Department of Modern Languages and Literatures and in Pluridisciplinary Studies at the Centre of Canadian and European Studies at the University of Montreal, Quebec, Canada. Before joining Henderson State University in 2008, she was Instructor of German at George Brown College in Toronto, and Instructor of German and French at Clarion University in Pennsylvania.

Three-Parameter Bayesian Estimation Problem

Michael Lloyd, Professor of Mathematics

Abstract

A Bayesian estimate of three parameters will be derived using PERT, triangular distributions, and R. An analyst at a large corporation introduced me to this problem.

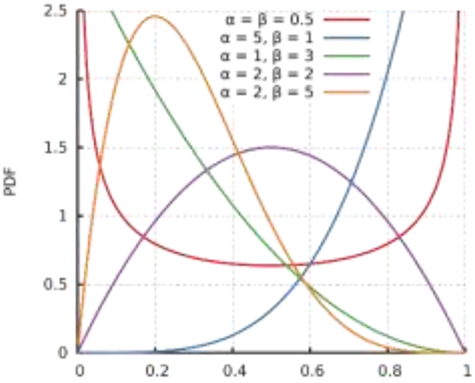
Distributions

Two-Parameter Beta Distribution $\mathrm{Beta}(\alpha, \beta)$ has the following probability density function:

$$\text{pdf } f(x) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{\mathrm{B}(\alpha,\beta)}, \text{ where } 0 < x < 1, \mathrm{B}(\alpha,\beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}, \alpha > 0, \beta > 0$$

The accompanying graph shows Beta probability density functions for various values of its parameters.

The Four-Parameter Beta Distribution $\mathrm{Beta}(\alpha, \beta, a, c)$ is a linear transformation of $\mathrm{Beta}(\alpha, \beta)$ that maps the support from $[0,1]$ to $[a, c]$. An application of the change-of-variable technique gives the following probability density function for the Four-Parameter Beta Distribution:

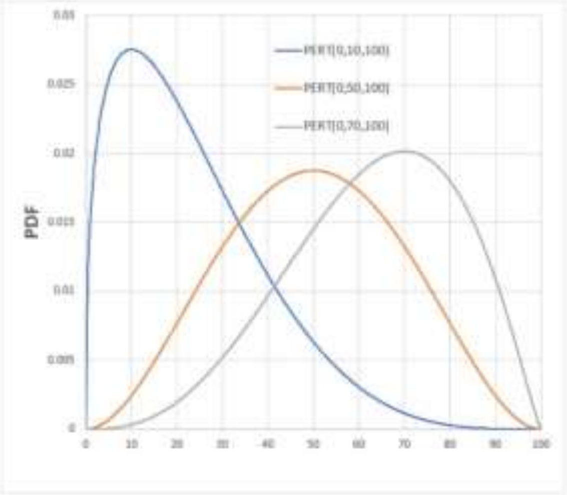


https://en.wikipedia.org/wiki/Beta_distribution

$$\text{pdf } f(x) = \frac{(x-a)^{\alpha-1}(c-x)^{\beta-1}}{\mathrm{B}(\alpha,\beta)(c-a)^{\alpha+\beta-1}}, \text{ where } a < x < c, \alpha > 0, \beta > 0, a < c$$

The $PERT(a, b, c)$ distribution is a special case of the Four-Parameter Beta Distribution where $a < b < c$, b is the mode and the mean is $\mu = \frac{a+4b+c}{6}$. That is,

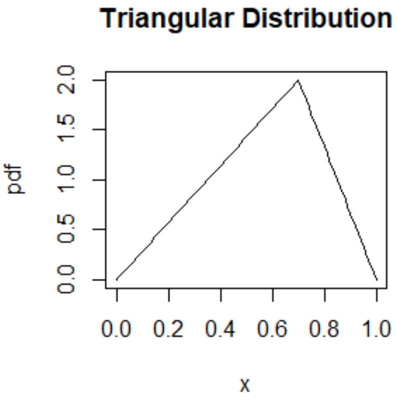
$$\mathrm{PERT}(a,b,c) = \mathrm{Beta}\left(\frac{4b+c-5a}{c-a}, \frac{5c-a-4b}{c-a}, a, c\right)$$



The accompanying graph shows typical PERT probability density functions for various values of the parameter.

This distribution was proposed in 1962 by CE Clark, and the abbreviation PERT originally stood for Program Evaluation and Review Technique.

https://en.wikipedia.org/wiki/PERT_distribution



As is done in practice, in this paper, we will replace the PERT distributions with the simpler Triangular Distribution $\mathrm{Triang}(a, b, c)$ where $a < b < c$, b is the mode.

