Statistical Language Models
2019
Week 5 part 1

Dr. Dave Campbell
davecampbell@math.carleton.ca

Approximate Course Outline

- Week 1: ShinyApps and Dashboarding
- Week 2: TidyText & obtaining data, dealing with time events
- Week 3: Regular Expressions; Word co-occurrence explorations
- Week 4: Sentiment Analysis; Stochastic process models
- Week 5: Exponential models for time between events.
- Week 6: Bayesian Basics; Author attribution models; hierarchical models
- Week 7: MCMC Diagnostics
- Week 8: Embeddings and Word2Vec; Cryptography
- Week 9: Clustering; Latent Dirichlet Allocation and topic models.
- Week 10: Variational Inference
- Week 11: Getting Fancier with Language Models
- Week 12: Student projects and presentations

Cervical Cancer Example

- N = 5736
- Observed cancer: Y = 1
- $P(\theta) \propto \theta^{\alpha-1}(1-\theta)^{\beta-1}$, this is a Beta(α,β) distribution, α,β are equivalent to the prior number of observed cancer and non-cancers
- Using the non-vaccinated group we may choose α = 36, β=5766

Cervical Cancer Example

- N = 5736
- Observed cancer: Y = 1
- $P(\theta) \propto \theta^{\alpha-1}(1-\theta)^{\beta-1}$, α,β are equivalent to the prior number of observed cancer and non-cancers. Here α = 36,β = 5766
- $P(Y \mid n, \theta) \propto \theta^{y}(1-\theta)^{n-y}$ this is a Binomial(n,θ)
- $P(\theta \mid Y, n) \propto P(Y \mid n, \theta)P(\theta) = \theta^{y+\alpha-1}(1-\theta)^{n-y+\beta-1}$ this is Beta(α+y,β+n-y)

MCMC

- Create a stochastic process X(t) such that its limiting probabilities match our target distribution.

- If the chain is irreducible, aperiodic, and not transient then a sample from Metropolis Hastings is guaranteed converge to the appropriate limiting probabilities.

Metropolis Hastings

- Start with X(t-1) = j
- Propose a value Y(t) | X(t-1)=j from transition probability matrix Q as a candidate for X(t)
- compute $\alpha_{ij} = min\left(\frac{P(Y)P_{ji}}{P(X[t]P_{ij}}, 1\right)$ and sample u~Unif(0,1)
- If u< α_{ij} then accept the proposal and set Xt=Y and if not then set X(t)=X(t-1).
- Repeat (T times) until you obtain a sufficient sample from the distribution of X

```
Niters = 1000000
samples = c(1/5736,  rep(0,1,Niters))
y = 1; N = 5736
a = 36 ; b = 5766
stepvar = .004
for(lp in 1:(Niters-1)){
       x = runif(1,samples[lp]-stepvar,samples[lp]+stepvar)                    # sample from arbitrary transition distribution
       if(x>0 & x<1){
            alpha =  dbinom(y,N,x)                *(x^(a-1))              *(1-x)^(b-1) /
                          (dbinom(y,N,samples[lp])*(samples[lp]^(a-1))*(1-samples[lp])^(b-1))
            u = runif(1)
                if(u<alpha){
                   samples[lp+1] = x
                }else{
                   samples[lp+1] = samples[lp]
                }
       }
         else{
            samples[lp+1] = samples[lp]
          }
}

hist(samples,probability=TRUE,500)
x = seq(0,.01,length=1000)
lines(x,dbeta(x, a+y, b+N-y ),lwd=4,col="red")
```

Language Model

- Probability Transition Matrix from "AN" to "NA", "NB",…"N_"
- Each row is a Dirichlet (sums to 1, 27 valid categories)
- $$P(\theta) = \frac{\Gamma(\alpha_A + \ldots + \alpha\_)}{\Gamma(\alpha_A) \ldots \Gamma(\alpha\_)} \theta_A^{\alpha_A - 1} \ldots \theta_{\_}^{\alpha\_ - 1}$$

- Update the language model using books.
- Data generating process (likelihood): P(book | θ)
- P(observed transitions from "AN" = [n_A,…n_]|θ) = multinomial for each transition
- $P(\text{observed transitions from AN} = [n_A, \ldots n\_] \mid \theta) \propto \theta_{AN,NA}^{n_A} \ldots \theta_{AN,N\_}^{n\_}$

Now the cypher

- Cypher ϕ, language model θ, and (deterministic) decoded message Y(ϕ)
- We can directly calculate P(Y(ϕ) | θ) by deciphering the message and summing over probabilities from transitions:
  $$P(Y(\phi) \mid \theta) = \prod_t P(Y_t(\phi) \mid \theta)$$
- Sample directly from P(Y(ϕ) | θ) using MCMC

- End result: a sample from the data generating mechanism $P(Y(\phi) \mid \theta)$ using MCMC

Author Attribution Models

- P(Author | text) = P(text | Author) P(Author) / P(text)

- Requires P(Author) to be feasible
- Possible models for P(text | Author)?