CS540 Machine learning
L8

Announcements

- Linear algebra tutorial by Mark Schmidt, 5:30 to 6:30 pm today, in the CS X-wing 8th floor lounge (X836).
- Move midterm from Tue Oct 14 to Thu Oct 16?
- Hw3sol handed out today
- Change in order

Last time

- Multivariate Gaussians
- Eigenanalysis
- MLE
- Use in generative classifiers

This time

- Naïve Bayes classifiers
- Bayesian parameter estimation I: Beta-Binomial model

Bayes rule for classifiers

Class-conditional density Class prior

Class posterior

$$p(y = c|x) = \frac{p(x|y = c)p(y = c)}{\sum_{c'} p(x|y = c')p(y = c')}$$

Normalization constant

Class prior

- Let $(Y_1,..,Y_C) \sim \text{Mult}(\pi, 1)$ be the class prior.

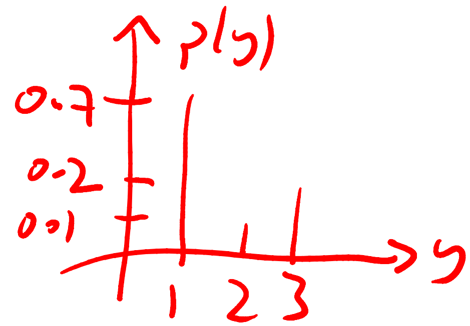
$$P(y_1, \ldots, y_C | \pi) = \prod_{c=1}^{C} \pi_c^{I(y_c=1)} \qquad \sum_{c=1}^{C} \pi_c = 1$$

- Since $\sum_c Y_c=1$, only one bit can be on. This is called a 1-of-C encoding. We can write Y=c instead.

$$Y=2 \equiv (Y_1,Y_2,Y_3) = (0,1,0)$$

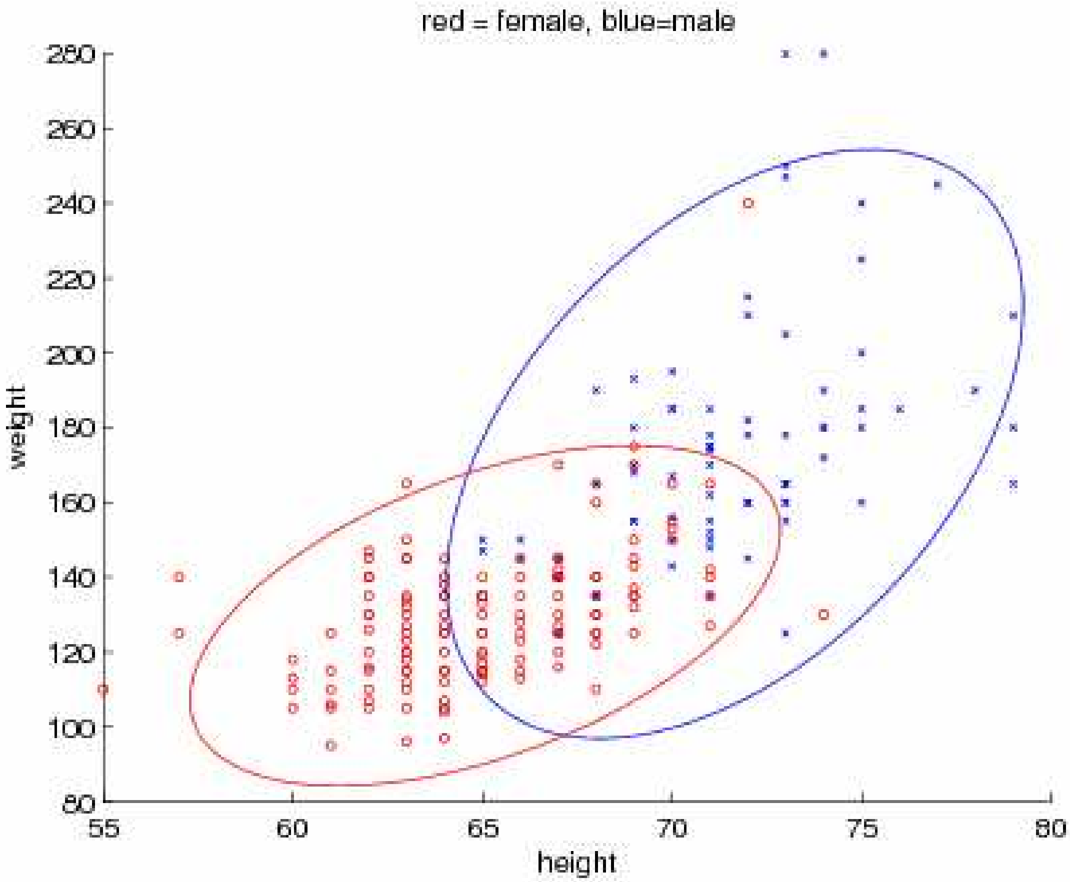
$$P(y|\pi) = \prod_{c=1}^{C} \pi_c^{I(y=c)} = \pi_y$$