Consulting Problem

Assume the random variables $(A, B, C) = (min, mode, max)$ are independent where

$$A \sim \mathrm{Triang}(6,10,14), B \sim \mathrm{Triang}(30,50,70), C \sim \mathrm{Triang}(900,1000,1100)$$

Think of this as a simple way to turn the following uncertainties into distributions:

$$A = 10 \pm 4, B = 50 \pm 20, C = 1000 \pm 100$$

Observe a single random sample from $\mathrm{PERT}(a, b, c)$ with pdf $f(x|a, b, c)$. We will update the mode parameters a, b, c for the particular sample $x = 8$ using Bayesian techniques. These numbers were supplied by an analyst at a large corporation who does not wish to be identified.

Let $h(p)$, where $p = (a, b, c)$, be the prior probability density function for the parameters $A, B,$ and C.

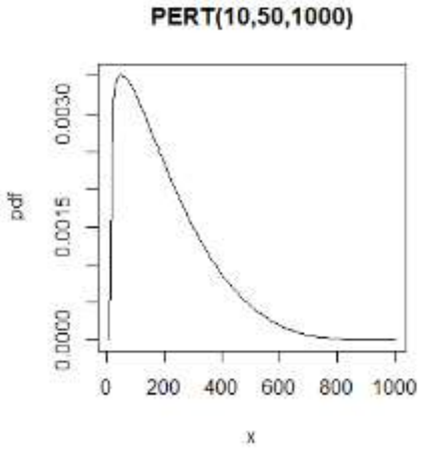
We will use the R library mc2d (Tools for Two-Dimensional Monte-Carlo Simulations) to easily create triangular densities. The following R script will create the prior probability distribution h for (A, B, C). By independence, the joint probability density function is the product of the marginal density functions.

```
library(mc2d)
prior = function(p) {
 dtriang( x=p[1], min=6, mode=10, max=14 )* # a pdf
 dtriang( x=p[2], min=30, mode=50, max=70 )* # b pdf
 dtriang( x=p[3], min=900, mode=1000, max=1100 )  # c pdf
 } # prior
```

We will use the R library cubature (Adaptive Multivariate Integration over Hypercubes) to check that h integrates to 1 over its support ($[6,14] \times [30,70] \times [900,1100] \subset \mathbb{R}^3$). The graph of h is a scalar field over this box with maximal density at $(10,50,1000)$.

```
library(cubature)
adaptIntegrate( prior, lowerLimit = c(6, 30, 900),
 upperLimit = c(14, 70, 1100) )
$integral
[1] 1
$error
[1] 2.220446e-16
$functionEvaluations
[1] 495
```

The following R script checks that the sample PERT probability density function $f(x, 10,50,1000)$ has area 1.



```
ol = function(x) { dpert(x, min=10, mode=50, max=1000) }
curve( ol , xlim=c(10,1000) ,ylab='pdf' ,
 main='PERT(10,50,1000)')
integrate( f=ol , lower=10, upper=1000 )
1 with absolute error < 9e-08
```

The joint distribution of (X, A, B, C) when $X = 8$ is $joint(a, b, c) = f(x = 8|a, b, c)h(a, b, c)$. The posterior probability distribution $p(a, b, c)$ for the parameter (A, B, C) given $x = 8$ is proportional this joint distribution: $p(a, b, c) \propto f(x = 8|a, b, c)h(a, b, c)$

```
joint = function(p)
 { dpert(x=8, min=p[1], mode=p[2], max=p[3]) * prior(p) }
```

```
# The variable p in R is a vector with length three
```

We used the cubature library to integrate the joint distribution over the support of the prior distribution while confirming that the error is small:

$$\int_{6}^{14}\int_{30}^{70}\int_{900}^{1100} joint\ da\ db\ dc \approx 0.0002406715$$

```
adaptIntegrate( joint, lowerLimit = c(6, 30, 900),
 upperLimit = c(14, 70, 1100) )
$integral
[1] 0.0002406715
$error
[1] 2.398408e-09
$functionEvaluations
[1] 3729
```

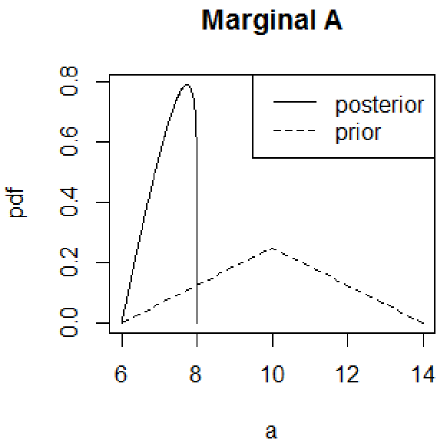
The proportionality constant is the reciprocal of this integral. Hence, the posterior probability density function $k(a, b, c|x = 8)$ is computed as follows:

```
post = function(p) { joint(p) / 0.0002406715 }
```

Here is the posterior marginal density function for the random variable A.

$$f_A(a) = \int_{30}^{70}\int_{900}^{1100} k(a, b, c|x = 8)\ dc\ db$$

Inspect the accompanying graph to see how the sample $x = 8$ had a large effect on the prior distribution to create the posterior distribution.



