Statistical Data Analysis 2

Mateusz Kapusta

27 października 2022

1 Start

Grading rules

- 50% exam (test as usuall)
- 15% and 15% for two labolatory projects
- 15% midterm test
- 5% lab activity

Pass with 50% as usual.

2 Lets gooooo

First some formulas. We denote joint probability of $mathcalD$ and Y $P(X, Y)$. As we know:

$$P(\mathcal{D}) = \sum_{Y} P(X, Y) \tag{1}$$

. Then we have conditional probability

$$P(\mathcal{D}|Y) = \frac{P(X, Y)}{P(Y)} \tag{2}$$

Theeeen

$$P(\mathcal{D}|Y) = P(Y|X)\frac{P(X)}{P(Y)} \tag{3}$$

aka Bayes theorem.

2.1 Statistical inference

Let try to reanalyse coin toss experiment. Let's assume that we have θ as our propability of heads. When we act in frequentist approach we want to estimate parameter θ using methodes as MLE. What we need to claryify we belive there is God's rule that there exist one and only θ value. In Bayesian framework we do not think aobut tru parameter but rather conditional probability $P(\theta|mathcalD)$ (X is observed data). Lets introduce

$$L(\theta) = P(mathcalD|\theta) \tag{4}$$

As we know

$$P(\mathcal{D}|\theta) = \theta^k(1-\theta)^{N-k} \tag{5}$$

When we have N tosses and k sucesses. We can of course use ML estimator and will in the limit converge. We take logliklihood

$$l(\theta) = k\log\theta + (N-k)\log(1-\theta) \tag{6}$$

In ML approach we know that $\hat{\theta} = \frac{k}{N}$ but we want more then point estimator! In frequentist approach we know we can use something like repeat data generating process but we can clearly see that this approach isn't going to be very usefull. There is better way, we can use propability to obtain k times sucess.

$$P(\hat{\theta}) = \theta^{\hat{\theta}N}(1-\theta)^{N(1-\hat{\theta})}\binom{N}{\hat{\theta}N} \tag{7}$$

So we know that we can in fact obtain prob denisty for $\hat{\theta}$. It is possible to conduct analysis of propability when we do something like bootstrap. In bayesian statistics we think about prior. It is purly our belif about the propability of parameter. Lets decompose 3.

- $P(\theta|\mathcal{D})$ - posterior
- $P(\mathcal{D}|\theta)$ - likelihood
- $P(\theta)$ - prior
- $P(\mathcal{D})$ - constant

Sometimes life is hard (and we need to use MCMC), sometimes is easy (if we choose easy prior known and conjugate prior) so choose wisely. One of the conjugate priors is beta distribution (for binomial likelihood).

$$\mathcal{B}(\theta|\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)+\Gamma(\beta)}\theta^{\alpha-1}(1-\theta)^{\beta-1} \tag{8}$$

This distribution is defined on $[0,1]$ and if $\beta = 1$ and $\alpha = 1$ we get uniform prior (in other cases it looks very diffrent so we can choose something that suits us). We also have mean of the distribution at $\mu = \frac{\alpha}{\beta+\alpha}$ so making $\alpha >> \beta$ makes distribution shifted to the right. Lets rewind

$$P(\mathcal{D}|\theta) = \binom{N}{k}\theta^k(1-\theta)^{N-k} \tag{9}$$

When we use Beta distribution as prior we get posterior

$$P(\theta|\mathcal{D}) = \mathcal{B}(\theta|k+\alpha, N-k+\beta) \tag{10}$$

so we know that posterior is very nice. Now lets introduce some new point estimators

- MAP (Maximum aposteriori estimate) $\theta_{MAP} = argmax_\theta P(\theta|\mathcal{D})$
- ML (Maximum likelihood estimate) $\theta_{ML} = argmax_\theta P(\mathcal{D}|\theta)$

In the limit of the number of samples both estimators converge to the same value due to the fact that prior is dominativ. With know data prior dominates and things differ.

3 Bayesian networks

Bayesian network consist of

- directed acyclic graph (DAG) $G = (V, E)$
- local propability distribution, one for each vertex

Propabilit distribution for joint values $X = (X_1, \ldots, X_l)$ is

$$P(X) = \Pi_i P(X_i|pa(X_i)) \tag{11}$$

so we just multiply over conditional propabilities between node and it's parrent. We can consider something like linear gaussian model so we have

$$P(X_n|pa(X_n)) = Norm(b_n + \omega_n^t X_{pa(n)}, va_n) \tag{12}$$

so each vertex is charaterized by two values, mean and covariance matrix.

3.1 Markov Blancket

Markov blancket is subset of Bayesian is set of parents, coparents and children for given vertex. It is usefull as it contains all informations to calculate conditional propability of value of given vertex. We can say, that $P(V|all) = P(V|MB(V))$. Simplifing things we can say that

$$P(V|all) = P(V|MB(V)) = \frac{\sum P(V|parents) + P(children|V, coparents)}{Normalization} \tag{13}$$

So it is important when we are sampling using MCMC.