- e.g., p(man)=0.7, p(woman)=0.1, p(child)=0.2



Correlated features

- Height and weight are not independent



Fitting the model

- Fit each class conditional density separately

$$
\begin{aligned}
\boldsymbol{\mu}_c &= \frac{1}{n_c}\sum_{i=1}^{n} I(y_i = c)\mathbf{x}_i = \frac{1}{n_c}\sum_{i:y_i=c}\mathbf{x}_i \\
\boldsymbol{\Sigma}_c &= \frac{1}{n_c}\sum_{i=1}^{n} I(y_i = c)(\mathbf{x}_i - \boldsymbol{\mu}_c)(\mathbf{x}_i - \boldsymbol{\mu}_c)^T \\
n_c &= \sum_{i=1}^{n} I(y_i = c)
\end{aligned}
$$

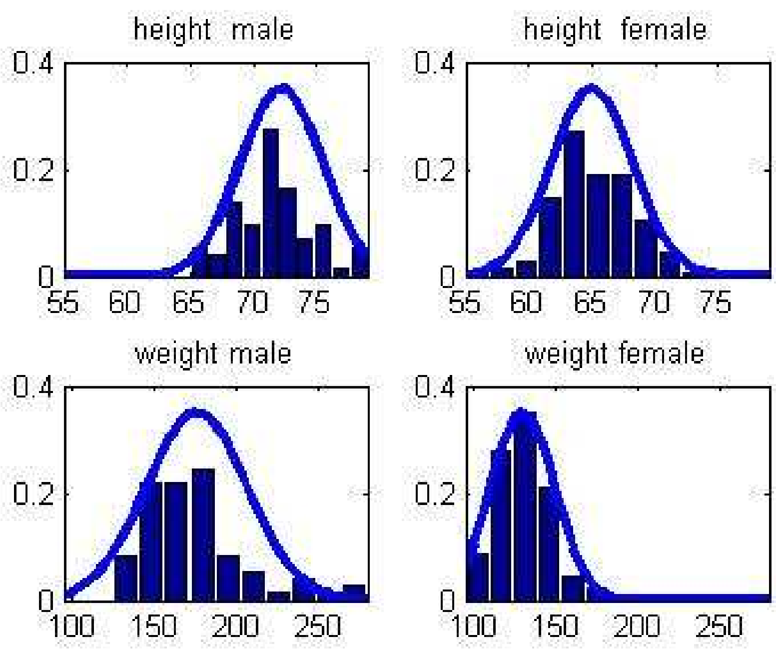
$$
\boldsymbol{\pi}_c = \frac{n_c}{n}
$$

Ignoring the correlation...

- If $X_j \in R$, we can use product of 1d Gaussians

  $X_j | y=c \sim N(\mu_{jc}, \sigma_{jc})$

$$p(x|y=c) = \prod_{j=1}^{d} \frac{1}{\sqrt{2\pi\sigma_{jc}^2}} \exp(-\frac{1}{2\sigma_{jc}^2}(x_j - \mu_{jc})^2)$$



$$\mathbf{\Sigma}_c = \begin{pmatrix} \sigma_{1c}^2 & \dots & 0 \\ & \ddots & \\ 0 & \dots & \sigma_{dc}^2 \end{pmatrix}$$

Document classification

- Let $Y \in \{1,\ldots,C\}$ be the class label and $x \in \{0,1\}^d$
- eg $Y \in$ {spam, urgent, normal},

  x_i = I(word i is present in message)
- Bag of words model

1 2 3 4 5 6 7
Words = {john, mary, sex, money, send, meeting, unk}

"John sent money to Mary after the meeting about money"

↓ Stop word removal

"john sent money mary after meeting about money"

