

Introduction to Bayes using Discrete Priors

Jim Albert

March 18, 2018

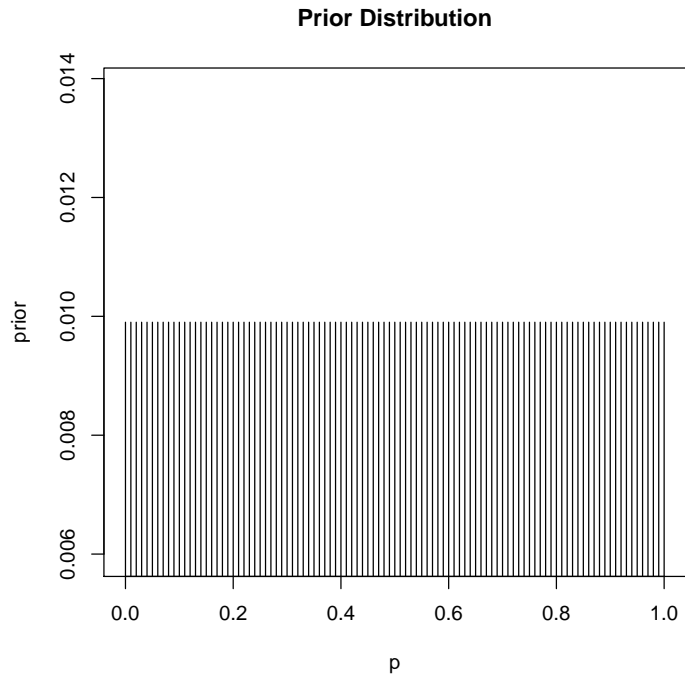
Learning About a Proportion

A Discrete Prior

Consider a population of “successes” and “failures” where the proportion of successes is p . Suppose p takes on the discrete set of values 0, .01, ..., .99, 1 and one assigns a uniform prior on these values. We enter the values of p and the associated probabilities into the vectors `p` and `prior`, respectively.

```
> p <- seq(0, 1, by = 0.01)
> prior <- 1 / 101 + 0 * p

> plot(p, prior,
+      type="h",
+      main="Prior Distribution")
```

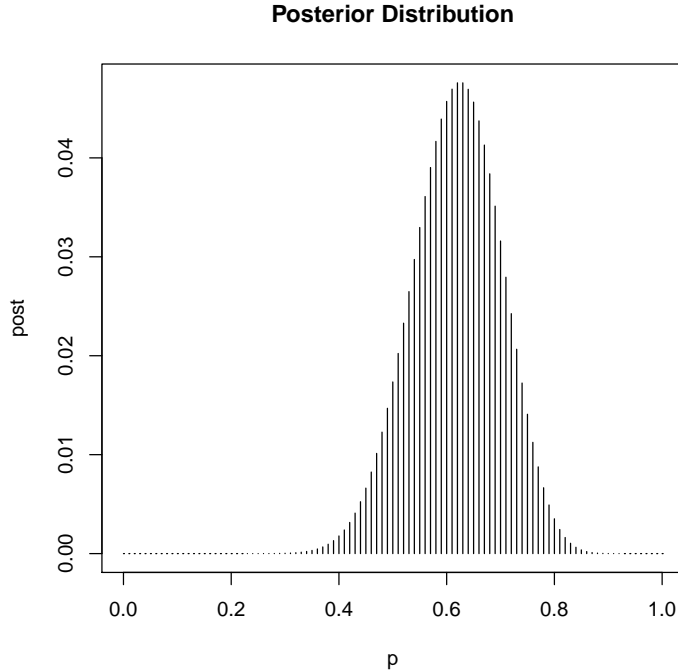


Posterior Distribution

Suppose one takes a random sample from the population without replacement and observes 20 successes and 12 failures. The function `pdisc` in the `LearnBayes` package computes the associated posterior probabilities for p . The inputs to `pdisc` are the prior (vector of values of p and vector of prior probabilities) and a vector containing the number of successes and failures.

```
> library(LearnBayes)
> post <- pdisc(p, prior, c(20, 12))

> plot(p, post,
+      type="h",
+      main="Posterior Distribution")
```



A highest probability interval for a discrete distribution is obtained using the `discint` function. This function has two inputs: the probability distribution matrix where the first column contains the values and the second column contains the probabilities, and the desired probability content. To illustrate, we compute a 90 percent probability interval for p from the posterior distribution.

```
> discint(cbind(p, post), 0.90)
```

```
$prob
```

```
[1] 0.9001245
```

```
$set
```

```
[1] 0.49 0.50 0.51 0.52 0.53 0.54 0.55 0.56 0.57 0.58 0.59 0.60 0.61 0.62 0.63  
[16] 0.64 0.65 0.66 0.67 0.68 0.69 0.70 0.71 0.72 0.73 0.74 0.75
```

The probability that p falls in the interval (0.49, 0.75) is approximately 0.90.

