Bayesian Statistics

Fabrizio Ruggeri

Istituto di Matematica Applicata e Tecnologie Informatiche
Consiglio Nazionale delle Ricerche
Via Alfonso Corti 12, I-20133, Milano, Italy, European Union
fabrizio@mi.imati.cnr.it
www.mi.imati.cnr.it/fabrizio/

PROPORTION OF HEAVY SLEEPERS

- Suppose a person is interested in learning about the sleeping habits of American college students
- She hears that doctors recommend eight hours of sleep for an average adult
- What proportion of college students get at least eight hours of sleep?
- Here we think of a population consisting of all American college students and let p represent the proportion of this population who sleep (on a typical night during the week) at least eight hours
- We are interested in learning about the location of p
- Examples taken from Albert, J. (2009), Bayesian Computation with R, Springer

PROPORTION OF HEAVY SLEEPERS

- The value of the proportion p is unknown
- From the Bayesian viewpoint, a person's beliefs about the uncertainty in this proportion are represented by a probability distribution placed on this parameter
- This distribution reflects the person's subjective prior opinion about plausible values of p
- A random sample of students from a particular university will be taken to learn about this proportion
- But first the researcher does some initial research to learn about the sleeping habits of college student
- This research will help her in constructing a prior distribution
- A first paper reports that most students spend only six hours per day sleeping, whereas a second one, based on a sample of 100 students, tells that ”approximately 70% reported receiving only five to six hours of sleep on the weekdays, 28% receiving seven to eight, and only 2% receiving the healthy nine hours for teenagers”

PROPORTION OF HEAVY SLEEPERS

- Based on this information, the person doing the study believes that college students generally get less than eight hours of sleep and so p (the proportion that sleep at least eight hours) is likely smaller than $.5$
- After some reflection, her best guess at the value of p is $.3$
- But it is very plausible that this proportion could be any value in the interval from 0 to $.5$
- A sample of 27 students is taken, and, in this group, 11 record that they had at least eight hours of sleep the previous night
- Based on the prior information and these observed data, the researcher is interested in estimating the proportion p
- In addition, she is interested in predicting the number of students that get at least eight hours of sleep if a new sample of 20 students is taken

PROPORTION OF HEAVY SLEEPERS

- Suppose that our prior density for p is denoted by $\pi(p)$
- If we regard a "success" as sleeping at least eight hours and we take a random sample with s successes and f failures, then the likelihood function is given by

  $L(p) \propto p^s(1-p)^f$
- The posterior distribution is given by $\pi(p|data) \propto L(p)\pi(p)$
- We demonstrate posterior distribution calculations using two different choices of the prior density $\pi(p)$ corresponding to two methods for representing the researcher's prior knowledge about the proportion p

PROPORTION OF HEAVY SLEEPERS

- A simple approach for assessing a prior for p is to write down a list of plausible proportion values and then assign weights to these values

- The person in our example believes that possible values of p are $.05, .15, .25, .35, .45, .55, .65, .75, .85, .95$

- Based on her beliefs, she assigns these values the corresponding weights $1, 5.2, 8, 7.2, 4.6, 2.1, 0.7, 0.1, 0, 0,$
which can be converted to prior probabilities by dividing each weight by the sum

- In R, we define `p` to be the vector of proportion values and `prior` the corresponding weights that we normalize to probabilities

```
p=seq(0.05,0.95,0.1)
prior=c(1,5.2,8,7.2,4.6,2.1,0.7,0.1,0,0)
prior=prior/sum(prior)
plot(p,prior,type="h",ylab="Prior Probability")
```

PROPORTION OF HEAVY SLEEPERS

- In our example, 11 of 27 students sleep a sufficient number of hours, so $s = 11$ and $f = 16$, and the likelihood function is $L(p) \propto p^{11}(1-p)^{16},\ 0 < p < 1$

- The R function `pdisc` in the package `LearnBayes` computes posterior probabilities

- The package `LearnBayes` has to be installed the first time, connecting to a CRAN mirror and downloading the package once found in the (very long) list of available packages

- The package can be used in a new session only after typing `library(LearnBayes)`

