Bayesian Concept Learning

Generative model: fully probabilistic model of all variables

Discriminative model: model only for target variables conditioned on observation

- Posterior = likelihood times ~~prior~~ prior normalized by data/evidence
- When data enough → posterior peaks on single data/concept: **MAP estimate**
- With more data → MAP converges to MLE, data overwhelms the prior
- Small or ambiguous dataset → plug-in approximation

Beta Binomial Model

"inferring probability coin shows heads, given series of observed tosses"

- **Likelihood** $P(D|\theta) = \theta^{N_1}(1-\theta)^{N_0}$, $N_1 = \sum(x=1)$ $N_0 = \sum(x=0)$, sufficient statistics

"Binomial P follows Beta distribution"

- **Prior**: conjugate prior when has same form as likelihood

$$\text{Beta}(\theta|a,b) \propto \theta^{a-1}(1-\theta)^{b-1}$$

prior parameters = hyperparameters: encode prior beliefs

- **Posterior**: $P(\theta|D) \propto \text{Bin}(N_1|\theta, N_0+N_1) \cdot \text{Beta}(\theta|a,b) \propto \text{Beta}(\theta|N_1+a, N_0+b)$

  Batch update = sequential update

  - Mean: $\dfrac{a+N_1}{a+b+N}$
  - Variance: $\dfrac{\hat{\theta}(1-\hat{\theta})}{N}$

- **Posterior predictive**: $P(x=1|D) = \int_0^1 P(x=1|\theta)P(\theta|D)\,d\theta = \int_0^1 \theta \,\text{Beta}(\theta|a,b)\,d\theta = E[\theta|D] = \dfrac{a}{a+b}$

  predict future observations

  - Equiv to plug-in posterior mean parameters

- Using MLE is poor when sample count is small: zero count / sparse data problem

  ↳ **Add one smoothing**: add 1 to counts

Dirichlet Multinomial

- Discrete multivariate variable

- **Likelihood** $P(D|\theta) = \prod^{n} \theta_n^{N_n}$
- **Prior**: $\text{Dir}(\theta|\alpha) = \dfrac{1}{B(\alpha)} \prod^{n} \theta_n^{\alpha_n - 1} \cdot \mathbb{I}(x \in S_n)$
- **Posterior**: $P(\theta|D) = \text{Dir}(\theta|\alpha_1+N_1, \dots, \alpha_n+N_n)$ $\left(\prod^{n} \theta^{\alpha_n+N_n-1}\right)$
- **Posterior predictive**: $P(x=j|D) = \dfrac{\alpha_j+N_j}{\alpha_0+N}$
- **Application**: language modeling | bag of words

General Generative Classifier

$$P(y=c|x,\theta) = \frac{P(y=c|\theta)P(x|y=c,\theta)}{\sum_{c'} P(y=c'|\theta)P(x|y=c',\theta)}$$

Generate data using class-conditional density $p(x|y=c)$ and class prior $p(y=c)$