↓ Tokenization

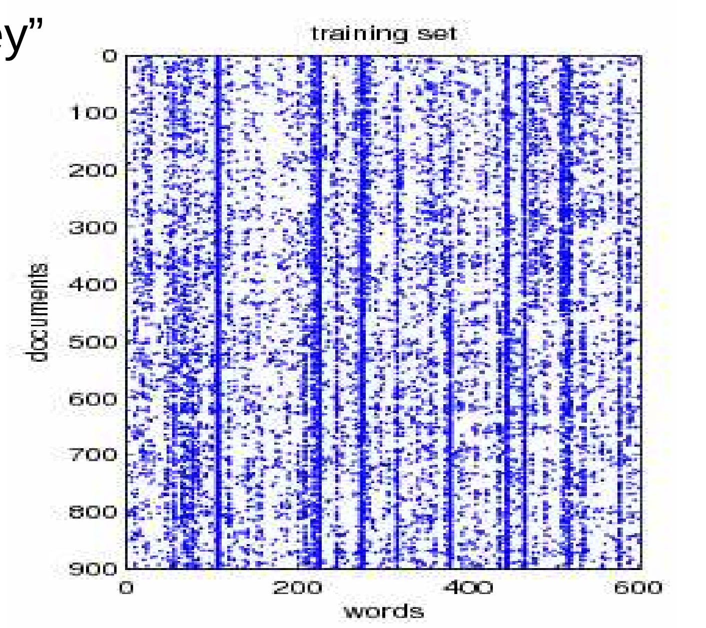
1 7 4 2 7 6 7 4

↓ Word counting

[1, 1, 0, 2, 0, 1]

↓ Thresholding (binarization)

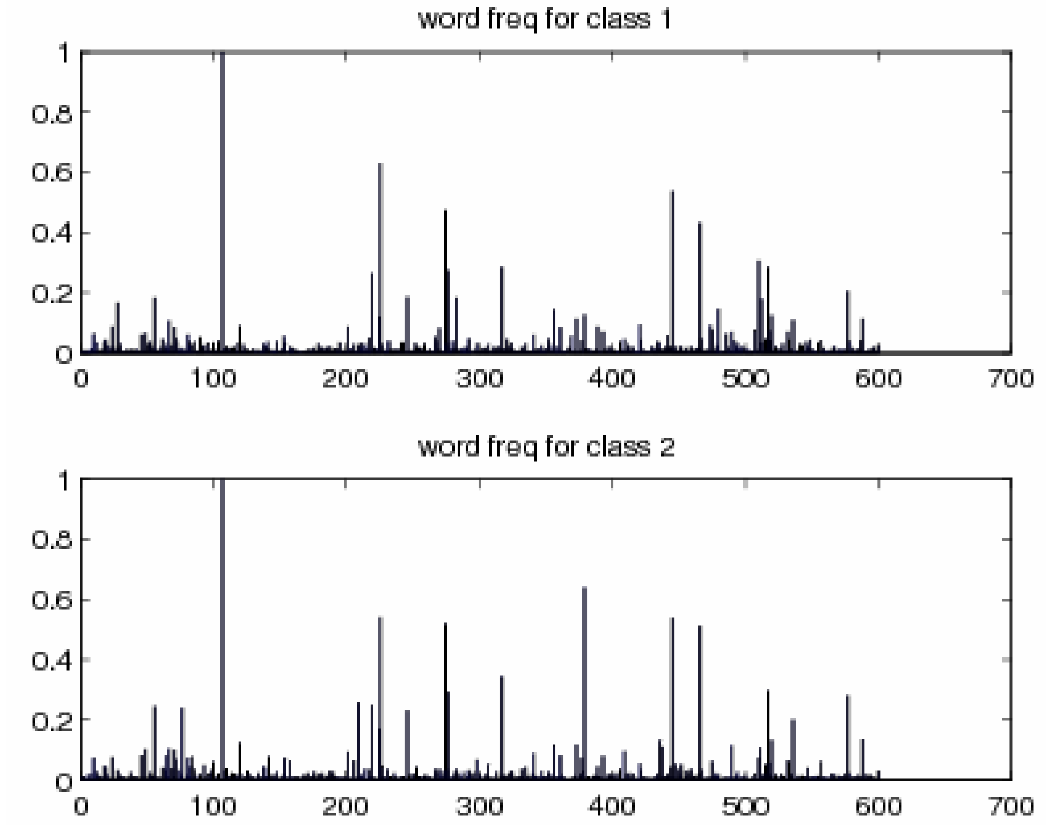
[1, 1, 0, 1, 0, 1]



Binary features (multivariate Bernoulli)

- Let $X_i|y=c \sim Ber(\mu_{ic})$ so $p(X_i=1|y=c) = \mu_{ic}$

$$p(\mathbf{x}|y=c,\boldsymbol{\mu}) = \prod_{j=1}^{d} \mu_{jc}^{I(x_j=1)} (1-\mu_{jc})^{I(x_j=0)}$$