Prediction

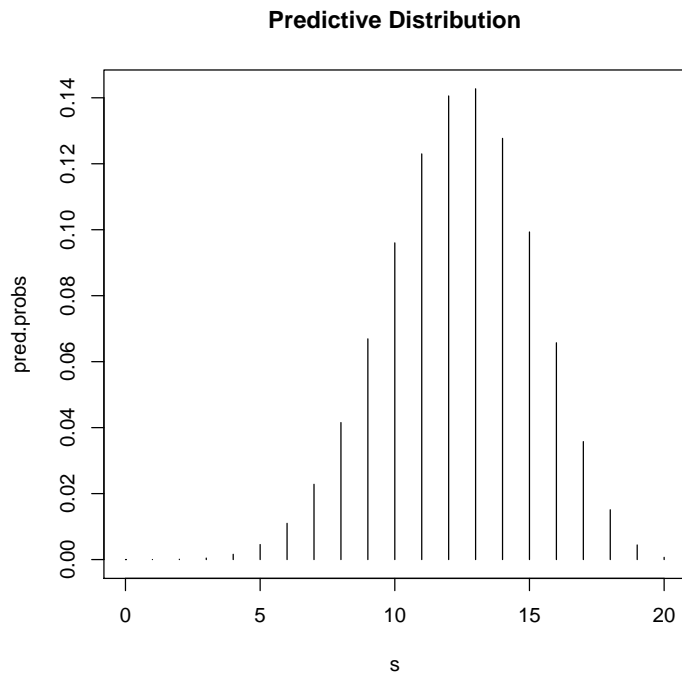
Suppose a new sample of size 20 is to be taken and we're interested in predicting the number of successes. The current opinion about the proportion is reflected in the posterior distribution stored in the vectors `p` and `post`. We store the possible number of successes in the future sample in `s` and the function `pdiscp` computes the corresponding predictive probabilities.

```

> n <- 20
> s <- 0:20
> pred.probs <- pdiscp(p, post, n, s)

> plot(s, pred.probs,
+      type="h",
+      main="Predictive Distribution")

```



Learning About a Poisson Mean

Discrete models can be used for other sampling distributions using the `discrete.bayes` function. To illustrate, suppose the number of accidents in a particular year is Poisson with mean λ . A priori one believes that λ is equally likely to take on the values 20, 21, ..., 30. We put the prior probabilities $1/11$, ..., $1/11$ in the vector `prior` and use the `names` function to name the components of this vector with the values of λ .

```

> prior <- rep(1/11, 11)
> names(prior) <- 20:30

```

One observes the number of accidents for ten weeks – these values are placed in the vector `y`:

```
> y <- c(24, 25, 31, 31, 22, 21, 26, 20, 16, 22)
```

To compute the posterior probabilities, we use the function `discrete.bayes`; the inputs are the Poisson sampling density `dpois`, the vector of prior probabilities `prior`, and the vector of observations `y`.

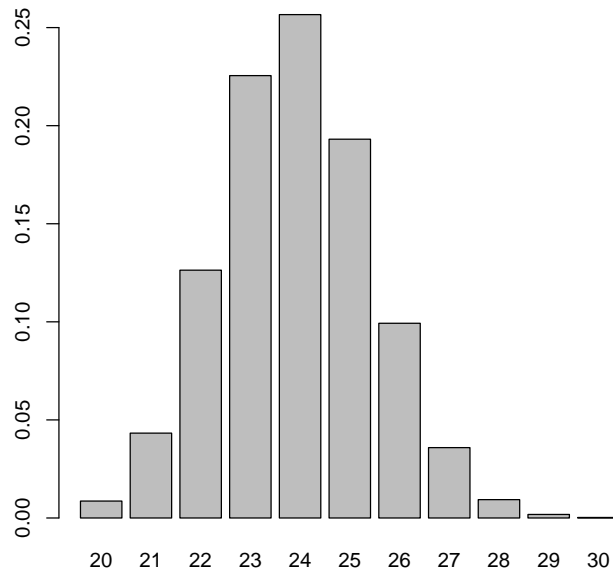
```
> post <- discrete.bayes(dpois, prior, y)
```

One can display the posterior probabilities by use of the `print` method, one displays the posterior probabilities by the `plot` method, and one summarizes the posterior distribution by the `summary` method.

```
> print(post)
```

	20	21	22	23	24	25
0.0086284462	0.0432554621	0.1263286996	0.2255195224	0.2566196329	0.1931041454	
	26	27	28	29	30	
0.0992616593	0.0358720144	0.0093507519	0.0017989279	0.0002607378		

```
> plot(post)
```



```
> summary(post)
```

```
$mean  
[1] 23.90475
```

```
$sd  
[1] 1.538209
```

```
$coverage  
[1] 0.9008337
```

```
$set  
[1] 22 23 24 25 26
```