- To use `pdisc`, one inputs the vector of proportion values `p`, the vector of prior probabilities `prior`, and a data vector `data` consisting of s and f

- The output of `pdisc` is a vector of posterior probabilities

- The `cbind` command is used to display a table of the prior and posterior probabilities

PROPORTION OF HEAVY SLEEPERS

- The `xyplot` function in the `lattice` package is used to construct comparative line graphs of the prior and posterior distributions (remember of installing `lattice`, if not installed yet, and typing `library(lattice)`)

- Posterior probability of falling in $\{.25, .35, .45\}$ is .940 ($p = .35, .45$ largest) and means: .315 (prior) and .382 (posterior)

```
library(LearnBayes);library(lattice)
data=c(11,16)
post=pdisc(p,prior,data)
round(cbind(p,prior,post),2)
PRIOR=data.frame("prior",p,prior)
POST=data.frame("posterior",p,post)
names(PRIOR)=c("Type","P","Probability")
names(POST)=c("Type","P","Probability")
data=rbind(PRIOR,POST)
xyplot(Probability~P|Type,data=data,layout=c(1,2),
+           type="h",lwd=3,col="black")
postmean=sum(p*post);priormean=sum(p*prior)
postmean;priormean
```

PROPORTION OF HEAVY SLEEPERS

- Since the proportion is a continuous parameter, an alternative approach is to construct a density $\pi(p)$ on the interval $(0, 1)$ that represents the person's initial beliefs

- Suppose she believes that p is equally likely to be smaller or larger than $.3$

- Moreover, she is 90% confident that p is less than $.5$

- *Beta* prior $\pi(p) \propto p^{a-1}(1-p)^{b-1},\ 0 < p < 1$

- The hyperparameters a and b are chosen to reflect the user's prior beliefs about p

- The mean of a beta prior is $m = a/(a + b)$ and the variance of the prior is $v = m(1 - m)/(a + b + 1)$, but it is difficult in practice for a user to assess values of m and v to obtain values of the beta parameters a and b

- It is easier to obtain a and b indirectly through statements about the percentiles

- Here the person believes that the median and 90-th percentiles of the proportion are given, respectively, by $.3$ and $.5$

PROPORTION OF HEAVY SLEEPERS

- The function `beta.select` in the `LearnBayes` package is useful for finding the shape parameters of the beta density that match this prior knowledge

- The inputs to `beta.select` are two lists, `quantile1` and `quantile2`, that define these two prior percentiles, and the function returns the values of the matching beta parameters

```
quantile2=list(p=.9,x=.5)
quantile1=list(p=.5,x=.3)
beta.select(quantile1,quantile2)
```

- We see that this prior information is matched with a beta density with $a = 3.26$ and $b = 7.19$

- Combining this beta prior with the likelihood function, one can show that the posterior density is also of the beta form with updated parameters $a+s = 3.26+11 = 14.26$ and $b + f = 7.19 + 16 = 23.19$

  $$\pi(p|data) \propto p^{14.26-1}(1-p)^{23.19-1}, \ 0 < p < 1$$

PROPORTION OF HEAVY SLEEPERS

- Example of a conjugate analysis, with prior and posterior of same functional form
- Since the prior, likelihood, and posterior are all in the beta family, we can use the R command `dbeta` (d=density) to compute the values of prior, likelihood, and posterior
- Note that the likelihood is not a density but here it is convenient to treat it as such
- The densities are displayed using three applications of the R `curve` command
- The graph helps show that the posterior density in this case compromises between the initial prior beliefs and the information in the data

```
  a=3.26;b=7.19;s=11;f=16
  curve(dbeta(x,a+s,b+f),from=0,to=1,xlab="p",
+ ylab="Density",lty=1,lwd=4)
  curve(dbeta(x,s+1,f+1),add=TRUE,lty=2,lwd=4
  curve(dbeta(x,a,b),add=TRUE,lty=3,lwd=4)
  legend(.7,4,c("Prior","Likelihood","Posterior"),
+ lty=c(3,2,1),lwd=c(3,3,3))
```