Fitting the model

$$
\begin{aligned}
\mu_{jc} &= \frac{1}{n_c} \sum_{i=1}^{n} I(y_i = c) I(x_{ij} = 1) = \frac{n_{jc}}{n_c} \\
n_{jc} &= \sum_{i=1}^{n} I(y_i = c, x_{ij} = 1)
\end{aligned}
$$

Class posterior

- Bayes rule

$$p(y=c|x) = \frac{p(y=c)p(x|y=c)}{p(x)} = \frac{\pi_c \overbrace{\prod_{i=1}^{d} \theta_{ic}^{I(x_i=1)}(1-\theta_{ic})^{I(x_i=0)}}^{f_c}}{p(x)}$$

- Since numerator and denominator are very small number, use logs to avoid underflow

$$\log p(y=c|x) = \log \pi_c + \sum_{i=1}^{d} I(x_i=1)\log\theta_{ic} + I(x_i=0)\log(1-\theta_{ic}) - \log p(x)$$

- How compute the normalization constant?

$$\log p(x) = \log[\sum_c p(y=c,x)] = \log[\sum_c \pi_c f_c]$$

Log-sum-exp trick

- Define

$$\begin{aligned}
\log p(x) &= \log[\sum_c \pi_c f_c] \\
b_c &= \log \pi_c + \log f_c \\
\log p(x) &= \log \sum_c e^{b_c} = \log \left[(\sum_c e^{b_c}) e^{-B} e^{B} \right] \\
&= \log \left[(\sum_c e^{b_c - B}) e^{B} \right] = \left[\log (\sum_c e^{b_c - B}) \right] + B \\
B &= \max_c b_c
\end{aligned}$$

$$\log(e^{-120} + e^{-121}) = \log\left(e^{-120}(e^{0} + e^{-1})\right) = \log(e^{0} + e^{-1}) - 120$$

- In Matlab, use Minka's function S = logsumexp(b)

```
logjoint = log(prior) + counts * log(theta) + (1-counts) * log(1-theta);
logpost = logjoint – logsumexp(logjoint)
```

$\log p(y=c, x)$

$\log p(y=c|x)$

Missing features

- Suppose the value of x_1 is unknown
- We can simply drop the term $p(x_1|y=c)$.

$$
\begin{aligned}
p(y=c|x_{2:d}) &\propto p(y=c, x_{2:d}) \\
&= \sum_{x_1} p(y=c, x_1, x_{2:d}) \\
&= \sum_{x_1} p(y=c) \prod_{j=1}^{d} p(x_j|y=c) \\
&= p(y=c)[\sum_{x_1} p(x_1|y=c)] \prod_{j=2}^{d} p(x_j|y=c) \\
&= p(y=c) \prod_{j=2}^{d} p(x_j|y=c)
\end{aligned}
$$

- This is a big advantage of generative classifiers over discriminative classifiers

Form of the class posterior

- We can derive an analytic expression for p(y=c|x) that will be useful later.

$$
\begin{aligned}
p(Y=c|x,\theta,\pi) &= \frac{p(x|y=c)p(y=c)}{\sum_{c'} p(x|y=c')p(y=c')} \\
&= \frac{\exp[\log p(x|y=c) + \log p(y=c)]}{\sum_{c'} \exp[\log p(x|y=c') + \log p(y=c')]} \\
&= \frac{\exp\left[\log \pi_c + \sum_i I(x_i=1)\log\theta_{ic} + I(x_i=0)\log(1-\theta_{ic})\right]}{\sum_{c'} \exp\left[\log \pi_{c'} + \sum_i I(x_i=1)\log\theta_{i,c'} + I(x_i=0)\log(1-\theta_{ic})\right]}
\end{aligned}
$$

Form of the class posterior

- From previous slide

$$p(Y=c|x,\theta,\pi) \quad \propto \quad \exp\left[\log \pi_c + \sum_i I(x_i=1)\log\theta_{ic} + I(x_i=0)\log(1-\theta_{ic})\right]$$

- Define

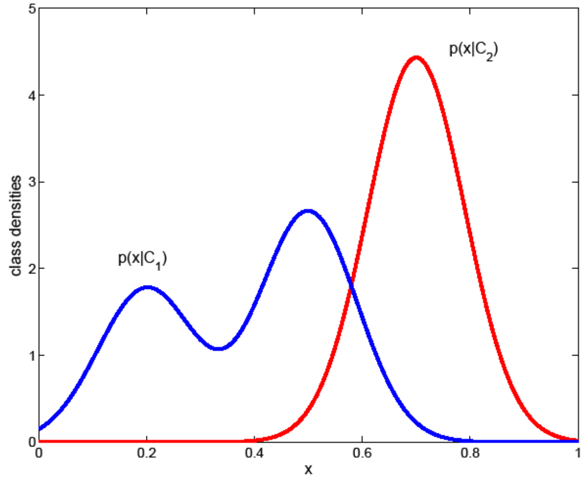
$$\begin{aligned} x' &= [1, I(x_1=1), I(x_1=0), \ldots, I(x_d=1), I(x_d=0)] \\ \beta_c &= [\log \pi_c, \log\theta_{1c}, \log(1-\theta_{1c}), \ldots, \log\theta_{dc}, \log(1-\theta_{dc})] \end{aligned}$$

