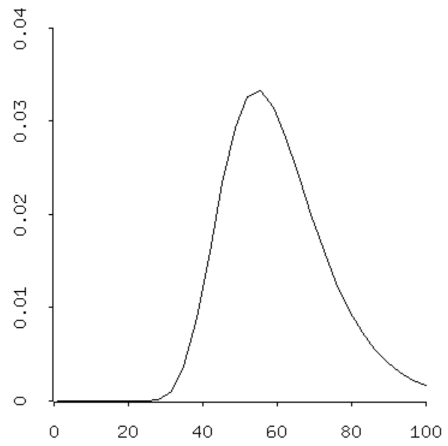
An example of running XLISPSTAT and finding the Laplace approximation for the marginal of θ_0. To start up xlispstat, run `xlispstat`

```
XLISP-PLUS version 2.1g
Portions Copyright (c) 1988, by David Betz.
Modified by Thomas Almy and others.
XLISP-STAT 2.1 Release 3.44 (Beta).
Copyright (c) 1989-1994, by Luke Tierney.


> (def Y  (list 65  156 100 134 16 108 121 4 39 143 56 26 22 1 1 5 65))

Y
> (def wbc (list 2300 750 4300 2600 6000 10500 10000 17000 5400 7000 9400
 32000 35000 100000 100000 52000 100000))

WBC

> (def  x (log (/ wbc 10000)))

X
> (defun loglik (theta)
  (let* ((theta0 (first theta))
 (theta1 (second theta))
 (t1x (* theta1 x)))
    (- (sum t1x)
       (* (length x) (log theta0))
       (/ (sum (* y (exp t1x)))
  theta0))))

LOGLIK
> (defun logprior (theta)
   (let ((theta0 (first theta))
 (theta1 (second theta)))
     (+ (log (* (/ 1 5) (exp (- (/ theta1 5)))))
(log (* (/ 1 theta0)
(exp (- (/ (^ (/ (- (log theta0) (log 52)) (log 2))  2) 2))))
     ))
     )
)

LOGPRIOR
> (defun logpost (theta) (+ (loglik theta) (logprior theta)))

LOGPOST
> (def post.obj (bayes-model #'logpost (list (exp 3.5) .8)))
; loading /usr/local/pkg/xlisp-3.50/lib/xlispstat/Autoload/bayes.fsl
; loading /usr/local/pkg/xlisp-3.50/lib/xlispstat/Autoload/maximize.fsl
maximizing...
Iteration 0.
```

```
Criterion value = -96.3475
Iteration 1.
Criterion value = -90.9338
Iteration 2.
Criterion value = -89.7458
Iteration 3.
Criterion value = -89.5976
Iteration 4.
Criterion value = -89.5928
Iteration 5.
Criterion value = -89.5928
Reason for termination: gradient size is less than gradient tolerance.


First Order Approximations to Posterior Moments:

Parameter 0                53.3812      (12.0091)
Parameter 1               0.468645      (0.174445)

POST.OBJ

> (send post.obj :moments)
((59.913015156807155 0.4691225085410134) (14.20681432090045 0.17930122572084103))
>
> (send post.obj :help :margin1)
loading in help file information - this will take a minute ...done
MARGIN1
Args: (g x &key (:derivstep .05) (spline t) profile)
Computes Laplace approximation to marginal posterior density of G at
points X. G can be an index or a function of the parameter vector. X
is a sequence that should include the modal value of G. If SPLINE is
true the log density is splined. If PROFILE is true, a profile of the
posterior is returned.
NIL

> (plot-lines (send post.obj :margin1 0 (rseq 1 100 100)))
```



```
> (def xy (transpose (send post.obj :margin1 0 (rseq 1 100 100))))
XY

> (dotimes (i 30) (apply #'format t "~g ~g ~%" (select xy i)))
1. 0.0
4.413793103448278 1.0884109175295808E-71
7.827586206896548 1.9357241861781495E-34
11.241379310344826 1.4724903396758523E-20
14.655172413793103 1.1832655713685282E-13
18.06896551724138 1.1075373802136736E-9
21.48275862068965 3.366876376572895E-7
24.896551724137932 1.4519358600559306E-5
28.310344827586206 1.8773721406910554E-4
31.72413793103448 1.1051526433276887E-3
35.137931034482754 3.8032444867512356E-3
38.55172413793103 8.961137916621675E-3
41.9655172413793 1.6043746735634846E-2
45.37931034482758 2.3436529190674255E-2
48.79310344827586 2.9366681853140706E-2
52.20689655172413 3.2722136081187084E-2
55.62068965517241 3.329373963891499E-2
59.03448275862069 3.155217887925768E-2
62.44827586206896 2.8274143708724254E-2
65.86206896551722 2.4237376834580417E-2
69.27586206896551 2.0056195374261548E-2
72.68965517241379 1.6135184087119243E-2
76.10344827586206 1.2691771283820525E-2
79.51724137931033 9.805394984903794E-3
82.93103448275861 7.46781731603752E-3
86.34482758620689 5.62341911371257E-3
89.75862068965516 4.197019866780534E-3
93.17241379310343 3.1108703456297986E-3
96.58620689655172 2.2937048412269485E-3
100. 1.6846071494067726E-3
```