PROPORTION OF HEAVY SLEEPERS

-
- We illustrate different ways of summarizing the beta posterior distribution to make inferences about the proportion of heavy sleepers p
- The beta cdf and inverse cdf functions `pbeta` and `qbeta` are useful in computing probabilities and constructing interval estimates for p
- Is it likely that the proportion of heavy sleepers is greater than $.5$?
- This is answered by computing the posterior probability $P(p \geq .5|data)$, i.e. $.0684$, as given by the R command
  $1 - pbeta(0.5, a + s, b + f)$
- This probability is small, so it is unlikely that more than half of the students are heavy sleepers
- A 90% interval estimate, i.e. $(.256, .513)$for p is found by computing the 5-th and 95-th percentiles of the beta density as given by the R command
  $qbeta(c(0.05, 0.95), a + s, b + f)$

PROPORTION OF HEAVY SLEEPERS

- These summaries are exact because they are based on R functions for the beta posterior density
- An alternative method of summarization of a posterior density is based on simulation
- In this case, we can simulate a large number of values from the beta posterior density and summarize the simulated output `ps`
- Using the random beta command `rbeta`, we simulate 1000 random proportion values from the $Beta(a+s,b+f)$ posterior and display the posterior as a histogram of the simulated values

```
ps=rbeta(1000,a+s,b+f)
hist(ps,xlab="p",main="")
```

- The probability that the proportion is larger than .5 is estimated using the proportion of simulated values in this range

```
sum(ps>=0.5)/1000
```

- Why does the probability change if I generate another time the sample `ps`?

PROPORTION OF HEAVY SLEEPERS

- The probability changes at each generation of `ps`, since each sample is different unless we set a ”seed” with the same number each time, i.e.

```
set.seed(22); ps=rbeta(1000,a+s,b+f)
sum(ps>=0.5)/1000
```

- The exact posterior mean of a $Beta(a+s, b+f)$ distribution is

$$\frac{a+s}{(a+s)+(b+f)} = .381,$$

while the approximate one is given by the sample mean (different at each generation unless a seed is set)

```
set.seed(22); ps=rbeta(1000,a+s,b+f); mean(ps)
(a+s)/(a+s+b+f)
```

- A 90% interval estimate, e.g. $(.260, .517)$, can be estimated by the 5-th and 95-th sample quantiles of the simulated sample

PROPORTION OF HEAVY SLEEPERS

- I can compare with a 95% interval estimate or another 90% interval estimate

```
quantile(ps,c(0.05,0.95))
quantile(ps,c(0.025, 0.975))
quantile(ps,c(0.02,0.92))
```

- Sometimes the, say 95%, interval with smallest measure is sought and it is called Highest Posterior Density (HPD) interval

- HPD intervals can be easily found for symmetric unimodal distribution, not for Beta

- These summaries of the posterior density for p based on simulation are approximately equal to the exact values based on calculations from the beta distribution and they are getting closer when increasing the size of the sample

PROPORTION OF HEAVY SLEEPERS

- So far we focused on learning about the population proportion of heavy sleepers p
- Suppose our person is also interested in predicting the number of heavy sleepers $\tilde{y}$ in a future sample of $m = 20$ students
- If the current beliefs about p are contained in the density $m(p)$, then the predictive density of $\tilde{y}$ is given by

$$f(\tilde{y}) = \int f(\tilde{y}|p)m(p)dp$$

- If $m(p) = \pi(p)$, prior density, then we refer to this as the *prior predictive density*
- If $m(p) = \pi(p|data)$, posterior density, then f is a *posterior predictive density*
- How to predict y ”successes” (heavy sleepers) in a new sample of m students?
  - Binomial model, dependent on p, for $\tilde{y}$ successes in a future sample of size m $f_B(y|m,p) = \binom{m}{y} p^y (1-p)^{m-y},\ y = 0, 1, \ldots, m$
  - Distribution on p based on the current knowledge, i.e posterior $\pi(p|data)$