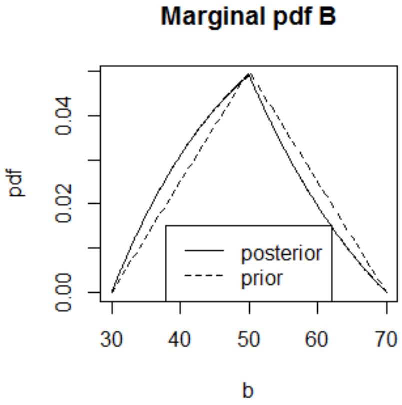
```
margA = function(x) {
 cA = function(q) { post( c(x,q) ) }
 adaptIntegrate(cA, lowerLimit = c(30, 900),
  upperLimit = c(70, 1100))$integral
} # marg A
xa = seq( 6 , 8 , .01 )
ya = unlist(lapply( xa , margA )) # flatten list
plot( ya ~ xa , type='l' , main='Marginal A' , xlim=c(6,14) ,
 xlab='a' , ylab = 'pdf' )
curve(dtriang( x , min=6, mode=10, max=14 ) , add=T ,lty=2)
legend( 'topright' , c('posterior','prior') , lty=1:2 )
```

Posterior marginal probability density function of the random variable B is given by

$$f_B(b) = \int_{6}^{14} \int_{900}^{1100} k(a, b, c|x = 8)\, dc\, da$$

```
margB = function(x) {
 sl = function(q) { post( c(q[1],x,q[2]) ) }
 adaptIntegrate( sl , lowerLimit = c(6, 900),
  upperLimit = c(14, 1100))$integral
} # marg B
xb = seq( 30 , 70 , .1 )
system.time( {
 yb = unlist(lapply( xb , margB ))
} )
plot( yb ~ xb , type='l' , main='Marginal pdf B' ,
 xlab='b' , ylab = 'pdf' )
curve( dtriang( x , min=30, mode=50, max=70 ) , add=T , lty=2 )
legend('bottom' , c('posterior','prior') , lty=1:2 )
```

It took my computer 22.6 seconds to execute the R command `lapply` to make the probability density vector. The sample $x = 8$ had small, but noticeable effect on the prior distribution to make the posterior.

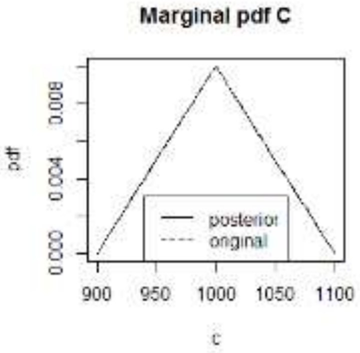


The posterior marginal density function of the random variable C is

$$f_C(c) = \int_{6}^{14} \int_{30}^{70} k(a, b, c|x = 8)\, db\, da$$

```
margC = function(x) {
 sl = function(q) { post( c(q,x) ) }
 adaptIntegrate(sl, lowerLimit = c(6, 30),
  upperLimit = c(14, 70))$integral
} # marg C
xc = seq( 900 , 1100 , 5 )
yc = unlist(lapply( xc , margC )) # flatten list
plot( yc ~ xc , type='l' , main='Marginal pdf C' ,
 xlab='c' , ylab = 'pdf' )
curve( dtriang( x , min=900, mode=1000, max=1100 ) , add=T ,
lty=2)
legend('bottom' , c('posterior','original') , lty=1:2 )
```

The sample $x = 8$ had negligible effect on the prior distribution.



Minimize $E[(\Theta - \theta_{est})^2 | X = 8]$ to estimate the parameter θ using the square-loss function. The new minimum is the expected value of the marginal distribution A computed using the posterior distribution. (The old value was 10.)

$$a_{est} = E[A|x = 8] = \int_{6}^{14} \int_{30}^{70} \int_{900}^{1100} a\, k(a, b, c|x = 8)\, dc\, db\, da \approx 7.27$$

```
mi = function(p) { p[1] * post(p) }
adaptIntegrate(mi, lowerLimit = c(6, 30, 900),
 upperLimit = c(14, 70, 1100))
$integral
[1] 7.269512
$error
[1] 6.980918e-05
$functionEvaluations
[1] 3795
```

Similarly, here is the new mode. (The old value was 50.)