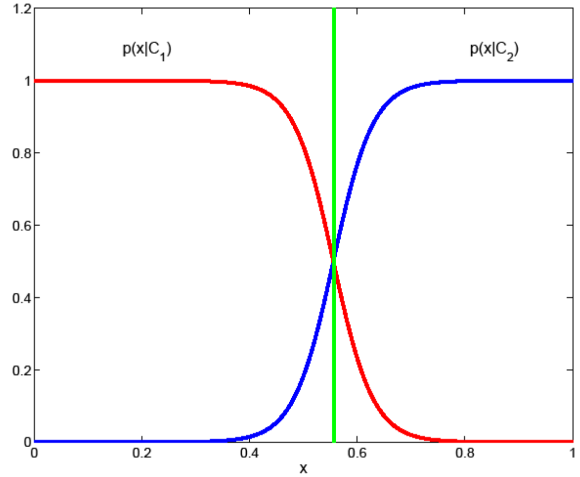
- Then the posterior is given by the softmax function

$$p(Y=c|x,\beta) = \frac{\exp[\beta_c^T x']}{\sum_{c'} \exp[\beta_{c'}^T x']}$$

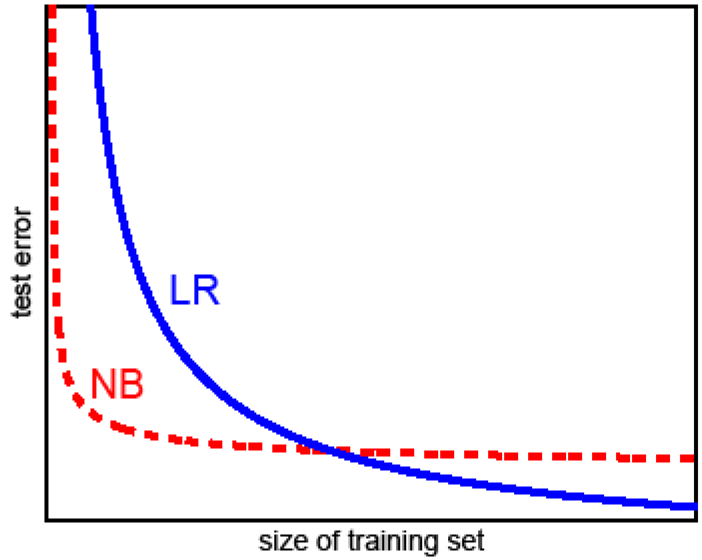
Discriminative vs generative

- Discriminative: p(y|x,theta)
- Generative: p(y,x|theta)





Logisitic regression vs naïve Bayes



	Discriminative	Generative
Easy to ﬁt?	No	Yes
Can handle basis function expansion?	Yes	No
Fit classes separately?	No	Yes
Handle missing data?	No	Yes
Best for	Large sample size	Small sample size

Sparse data problem

- Consider naïve Bayes for binary features.

	Spam	Ham
Limited	1	2
Time	10	9
Offer	0	0
Total	Ns	Nh

X = “you will receive our limited time offer if you send us $1M today”

$$p(\mathbf{x}|y = S) = (1/N_s)(10/N_s)(0/N_s) = 0$$

MLE overfits the data

Outline

- Bayes: what/why?
- Bernoulli

Fundamental principle of Bayesian statistics

- In Bayesian stats, everything that is uncertain (e.g., θ) is modeled with a probability distribution.
- We incorporate everything that is known (e.g., D) is by conditioning on it, using Bayes rule to update our prior beliefs into posterior beliefs.

Likelihood

Posterior probability

Prior probability

$$p(h \mid d) = \frac{p(d \mid h)\, p(h)}{\sum_{h' \in H} p(d \mid h')\, p(h')}$$

Bayesian inference = Inverse probability theory

In praise of Bayes

- Bayesian methods are conceptually simple and elegant, and can handle small sample sizes (e.g., one-shot learning) and complex hierarchical models without overfitting.
- They provide a single mechanism for answering all questions of interest; there is no need to choose between different estimators, hypothesis testing procedures, etc.
- They avoid various pathologies associated with orthodox statistics.
- They often enjoy good frequentist properties.