PROPORTION OF HEAVY SLEEPERS

- Consider the case of a discrete distribution $\pi(p|data)$ defined for $\{p_i\}$
- $\Rightarrow f(\tilde{y}) = \sum_i f_B(\tilde{y}|m, p_i)\pi(p_i|data)$
- The function `pdiscp` in the `LearnBayes` package can be used to compute the predictive probabilities when p is given a discrete distribution
- `p` is a vector of proportion values and `prior` a vector of current probabilities
- m is the future sample size and ys the numbers of successes
- The output is a vector of the corresponding predictive probabilities

```
p=seq(0.05,0.95,.1)
prior= c(1,5.2,8,7.2,4.6,2.1,0.7,0.1,0,0)
prior=prior/sum(prior);m=20;ys=0:20
pred=pdiscp(p,prior,m,ys)
round(cbind(0:20,pred),3)
```

PROPORTION OF HEAVY SLEEPERS

- We see from the output that the most likely numbers of successes in this future sample are $\tilde{y} = 5$ and $\tilde{y} = 6$.

- Suppose instead that we model our beliefs about p using a $Beta(a, b)$ prior

- In this case, we can analytically integrate out p to get a closed-form expression for the predictive density

$$\begin{aligned} f(\tilde{y}) &= \int f_B(\tilde{y}|m,p)\pi(p)dp \\ &= \binom{m}{\tilde{y}} \frac{B(a+\tilde{y}, b+m-\tilde{y})}{B(a,b)}, \; \tilde{y} = 0, 1, \ldots, m \end{aligned}$$

where $B(a,b) = \dfrac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}$ is the beta function

PROPORTION OF HEAVY SLEEPERS

- The predictive probabilities using the beta density are computed using `pbetap`

- The inputs to this function are the vector `ab` of beta parameters a and b, the size of the future sample `m`, and the vector of numbers of successes `y`

- The output is a vector of predictive probabilities corresponding to the values in `y`

- We illustrate this computation using the $Beta(3.26, 7.19)$ prior used to reflect the person's beliefs about the proportion of heavy sleepers at the school

```
ab=c(3.26,7.19)
m=20;ys=0:20
pred=pbetap(ab,m,ys)
```

- We computed the predictive density for two choices of prior densities

- One convenient way of computing a predictive density for any prior is by simulation

- To obtain $\tilde{y}$, we first simulate, say, p^* from $\pi(p)$, and then simulate $\tilde{y}$ from the binomial distribution $f_B(\tilde{y}|p^*)$

PROPORTION OF HEAVY SLEEPERS

- We demonstrate this simulation approach for the $Beta(3.26, 7.19)$ prior
- We first simulate 1000 draws from the prior and store the simulated values in `p`
- Then we simulate values of $\tilde{y}$ for these random p's using the `rbinom` function
- To summarize the simulated draws of $\tilde{y}$, we first use the `table` command to tabulate the distinct values
- We save the frequencies of $\tilde{y}$ in a vector `freq`
- Then we convert the frequencies to probabilities by dividing each frequency by the sum and use the `plot` command to graph the predictive distribution

```
p=rbeta(1000,3.26,7.19)
y=rbinom(1000,20,p)
table(y)
freq=table(y)
ys=as.integer(names(freq))
predprob=freq/sum(freq)
plot(ys,predprob,type="h",xlab="y", ylab="Predictive Probability")
```

PROPORTION OF HEAVY SLEEPERS

- Suppose we wish to summarize this discrete predictive distribution by an interval that covers at least 90% of the probability
- The R function `discint` in the `LearnBayes` package is useful for this purpose
- In the output, the vector `ys` contains the values of $\tilde{y}$ and `predprob` contains the associated probabilities found from the table output
- The matrix `dist` contains the probability distribution with the columns `ys` and `predprob`
- The function `discint` has two inputs: the matrix `dist` and a given coverage probability `covprob`
- The output is a list where the component `set` gives the credible set and `prob` gives the exact coverage probability

```
dist=cbind(ys,predprob)
dist
covprob=.9
discint(dist,covprob)
```

