(\mu_i,\sigma_i^2;D) = P(x_i[1],\dots,x_1[n]|\mu_i,\sigma_i^2,C) = \prod_{j=1}^N \mathrm{pdf}_G(x_i;\mu_i,\sigma_i^2)$$

where pdf_G stands for the probability density function of a Gaussian distribution with parameters μ_i,σ_i^2. Hence, the log-likelihood is

$$\begin{aligned}\ell(\mu_i,\sigma_i^2;D) &= \log\left(L(\mu_i,\sigma_i^2,D)\right)\\ &= \log\left(\prod_{j=1}^N \mathrm{pdf}_G(x_i;\mu_i,\sigma_i^2)\right)\\ &= \sum_i^N \log\mathrm{pdf}_G(x_i;\mu_i,\sigma_i^2)\end{aligned}$$

We can compute the derivative of the log-likelihood. Given that

$$\ell(\mu_i, \sigma_i^2; D) = -N \log \sqrt{w\pi\sigma_i^2} + \sum_{j=1}^{N} -\frac{(x_i[j]-\mu_i)^2}{2\sigma_i^2}$$

then

$$\frac{\partial \ell}{\partial \sigma_i^2} = -N \frac{(2\pi)/2}{\left(\sqrt{2\pi\sigma_i^2}\right)^2} + \sum_{j=1}^{N} \frac{2}{(x_i[j]-\mu_i)}(2\sigma_i^2)^2 = 0$$

$$-N\frac{1}{2\sigma_i^2} + 2\frac{2}{(2\sigma_i^2)^2}\sum_{j=1}^{N}(x_i[j]-\mu_i)^2 = 0$$

which implies

$$\sigma_i^2 - \sum_{j}^{N}(x_i[j]-\mu_i)^2$$

which is the expression of the sample variance. The same happens when this process is applied to the expectation μ_i.

Consider the following problem, where $X_1, \ldots, X_K$ are continuous random variables and the parents of a continuous random variable Y.

INSERTAR DIBUJO

The linear gaussian model

$$\begin{aligned} P(Y|X_1, \ldots, X_K) &\sim \mathcal{N}(\beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k,) \\ &= \mathcal{N}(\beta^T \mathbf{x}^T, \sigma^2) \end{aligned}$$

Then, we could compute $P(X_{t+1}|X_t, v_t)$

$$P(X_{t+1}|X_t, v_t) = \mathcal{N}(X_t + v_t, \sigma^2)$$