

BAYESIAN LEARNING – LECTURE 3

LECTURE 3: MULTI-PARAMETER. MARGINALIZATION.

MATTIAS VILLANI

DEPARTMENT OF STATISTICS

STOCKHOLM UNIVERSITY

AND

DEPARTMENT OF COMPUTER AND INFORMATION SCIENCE

LINKÖPING UNIVERSITY

- **Multiparameter** models
- **Marginalization**
- **Normal model with unknown variance**
- Bayesian analysis of **multinomial data**
- Bayesian analysis of **multivariate normal data**

- Models with **multiple parameters** $\theta_1, \theta_2, \dots$
- Examples: $x_i \stackrel{iid}{\sim} N(\theta, \sigma^2)$; multiple regression ...
- **Joint posterior distribution**

$$p(\theta_1, \theta_2, \dots, \theta_p | y) \propto p(y | \theta_1, \theta_2, \dots, \theta_p) p(\theta_1, \theta_2, \dots, \theta_p).$$

$$p(\theta | y) \propto p(y | \theta) p(\theta).$$

- **Marginalize** out parameter of no direct interest (**nuisance**).
- Example: $\theta = (\theta_1, \theta_2)'$. **Marginal posterior** of θ_1

$$p(\theta_1 | y) = \int p(\theta_1, \theta_2 | y) d\theta_2 = \int p(\theta_1 | \theta_2, y) p(\theta_2 | y) d\theta_2.$$

■ Model

$$x_1, \dots, x_n \stackrel{iid}{\sim} N(\theta, \sigma^2)$$

■ Prior

$$p(\theta, \sigma^2) \propto (\sigma^2)^{-1}$$

■ Posterior

$$\begin{aligned}\theta | \sigma^2, \mathbf{x} &\sim N\left(\bar{x}, \frac{\sigma^2}{n}\right) \\ \sigma^2 | \mathbf{x} &\sim \text{Inv} - \chi^2(n-1, s^2),\end{aligned}$$

where

$$s^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1}$$

is the usual sample variance.

■ **Simulating** from the posterior :

1. Draw $X \sim \chi^2(n-1)$
2. Compute $\sigma^2 = \frac{(n-1)s^2}{X}$ (this a draw from $\text{Inv-}\chi^2(n-1, s^2)$)
3. Draw a θ from $N\left(\bar{x}, \frac{\sigma^2}{n}\right)$ conditional on the previous draw σ^2
4. Repeat step 1-3 many times.

■ The sampling is implemented in the R program `NormalNonInfoPrior.R`

■ We may derive the **marginal posterior** analytically as

$$\theta|\mathbf{x} \sim t_{n-1}\left(\bar{x}, \frac{s^2}{n}\right).$$

MULTINOMIAL MODEL WITH DIRICHLET PRIOR

- **Categorical counts:** $y = (y_1, \dots, y_K)$, where $\sum_{k=1}^K y_k = n$.
- y_k = no of observations in k th category. Brand choices.
- **Multinomial model:**

$$p(y|\theta) \propto \prod_{k=1}^K \theta_k^{y_k}, \text{ where } \sum_{k=1}^K \theta_k = 1.$$

- **Dirichlet prior:** $\text{Dirichlet}(\alpha_1, \dots, \alpha_K)$

$$p(\theta) \propto \prod_{k=1}^K \theta_k^{\alpha_k - 1}.$$

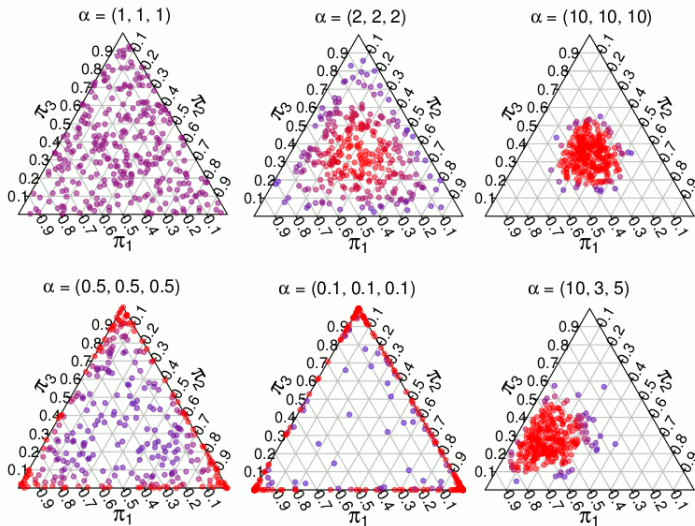
- Moments of $\theta = (\theta_1, \dots, \theta_K)' \sim \text{Dirichlet}(\alpha_1, \dots, \alpha_K)$

$$E(\theta_k) = \frac{\alpha_k}{\sum_{j=1}^K \alpha_j}$$

$$V(\theta_k) = \frac{E(\theta_k) [1 - E(\theta_k)]}{1 + \sum_{j=1}^K \alpha_j}$$

DIRICHLET DISTRIBUTION

Draws from a 3-dimensional Dirichlet with different α



■ 'Non-informative': $\alpha_1 = \dots = \alpha_K = 1$ (uniform and proper).

■ **Simulating** from the Dirichlet distribution:

- Generate $x_1 \sim \text{Gamma}(\alpha_1, 1), \dots, x_K \sim \text{Gamma}(\alpha_K, 1)$.
- Compute $y_k = x_k / (\sum_{j=1}^K x_j)$.
- Then $y = (y_1, \dots, y_K) \sim \text{Dirichlet}(\alpha_1, \dots, \alpha_K)$.