HEART TRANSPLANT MORTALITY RATE

- Consider the problem of learning about the rate of success of heart transplant surgery of a particular hospital in the United States
- A first problem is related to notion of ”success” in this case
- A very unrealistic notion would be ”success = the person has the same chance of dying in the next five years as another one who did not need a transplant”
- For this hospital, we observe the number of transplant surgeries n, and the number of deaths within 30 days of surgery y is recorded (this is the notion of ”success” that we consider)
- One could predict the probability of death for an individual patient
- This prediction could be based on a model (e.g. logistic regression) that uses information such as patients’ medical condition before surgery, gender, and race but we will make the very simplifying assumption that all patients have the same conditions

HEART TRANSPLANT MORTALITY RATE

- We will rather focus on the expected number of deaths in a hospital and such number will depend on two quantities
  - The number n of transplants
  - The mortality rate λ per each individual
- Here λ is the parameter of interest (we have always to identify what we are interested in!) and we will analyze it following a Bayesian approach, i.e. using both data and expertise (when available)
- We model the number of deaths y with a Poisson distribution with mean $n\lambda$
- A Poisson r.v. $X \sim \mathcal{P}(\theta)$ has density $f(x|\theta) = \frac{\theta^x}{x!}e^{-\theta},\ x = 0, 1, 2, \ldots$
- The mean of such r.v. is $E(X) = \theta$
- Given a sample $\underline{X} = (X_1, \ldots, X_m)$ from $X \sim \mathcal{P}(\theta)$, the MLE is $\widehat{\theta} = \frac{\sum_{i=1}^m X_i}{m}$

HEART TRANSPLANT MORTALITY RATE

- In our case (y deaths out of n transplants), the MLE is $\widehat{\lambda} = y/n$
- Unfortunately, this estimate can be poor when the number of deaths y is close to zero (consider also the cases of 1 deaths out of 5 transplants vs. 2 deaths out of 5 transplants)
- In this situation when small death counts are possible, it is desirable to use a Bayesian estimate that uses prior knowledge about the size of the mortality rate
- A gamma distribution $\mathcal{G}(\alpha, \beta)$ is a convenient choice for a prior on λ
- Its density is $f(\lambda) = \dfrac{\beta^\alpha}{\Gamma(\alpha)}\lambda^{\alpha-1}e^{-\beta\lambda},\ \lambda, \alpha, \beta > 0$
- A convenient source of prior information is heart transplant data from a small group of hospitals that we believe has the same rate of mortality as the rate from the hospital of interest

HEART TRANSPLANT MORTALITY RATE

- Suppose we observe the number of deaths z_j and the number of transplants n_j for ten hospitals ($j = 1, \ldots, 10$), where each z_j is Poisson with mean $n_j \lambda$
- We would like to find a posterior distribution for λ based on those past data and use it as a prior distribution for λ for this particular hospital
- If we assign λ the standard *noninformative prior* $\pi(\lambda) \propto \lambda^{-1}$, then the updated distribution for λ, given these data from the ten hospitals, is

  $$\pi(\lambda) \propto \lambda^{\sum_{j=1}^{10} z_j - 1} e^{-\lambda \sum_{j=1}^{10} n_j}$$

- The information leads to a prior $\mathcal{G}(\alpha, \beta)$ with $\alpha = \sum_{j=1}^{10} z_j$ and $\beta = \sum_{j=1}^{10} n_j$
- In this example, we consider $\sum_{j=1}^{10} z_j = 16$ and $\sum_{j=1}^{10} n_j = 15174$ so that the prior is $\mathcal{G}(16, 15174)$
- The posterior is $\mathcal{G}(\alpha + y, \beta + n)$ since $\pi(\lambda | y, n) \propto \lambda^y e^{-n\lambda} \cdot \lambda^{\alpha - 1} e^{-\beta \lambda}$

HEART TRANSPLANT MORTALITY RATE

- We now present an approach useful to check if prior and ssmling densities have been specified properly

- The (prior) predictive density of y (i.e. before any data are observed) can be computed using the formula

$$f(y) = \frac{f(y|\lambda)\pi(\lambda)}{\pi(\lambda|y)}$$

- That follows from Bayes Theorem: $\pi(\lambda|y) = \dfrac{f(y|\lambda)\pi(\lambda)}{f(y)}$