Why isn’t everyone a Bayesian?

- The need for a prior.
- Computational issues.

The need for a prior

- Bayes rule requires a prior, which is considered "subjective".
- However, we know learning without assumptions is impossible (no free lunch theorem).
- Often we actually have informative prior knowledge.
- If not, it is possible to create relatively "uninformative" priors to represent prior ignorance.
- We can also estimate our priors from data (*empirical Bayes).*
- We can use posterior predictive checks to test goodness of fit of both prior and likelihood.

Computational issues

- Computing the normalization constant requires integrating over all the parameters

$$p(\theta|D) = \frac{p(\theta)p(D|\theta)}{\int p(\theta')p(D|\theta')d\theta'}$$

- Computing posterior expectations requires integrating over all the parameters

$$Ef(\Theta) = \int f(\theta)p(\theta|D)d\theta$$

Approximate inference

- We can evaluate posterior expectations using Monte Carlo integration

$$Ef(\Theta) = \int f(\theta)p(\theta|D)d\theta \approx \frac{1}{N}\sum_{s=1}^{N} f(\theta^s) \quad \text{where } \theta^s \sim p(\theta|D)$$

- Generating posterior samples can be tricky
  - Importance sampling
  - Particle filtering
  - Markov chain Monte Carlo (MCMC)
- There are also deterministic approximation methods
  - Laplace
  - Variational Bayes
  - Expectation Propagation

Conjugate priors

- For simplicity, we will mostly focus on a special kind of prior which has nice mathematical properties.
- A prior $p(\theta)$ is said to be *conjugate* to a likelihood $p(D|\theta)$ if the corresponding posterior $p(\theta|D)$ has the same functional form as $p(\theta)$.
- This means the prior family is *closed under Bayesian updating.*
- So we can recursively apply the rule to update our beliefs as data streams in (online learning).
- A natural conjugate prior means $p(\theta)$ has the same functional form as $p(D|\theta)$.

Example: coin tossing

- Consider the problem of estimating the probability of heads θ from a sequence of N coin tosses, D = $(X_1, \ldots, X_N)$
- First we define the likelihood function, then the prior, then compute the posterior. We will also consider different ways to predict the future.
- MLE is

$$\hat{\theta} = \frac{N_1}{N}$$

- Suffers from sparse data problem

Black swan paradox

- Suppose we have seen N=3 white swans. What is the probability that swan X_{N+1} is black?
- If we plug in the MLE, we predict black swans are impossible, since $N_b=N_1=0$, $N_w=N_0=3$

$$\hat{\theta}_{MLE} = \frac{N_b}{N_b + N_w} = \frac{0}{N}, \;\; p(X = b|\hat{\theta}_{MLE}) = \hat{\theta}_{MLE} = 0$$

- However, this may just be due to sparse data.
- Below, we will see how Bayesian approaches work better in the small sample setting.

The beta-Bernoulli model

- Consider the probability of heads, given a sequence of N coin tosses, $X_1, \ldots, X_N$.
- Likelihood

$$p(D|\theta) = \prod_{n=1}^{N} \theta^{X_n} (1-\theta)^{1-X_n} = \theta^{N_1} (1-\theta)^{N_0}$$

- Natural conjugate prior is the Beta distribution

$$p(\theta) = Be(\theta|\alpha_1, \alpha_0) \propto \theta^{\alpha_1 - 1} (1-\theta)^{\alpha_0 - 1}$$

- Posterior is also Beta, with updated counts

$$p(\theta|D) = Be(\theta|\alpha_1 + N_1, \alpha_0 + N_0) \propto \theta^{\alpha_1 - 1 + N_1} (1-\theta)^{\alpha_0 - 1 + N_0}$$