■ **Prior-to-Posterior updating:**

Model: $y = (y_1, \dots, y_K) \sim \text{Multin}(n; \theta_1, \dots, \theta_K)$

Prior: $\theta = (\theta_1, \dots, \theta_K) \sim \text{Dirichlet}(\alpha_1, \dots, \alpha_K)$

Posterior: $\theta|y \sim \text{Dirichlet}(\alpha_1 + y_1, \dots, \alpha_K + y_K)$.

EXAMPLE: MARKET SHARES

- A recent survey among consumer smartphones owners in the U.S. showed that among the 513 respondents:
 - 180 owned an iPhone
 - 230 owned an Android phone
 - 62 owned a Blackberry phone
 - 41 owned some other mobile phone.
- Previous survey: iPhone 30%, Android 30%, Blackberry 20% and Other 20%.
- **Pr(Android has largest share | Data)**
- Prior: $\alpha_1 = 15, \alpha_2 = 15, \alpha_3 = 10$ and $\alpha_4 = 10$ (prior info is equivalent to a survey with only 50 respondents)
- Posterior: $(\theta_1, \theta_2, \theta_3, \theta_4) | \mathbf{y} \sim \text{Dirichlet}(195, 245, 72, 51)$

R CODE FOR MARKET SHARE EXAMPLE

```
# Setting up data and prior
y <- c(180,230,62,41) # The cell phone survey data (K=4)
alpha <- c(15,15,10,10) # Dirichlet prior hyperparameters
nIter <- 1000 # Number of posterior draws

# Defining a function that simulates from a Dirichlet distribution
SimDirichlet <- function(nIter, param){
  nCat <- length(param)
  thetaDraws <- as.data.frame(matrix(NA, nIter, nCat)) # Storage.
  for (j in 1:nCat){
    thetaDraws[,j] <- rgamma(nIter,param[j],1)
  }
  for (i in 1:nIter){
    thetaDraws[i,] = thetaDraws[i,]/sum(thetaDraws[i,])
  }
  return(thetaDraws)
}

# Posterior sampling from Dirichlet posterior
thetaDraws <- SimDirichlet(nIter,y + alpha)
```

R CODE FOR MARKET SHARE EXAMPLE, CONT

```
# Posterior mean and standard deviation of Androids share (in %)
message(mean(100*thetaDraws[,2]))

## 43.5082595875674

message(sd(100*thetaDraws[,2]))

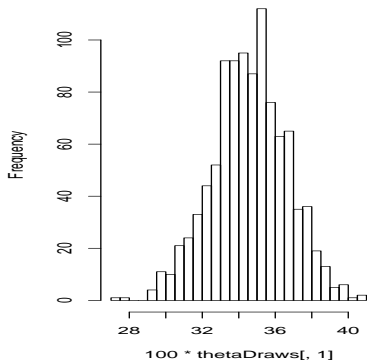
## 2.09082277718707

# Computing the posterior probability that Android is the largest
PrAndroidLargest <- sum(thetaDraws[,2]>apply(thetaDraws[,c(1,3,4)],1,max))/nIter
message(paste('Pr(Android has the largest market share) = ', PrAndroidLargest))

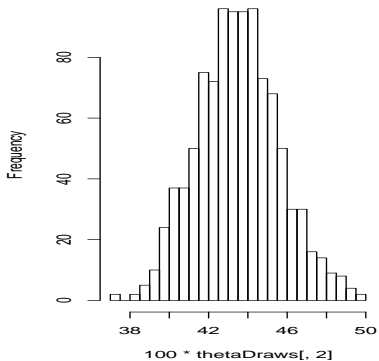
## Pr(Android has the largest market share) = 0.995
```

R CODE FOR MARKET SHARE EXAMPLE, CONT

iPhone market share (%)



Android market share (%)



■ Model

$$y_1, \dots, y_n \stackrel{iid}{\sim} N_p(\mu, \Sigma)$$

where Σ is a known covariance matrix.

■ Density

$$p(y|\mu, \Sigma) = |2\pi\Sigma|^{-1/2} \exp\left(-\frac{1}{2}(y - \mu)' \Sigma^{-1}(y - \mu)\right)$$

■ Likelihood

$$\begin{aligned} p(y_1, \dots, y_n|\mu, \Sigma) &\propto |\Sigma|^{-n/2} \exp\left(-\frac{1}{2} \sum_{i=1}^n (y_i - \mu)' \Sigma^{-1}(y_i - \mu)\right) \\ &= |\Sigma|^{-n/2} \exp\left(-\frac{1}{2} \text{tr} \Sigma^{-1} S_\mu\right) \end{aligned}$$

where $S_\mu = \sum_{i=1}^n (y_i - \mu)(y_i - \mu)'$.

■ Prior

$$\mu \sim N_p(\mu_o, \Lambda_o)$$

■ Posterior

$$\mu|y \sim N(\mu_n, \Lambda_n)$$

where

$$\begin{aligned}\mu_n &= (\Lambda_o^{-1} + n\Sigma^{-1})^{-1}(\Lambda_o^{-1}\mu_o + n\Sigma^{-1}\bar{y}) \\ \Lambda_n^{-1} &= \Lambda_o^{-1} + n\Sigma^{-1}\end{aligned}$$

- Posterior mean is a weighted average of prior and data information.
- **Noninformative prior:** let the precision go to zero: $\Lambda_o^{-1} \rightarrow 0$.