- By using the posterior density, one performs inference about the unknown parameter conditional on the Bayesian model that includes the assumptions of sampling density and the prior density

HEART TRANSPLANT MORTALITY RATE

- One can check the validity of the proposed model by inspecting the predictive density
- If the observed data value y_{obs} is consistent with the predictive density $f(y)$, then the model seems reasonable
- On the other hand, if y_{obs} is in the extreme tail portion of the predictive density, then this casts doubt on the validity of the Bayesian model, and perhaps the prior density or the sampling density has been misspecified

HEART TRANSPLANT MORTALITY RATE

- We consider inference about the heart transplant death rate for two hospitals: one that has experienced a small number of surgeries and a second that has experienced many surgeries

- First consider hospital A, which experienced only one death ($y_{obs} = 1$) with 66 transplants

- The MLE of this hospital's rate, 1/66, is suspect due to the small observed number of deaths

- The following R calculations illustrate the Bayesian calculations

- After the gamma prior parameters α and β and number of surgeries n are obtained, the predictive density of the values $y = 0, 1, \ldots, 10$ is found by using the preceding formula and the R functions $dpois$ and `dgamma`

- The formula for the predictive density is valid for all λ, but to ensure that there is no underflow in the calculations, the values of $f(y)$ are computed for the prior mean value $\lambda = \alpha/\beta$

HEART TRANSPLANT MORTALITY RATE

- Note that practically all of the probability of the predictive density is concentrated on the two values $y = 0$ and $y = 1$
- The observed number of deaths ($y_{obs=1}$) is in the middle of this predictive distribution, so there is no reason to doubt our Bayesian model
- The posterior density of λ can be summarized by simulating 1000 values from the gamma density

```
alpha=16;beta=15174
yobs=1; n=66
y=0:10
lam=alpha/beta
py=dpois(y,lam*n)*dgamma(lam,alpha,beta)/dgamma(lam,alpha+y,beta+n)
cbind(y,round(py,3))
lambdaA=rgamma(1000,alpha+yobs,beta+n)
par(mfrow=c(2,1));hist(lambdaA);boxplot(lambdaA)
summary(lambdaA)
```

HEART TRANSPLANT MORTALITY RATE

- Consider the estimation of a different hospital that experiences many surgeries
- Hospital B had $y_{obs} = 4$ deaths, with $n = 1767$ transplants
- For these data, we again have R compute the prior predictive density and simulate 1000 draws from the posterior density using the `rgamma` command
- Again we see that the observed number of deaths seems consistent with this model since $y_{obs} = 4$ is not in the extreme tails of this distribution

```
n=1767; yobs=4; y=0:10
py=dpois(y,lam*n)*dgamma(lam,alpha,beta)/dgamma(lam,alpha+y,beta+n)
cbind(y,round(py,3))
lambdaB=rgamma(1000,alpha+yobs,beta+n)
```

HEART TRANSPLANT MORTALITY RATE

- To see the impact of the prior density on the inference, it is helpful to display the prior and posterior distributions on the same graph
- Density estimates of the simulated draws from the posterior distributions of the rates are shown for hospitals A and B
- The gamma prior density is also displayed in each case
- We see that for hospital A, with relatively little experience in surgeries, the prior information is significant and the posterior distribution resembles the prior distribution
- In contrast, for hospital B, with many surgeries, the prior information is less influential and the posterior distribution resembles the likelihood function

```
par(mfrow=c(2,1))
plot(density(lambdaA),main="HOSPITAL A",xlab="lambdaA",lwd=3)
curve(dgamma(x,alpha,beta),add=TRUE)
legend("topright",legend=c("prior","posterior"),lwd=c(1,3))
plot(density(lambdaB),main="HOSPITAL B",xlab="lambdaB",lwd=3)
curve(dgamma(x,alpha,beta),add=TRUE)
legend("topright",legend=c("prior","posterior"),lwd=c(1,3))
```