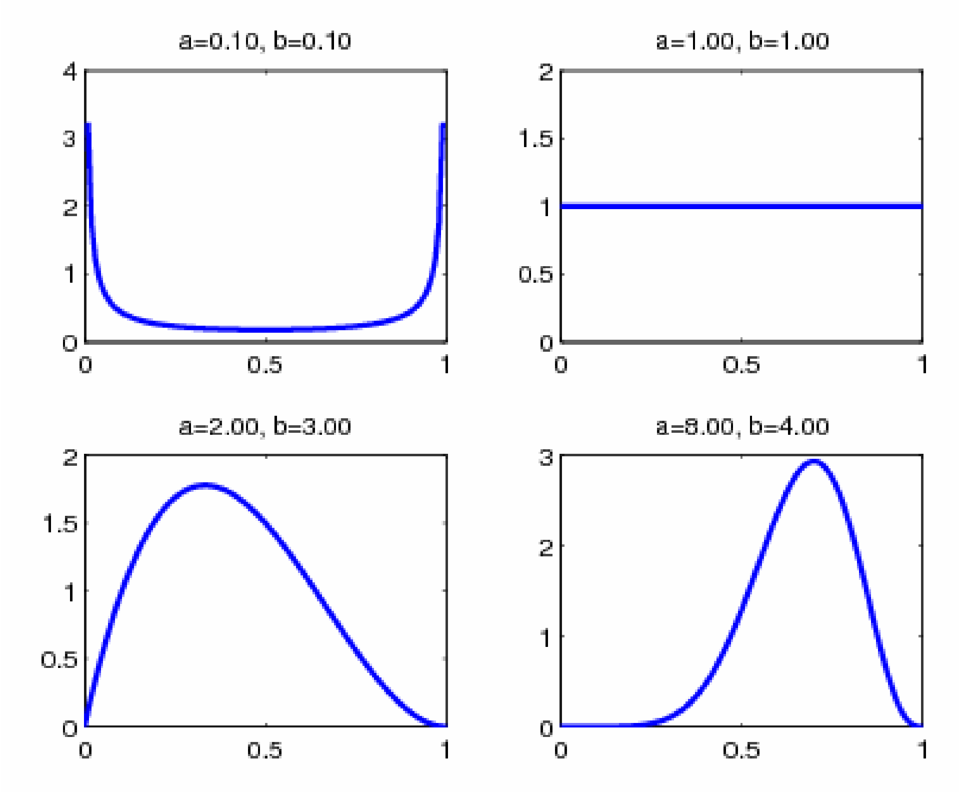
Just combine the exponents in θ and $(1-\theta)$ from the prior and likelihood

The beta distribution

- Beta distribution $p(\theta|\alpha_1, \alpha_0) = \frac{1}{B(\alpha_1, \alpha_0)} \theta^{\alpha_1 - 1}(1-\theta)^{\alpha_0 - 1}$
- The normalization constant is the beta function

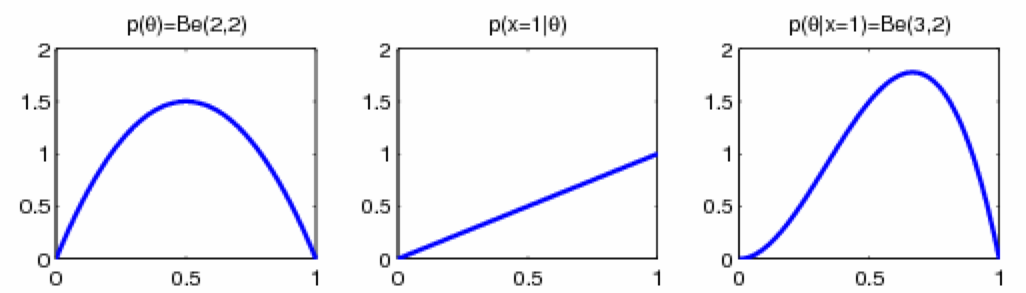
$$B(\alpha_1, \alpha_0) = \int_0^1 \theta^{\alpha_1 - 1}(1-\theta)^{\alpha_0 - 1} d\theta = \frac{\Gamma(\alpha_1)\Gamma(\alpha_0)}{\Gamma(\alpha_1 + \alpha_0)}$$

$$E[\theta] = \frac{\alpha_1}{\alpha_1 + \alpha_0}$$

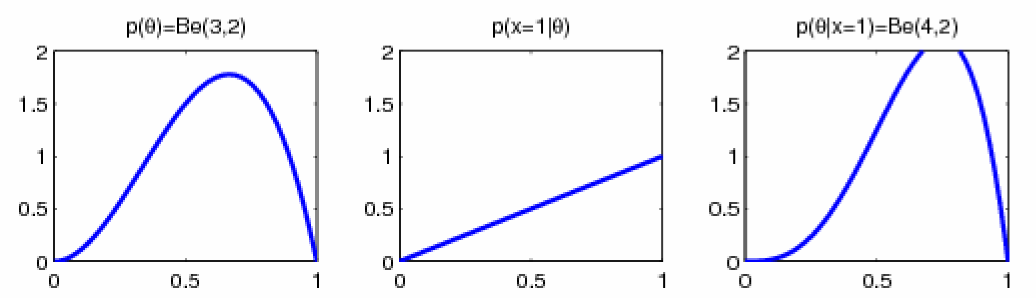


Updating a beta distribution

- Prior is Beta(2,2). Observe 1 head. Posterior is Beta(3,2), so mean shifts from 2/4 to 3/5.