3.2 Conditional independence

We say that A and B are conditional independent given C and wreite $A \perp B|C$. If A and B are children of C then we have conditional independence, the same when there is connection between A and B through C but not when C is child of A and B. In fact, when A and B meet head to head we have independence but not conditional independence. Now let write some more definitions

- let A and B amd z is set of other nodes
- If the arrows between our points meet head to head and head to tail path is blocked.
- If the arrows meet at node C heat to head and nether the node nor it's desendence belong to z (no "explaining" away)

4 Time to learn

We can decompose learing process to parameter learning (modeling parameters of graph) and structure learning (determining structure of network). We can know structure of graph, know all data or any

combination.

4.1 Known graph, known data

Well, we can just use MAP estimation and we are fine. We treat given parameter as random varible (hidden), assume we have some hyperparameters, we take some prior and then just use MAP. In discrete case we use something like binomial distribution. In case of many posisble outcomes we can use multinomial distribution (with conjugate prior in form of dirichlet distrubution).

4.2 Known data, unknown graph

Well, just want to know the model, we can write

$$P(G|D) \propto P(D|G)P(G) = \int P(D|G,\theta)P(\theta|G)P(G)\mathrm{d}\theta \tag{14}$$

Its bad but we can sample. One important thing - Bayes factor $\frac{p(\mathcal{D}|M_i)}{P(\mathcal{D}|M_j)}$. It is used to decide which model is better.

4.3 Markov chains as examples of bayesian networks

Lets think of the chain that is composed of N parts X_i. Because we have chain X_i is dependent only on X_{i-1}. When we have Markov property

$$P(X_i|X_{i-1}) = P(X_2|X_1) \tag{15}$$

. As always when dealing with markov chains we have also classical transition matrix T. Example: CpG islands. We considere DNA sequence. Sometimes CG pars will be more often then others what is caled CpG island. When we have supervised learing and somebody gave us some data labeld as cpg or not cpg ilands we can imagine, that we have two models for two cases labeld T^+, T^-. We then can use something like log-odds

$$S(X) = \log \frac{P(X|T^+)}{P(X|T^-)} \tag{16}$$

. We can set a threshold for clasification and we have a classificator!

5 (Gaussian) Mixture Models

Lets get into, we have thata, that is clustered (n components, d dimensional data). Each compenet have weights π_i and are assumed to represent propability of sampling the obejct from given subtype. We have hidden random varible z_i and we assume

$$Z_j \sim Mult(\pi, 1)$$
$$x_i|z_i \sim \mathcal{N}(\mu_z, \Sigma_z)$$

. **We do not observe to which cluster observation belongs!** In case of clustering we just want to infer what is z_i, we can do it using what we previosly introduced so

$$P(z_i = j|x_i) \propto \pi_i \phi(P(x_i|\mu_j, \Sigma_j)) \tag{17}$$

We can also imagine, that we do not know π_j or parameters of gauss distributions. In this case we can maximize log-likelihood of our date with respoect to parameters. Well, in case when we have only one distribution it is easy, if not we are essentialy screwd as there is no ansewr in closed form. We need to know z_i to estimate parameters if we want to work in case $n > 1$ as we can introduce something like complete log-likelihood (we know not only x_i but also z_i). But there is also one more way...

5.1 Expectetion Maximalization

This is what we can do

- Initalize our varibles, z_i and μ_j, Σ_j
- Alternate until convergence
  - Compute soft class membersips given current parameters (soft means distribution of propabilities, not point estimate)
  - Use point estimates to determine cluster membership, estimate π_i and then compute μ_j Σ_j using techniques we previosly mentioned.

What we see, that we take some values, perform easy task that we described previosly. We always aproximate and then itarate. It can be shown that this algorithm converges but only to local minimum.

Time for some cool calculations, there is hidden varible Z, observed varibles X and parameters of Z and X λ and θ respectively:

$$l_{obs}(\theta, \lambda) = \log \sum_{Z} P(X, Z|\theta, \lambda) \tag{18}$$

$$= \log \sum_{Z} q(z) \frac{P(X, Z|\theta, \lambda)}{q(z)} \tag{19}$$

$$= \log \tag{20}$$

5.2 Kullback-Leibler divergence

The KL divergence measures distance between tho propability distributions $Q(X)$,$P(X)$.

$$D_{KL}(P||X) = -\sum_{X} P(X) \log \frac{Q(X)}{P(X)} \tag{21}$$

. This is always nonnegativ and equal to 0 only, if $Q(X) = P(X)$. what is important this is not symetrical! After some math we obtain, that

$$\updownarrow(\theta) = F(q, \theta) + D_{KL}(q||P) \tag{22}$$

.