TEST OF THE FAIRNESS OF A COIN

- Suppose you are interested in assessing the fairness of a coin
- You observe y, binomially distributed with parameters n and p, and you are interested in testing the hypothesis H that $p = .5$
- If y is observed, then it is usual practice to make a decision on the basis of the `p-value`: $2 \min\{P(Y \leq y), P(Y \geq y)\}$
- If this p-value is small, then you reject the hypothesis H and conclude that the coin is not fair
- Suppose, for example, the coin is flipped 20 times and only 5 heads are observed
- In R we compute the probability .021 of obtaining five or fewer heads

```
pbinom(5,20,0.5)
```

- The p-value here is $2 \times .021 = .042$
- Since this value is smaller than the common significance level of .05, you would decide to reject the hypothesis H and conclude that the coin is not fair

TEST OF THE FAIRNESS OF A COIN

- Consider this problem from a Bayesian perspective
- There are two possible models here – either the coin is fair ($p = .5$) or the coin is not fair ($p \neq .5$)
- Suppose that you are indifferent between the two possibilities, so you initially assign each model a probability of $1/2$
- Now, if you believe the coin is fair, then your entire prior distribution for p would be concentrated on the value $p = .5$
- If instead the coin is unfair, you would assign a different prior distribution on $(0, 1)$, call it $\pi_1(p)$, that would reflect your beliefs about the probability of an unfair coin
- Suppose you assign a $Beta(a, a)$ prior on p
- This beta distribution is symmetric about .5 – it says that you believe the coin is not fair, and the probability is close to $p = .5$

TEST OF THE FAIRNESS OF A COIN

- To summarize, your prior distribution in this testing situation can be written as the mixture

  $\pi(p) = .5I(p = .5) + .5I(p \neq .5)\pi_1(p)$,
  where $I(A)$ is an indicator function equal to 1 if the event A is true and otherwise is equal to 0

- After observing the number of heads in n tosses, we would update our prior distribution by Bayes' rule

TEST OF THE FAIRNESS OF A COIN

- After observing the number of heads in n tosses, we would update our prior distribution by Bayes' rule

- The posterior density for p can be written as

  $\pi(p|y) = \lambda(y) I(p = .5) + (1 - \lambda(y))\pi_1^*(p|y)$, where

  – $\pi_1^*(p|y)$ is $Beta(a + y, a + n - y)$

  – $\lambda(y)$ is the posterior probability of the model where the coin is fair, i.e.

  $$\lambda(y) = \frac{.5p(y|.5)}{.5p(y|.5) + .5f_1(y)}$$

  – $p(y|.5)$ is the binomial density for y when $p = .5$

  – $f_1(y)$ is the (prior) predictive density for y using the beta density $\pi_1(p)$ (i.e. the integral of binomial*beta w.r.t. p)

- In R, the posterior probability of fairness $\lambda(y)$ is easily computed

TEST OF THE FAIRNESS OF A COIN

- The R command `dbinom` will compute the binomial probability $p(y|.5)$
- The predictive density for y can be computed using the identity

$$f_1(y) = \frac{f(y|p)\pi_1(p)}{\pi_1^*(p|y)}$$

- Assume first that we assign a $Beta(10, 10)$ prior for p when the coin is not fair and we observe $y = 5$ heads in $n = 20$ tosses
- The posterior probability of fairness (0.280) is stored in the R variable `lambda`
- We get the surprising result that the posterior probability of the hypothesis of fairness H is $.28$, which is less evidence against fairness than is implied by the p-value calculation above

```
n=20;y=5;a=10;p=0.5
m1=dbinom(y,n,p)*dbeta(p,a,a)/dbeta(p,a+y,a+n-y)
lambda=dbinom(y,n,p)/(dbinom(y,n,p)+m1)
lambda
```

TEST OF THE FAIRNESS OF A COIN

- The function `pbetat` in the `LearnBayes` package performs a test of a binomial proportion
- The inputs to the function are:
  - the value of p to be tested
  - the prior probability of that value
  - a vector of parameters of the beta prior when the hypothesis is not true
  - a vector of numbers of successes and failures
- In this example, the syntax would be `pbetat(p,.5,c(a,a),c(y,n-y))`
- The output variable `post` is the posterior probability that $p = .5$, which agrees with the calculation
- The output variable `bf` is the *Bayes factor* in support of the null hypothesis