Sampling for estimation

Idea: Parameter estimation via Simulation

$$y = \beta_0 + \beta_1 x + \epsilon \qquad \epsilon \sim N(0, \sigma^2)$$

$$y \sim N(\beta_0 + \beta_1 x, \sigma_\epsilon^2)$$ Assume known.

$$\beta \sim N(\vec{\emptyset}, \vec{\sigma}^2 \underline{I})$$ Prior distribution over β

Suppose we observe data $D = \{x, y\}^N$

$$P(\beta | D) = ?$$

$$= \frac{P(D|\beta)\, P(\beta)}{P(D)}$$ Bayes!

Note that D is observed

$$P(D|\beta, \sigma_\epsilon^2) = \prod_i P(y_i | x_i, \beta, \sigma_\epsilon^2)$$

$$= \prod_i N(y_i | \beta_0 + \beta_1 x_i, \sigma_\epsilon^2)$$

$$P(\beta) = N(\beta | \vec{\emptyset}, \vec{\sigma}^2 I)$$

Here is a dumb algorithm to estimate β.

For T steps

Sample a $\hat{\beta}_t$ somehow

$w_t \leftarrow P(\hat{\beta}_t \mid D)$

$Z = \sum_t w_t$

$\hat{\beta}^* \leftarrow \sum_t \frac{w_t \hat{\beta}_t}{Z}$

weighted average over $\hat{\beta}$'s

A type of Importance Sampling. We use weighted samples to approx. a posterior over β. Weights here are unnormalized: hence Z.

Note: Can use same strategy to <u>predict</u> $\hat{y}_i$. How?

<u>More</u> <u>generally</u>: Assume a data generating distribution f w. parameters θ, and a prior π.

$$\underbrace{\int f(\theta)\pi(\theta)\, d\theta}_{\text{might be intractable}} \approx \underbrace{\frac{1}{S}\sum_{s=1}^{S} f(\theta^s) \qquad \theta^s \sim \pi}_{\text{approximate by sampling}}$$

"Monte Carlo" Integration

$$\frac{1}{S}\sum_{s=1}^{S} \frac{w^s}{Z}\theta^s \quad \text{where} \quad Z = \sum_{s=1}^{S} w^s$$

and w^s is an <u>unnormalized</u> joint $P(D, \theta^s)$.

Obviously, π is critical here.

The key trick:

$$P(\theta|D) = \frac{P(D|\theta)\,P(\theta)}{P(D)} \propto P(D,\theta)$$

[See Jupyter Notebook]

For more complex cases, simple importance weighting is not going to fly. It would take forever to find good θ.

Can we be smarter about picking θ^s?

<u>Markov Chain Monte Carlo</u> (MCMC) is a method that tries to simulate draws from a dist. of interest.

$$P(\theta^{(s+1)} \mid \theta^1 \dots \theta^s) = \underbrace{T(\theta^{(s+1)} \leftarrow \theta^s)}_{\text{Transition probability from current parameters.}}$$

Metropolis-Hastings is a particular version of MCMC. Basically, start somewhere θ^0 then:

Make a proposal θ^{t+1} that you accept or reject with some probability.

The accept p should be high if it is a better fit.

Gibbs Sampling is a simple recipe where we update a particular parameter θ_j conditioned on all others.

Gibbs-Sample

Initialize $\theta^0 \sim q()$

for T steps

$$\theta_1^{\tau} \sim P(\theta_1 \mid \theta_2^{(\tau-1)}, \theta_3^{(\tau-1)}, \ldots, \theta_m^{(\tau-1)})$$

$$\theta_2^{\tau} \sim P(\theta_2 \mid \theta_1^{(\tau-1)}, \theta_2^{(\tau-1)}, \ldots \theta_m^{(\tau-1)})$$

...

$$\theta_m^{\tau} \sim P(\theta_m \mid \theta_1^{(\tau-1)}, \theta_2^{(\tau-1)}, \ldots \theta_{m-1}^{(\tau-1)})$$

Return θ^{τ}

(Content below derived from Jordan Boyd-Graber)

Coming back to LDA

$$\left.\begin{aligned}
\beta_k &\sim \text{Dirichlet}(\eta) \\
\theta_d &\sim \text{Dirichlet}(\alpha) \\
z_{d,n} &\sim \text{Discrete}(\theta_d) \\
w_{d,n} \mid z_{d,n} &\sim \text{Discrete}(\beta_{z_{d,n}})
\end{aligned}\right\} \text{the model}$$

For LDA, we will estimate the probability of a specific word's topic assignment, conditioned on all other assignments.

$$P(z_{d,n} = k \mid \underbrace{\vec{z}_{-d,n}}_{\text{all other word topic assignments}}, \vec{w}, \alpha, \lambda)$$

$$= \frac{P(z_{d,n} = k, \vec{z}_{-d,n} \mid \vec{w}, \alpha, \lambda)}{P(\vec{z}_{-d,n} \mid \vec{w}, \alpha, \lambda)}$$

Requires integrating out Θ and β. This is a bit hairy, but we end with:

$$= \underbrace{\frac{n_{d,k} + \alpha_k}{\sum_i^K n_{d,i} + \alpha_i}}_{\text{How much this doc "likes" this topic}} \cdot \underbrace{\frac{V_{k,w_{d,n}} + \eta_{w_{d,n}}}{\sum_{w'} V_{k,w'} + \eta_{w'}}}_{\text{How much topic } k \text{ "likes" this word}}$$

Count of topic k in doc d

Count of topic k using word $w_{d,n}$

Note! Given $P(z_{d,n})$ for all $z_{d,n}$, we can derive Θ, β