$$b_{est} = E[B|x = 8] = \int_{6}^{14} \int_{30}^{70} \int_{900}^{1100} b\, k(a, b, c|x = 8)\, dc\, db\, da \approx 48.5$$

```
mo = function(p) { p[2] * post(p) }
adaptIntegrate(mo, lowerLimit = c(6, 30, 900),
 upperLimit = c(14, 70, 1100))
$integral
[1] 48.46184
$error
[1] 0.0004537082
$functionEvaluations
[1] 3267
```

As expected, the new maximum is the same as the old value of 1000 to four significant digits:

$$c_{est} = E[C|x = 8] = \int_{6}^{14} \int_{30}^{70} \int_{900}^{1100} c\, k(a, b, c|x = 8)\, dc\, db\, da \approx 1000$$

```
ma = function(p) { p[3] * post(p) }
adaptIntegrate(ma, lowerLimit = c(6, 30, 900),
 upperLimit = c(14, 70, 1100))
$integral
[1] 999.6467
$error
[1] 0.009829518
$functionEvaluations
[1] 3465
```

Minimize $E[|\Theta - \theta_{est}||X = 8]$ to estimate the parameter θ using the absolute-value loss function. This is computation is slightly more tedious. Here the absolute loss estimate for a:

$$\frac{1}{2} = P\left[A < \mathrm{M}_{A|X=8}\right] = \int_{6}^{a_{est}} \int_{30}^{70} \int_{900}^{1100} k(a, b, c|x = 8)\; dc\; db\; da \to a_{est} \approx 7.33$$

```
F.A = function(x){
adaptIntegrate( post , lowerLimit = c(6, 30, 900),
 upperLimit = c(x, 70, 1100) )$integral - 0.5 }
uniroot( F.A , interval = c(6,14) )
```

Here the absolute loss estimate for b:

$$\frac{1}{2} = P\left[B < \mathrm{M}_{B|X=8}\right] = \int_{6}^{14} \int_{30}^{b_{est}} \int_{900}^{1100} k(a, b, c|x = 8)\; dc\; db\; da \to b_{est} \approx 48.4$$

```
F.B = function(x){
adaptIntegrate( post , lowerLimit = c(6, 30, 900),
 upperLimit = c(14 , x, 1100) )$integral - 0.5 }
uniroot( F.B , interval = c(30,70) )
```

$$\frac{1}{2} = P\left[C < \mathrm{M}_{C|X=8}\right] = \int_{6}^{14} \int_{30}^{70} \int_{900}^{c_{est}} k(a, b, c|x = 8)\; dc\; db\; da \to c_{est} \approx 1000$$

```
F.C = function(x){
adaptIntegrate( post , lowerLimit = c(6, 30, 900),
 upperLimit = c(14 , 70, x) )$integral - 0.5 }
uniroot( F.C , interval = c(900,1100) )
```

Conclusion

Recall that the prior parameters were $(a, b, c) = (10, 50,1000)$. The updated parameters using the square-loss function are $(7.27,48.5,1000)$. It was slightly less convenient to compute the updated parameters using the absolute-value loss function, and it did not matter much: $(7.33,48.4,1000)$. The sample of $x = 8$ had the most effect on the minimum parameter and almost no effect on the maximum parameter.

References

- https://en.wikipedia.org/wiki/Beta_distribution
- https://en.wikipedia.org/wiki/PERT_distribution
- https://en.wikipedia.org/wiki/Triangular_distribution
- *Probability and Statistical Inference* 9th edition by Robert V. Hogg, Elliot A. Tanis, Dale L. Zimmerman ©2015 by Prentice Hall.
- R Core Team (2019). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.
- R. Pouillot, M.-L. Delignette-Muller (2010), Evaluating variability and uncertainty in microbial quantitative risk assessment using two R packages. International Journal of Food Microbiology. 142(3):330-40
- Balasubramanian Narasimhan, Steven G. Johnson, Thomas Hahn, Annie Bouvier, and Kiên Kiêu (2018). cubature: Adaptive Multivariate Integration over Hypercubes. R package version 2.0.3. https://CRAN.R-project.org/package=cubature
- Hans W. Borchers (2018). pracma: Practical Numerical Math Functions. R package version 2.2.2. https://CRAN.R-project.org/package=pracma

Biographical Sketch

Michael Lloyd graduated cum laude and in the honors program in Chemical Engineering with a B.S. in 1984. He accepted a position at Henderson State University in 1993 shortly after earning his Ph.D. in Mathematics (Probability Theory) from Kansas State University. He has presented papers at meetings of the Academy of Economics and Finance, the American Mathematical Society, the Arkansas Conference on Teaching, and the Southwest Arkansas Council of Teachers of Mathematics. He has been an active member of the Mathematical Association of America since 1993, earned 18 hours in computer science, and has been an Advanced Placement statistics consultant since 2002.