- Prior is Beta(3,2). Observe 1 head. Posterior is Beta(4,2), so mean shifts from 3/5 to 4/6.



Setting the hyper-parameters

- The prior *hyper-parameters* α_1, α_0 can be interpreted as *pseudo counts.*
- The *effective sample size* (strength) of the prior is $\alpha_1+\alpha_0$.
- The prior mean is $\alpha_1/(\alpha_1+\alpha_0)$.
- If our prior belief is p(heads) = 0.3, and we think this belief is equivalent to about 10 data points, we just solve

$$\alpha_1 + \alpha_0 = 10, \quad \frac{\alpha_1}{\alpha_1 + \alpha_0} = 0.3$$

Posterior mean

- Let $N=N_1 + N_0$ be the amount of data, and $M=\alpha_0+\alpha_1$ be the amount of virtual data.

The posterior mean is a convex combination of prior mean α_1/M and MLE N_1/N

$$
\begin{aligned}
E[\theta|\alpha_1, \alpha_0, N_1, N_0] &= \frac{\alpha_1 + N_1}{\alpha_1 + N_1 + \alpha_0 + N_0} = \frac{\alpha_1 + N_1}{N + M} \\
&= \frac{M}{N+M}\frac{\alpha_1}{M} + \frac{N}{N+M}\frac{N_1}{N} \\
&= w\frac{\alpha_1}{M} + (1-w)\frac{N_1}{N}
\end{aligned}
$$

$w = M/(N+M)$ is the strength of the prior relative to the total amount of data

We *shrink* our estimate away from the MLE towards the prior (a form of regularization).

MAP estimation

- It is often easier to compute the posterior mode (optimization) than the posterior mean (integration).
- This is called maximum a posteriori estimation.

$$\hat{\theta}_{MAP} = \arg\max_{\theta} p(\theta|D)$$

- This is equivalent to penalized likelihood estimation.

$$\hat{\theta}_{MAP} = \arg\max_{\theta} \log p(D|\theta) + \log p(\theta)$$

- For the beta distribution,

$$MAP = \frac{\alpha_1 - 1}{\alpha_1 + \alpha_0 - 2}$$

Posterior predictive distribution

- We integrate out our uncertainty about θ when predicting the future (hedge our bets)

$$p(X|D) \quad = \quad \int p(X|\theta)p(\theta|D)d\theta$$

- If the posterior becomes peaked

$$p(\theta|D) \rightarrow \delta(\theta - \hat{\theta})$$

we get the *plug-in principle.*

$$p(x|D) = \int p(x|\theta)\delta(\theta - \hat{\theta})d\theta = p(x|\hat{\theta})$$

Sifting property of delta functions

Posterior predictive distribution

- Let α_i' = updated hyper-parameters.
- In this case, the posterior predictive is equivalent to plugging in the posterior mean parameters

$$
\begin{aligned}
p(X=1|D) &= \int_0^1 p(X=1|\theta)p(\theta|D)d\theta \\
&= \int_0^1 \theta \, \mathrm{Beta}(\theta|\alpha_1', \alpha_0')d\theta = E[\theta] = \frac{\alpha_1'}{\alpha_0' + \alpha_1'}
\end{aligned}
$$

- If α_0=α_1=1, we get *Laplace's rule of succession* (add one smoothing)

$$p(X=1|N_1, N_0) = \frac{N_1 + 1}{N_1 + N_0 + 2}$$

Solution to black swan paradox

- If we use a Beta(1,1) prior, the posterior predictive is

$$p(X=1|N_1,N_0)=\frac{N_1+1}{N_1+N_0+2}$$

  so we will never predict black swans are impossible.
- However, as we see more and more white swans, we will come to believe that black swans are pretty rare.

Summary of beta-Bernoulli model

- Prior $p(\theta) = \mathrm{Beta}(\theta|\alpha_1, \alpha_0) = \frac{1}{B(\alpha_1, \alpha_0)}\theta^{\alpha_1 - 1}(1-\theta)^{\alpha_0 - 1}$
- Likelihood $p(D|\theta) = \theta^{N_1}(1-\theta)^{N_0}$
- Posterior $p(\theta|D) = \mathrm{Beta}(\theta|\alpha_1 + N_1, \alpha_0 + N_0)$
- Posterior predictive $p(X=1|D) = \frac{\alpha_1 + N_1}{\alpha_1 + \alpha_0 + N}$