Working Paper 92-39
Statistics and Econometrics Series 27
October 1992

División de Economía
Universidad Carlos III de Madrid
Calle Madrid, 126
28903 Getafe
Fax: (341) 624-9849

BAYESIAN ECONOMETRICS: CONJUGATE ANALYSIS AND REJECTION SAMPLING USING *MATHEMATICA*

Eduardo Ley and Mark F.J. Steel*

Abstract

Mathematica is a powerful "system for doing mathematics by computer" which runs on personal computers (Macs and MS-DOS machines), workstations and mainframes. Here we show how Bayesian methods can be implemented in *Mathematica*. One of the drawbacks of Bayesian techniques is that they are computation-intensive, and every computation is a little different. Since *Mathematica* is so flexible, it can easily be adapted to solving a number of different Bayesian estimation problems. We illustrate the use of *Mathematica* functions (i) in a traditional conjugate analysis of the linear regression model and (ii) in a completely nonstandard model -where rejection sampling is used to sample from the posterior.

Key words:
Bayes, Rejection Sampling, *Mathematica*.

*Ley, Departamento de Economía, Universidad Carlos III de Madrid; Steel, Departamento de Estadística y Econometría, Universidad Carlos III de Madrid.

Preliminary

Bayesian Econometrics: Conjugate Analysis and Rejection Sampling using *Mathematica*

Eduardo Ley
Mark F.J. Steel
Universidad Carlos III de Madrid

July, 1992
Version: September 21, 1992

Abstract. *Mathematica* is a powerful "system for doing mathematics by computer" which runs on personal computers (Macs and MS-DOS machines), workstations and mainframes. Here we show how Bayesian methods can be implemented in *Mathematica*. One of the drawbacks of Bayesian techniques is that they are computation-intensive, and every computation is a little different. Since *Mathematica* is so flexible, it can easily be adapted to solving a number of different Bayesian estimation problems. We illustrate the use of *Mathematica* functions *(i)* in a traditional conjugate analysis of the linear regression model and *(ii)* in a completely nonstandard model—where rejection sampling is used to sample from the posterior.

Keywords. Bayes, Rejection Sampling, *Mathematica*

Address. Eduardo Ley, Departamento de Economía, Universidad Carlos III, calle Madrid 126, 28903–Getafe, Madrid. Email: `edley@eco.uc3m.es`. Mark Steel, Departamento de Estadística y Econometría, Universidad Carlos III, calle Madrid 126, 28903–Getafe, Madrid. Email: `steel@eco.uc3m.es`.

Bayesian Econometrics:
Conjugate Analysis and Rejection Sampling
using *Mathematica*

Eduardo Ley
Mark F.J. Steel

```
In[1]:= SetDirectory["calvin:Mma:MmaEconBook:Bayesian:BayesE:"];
        <<BayesE.m
```

1. Introduction

In real-world problems we are invariably faced with making decisions in an environment of uncertainty (see also the essay by R. Korsan in this volume). A statistical paradigm then becomes essential for extracting information from observed data and using this to improve our knowledge about the world (inference), and thus guiding us in the decision problem at hand. The underlying probability interpretation for a Bayesian is a subjective one, referring to a personal degree of belief. The rules of probability calculus are used to examine how prior beliefs are transformed to posterior beliefs by incorporating data information. The sampling model is a "window" [see Poirier (1988)] through which the researcher views the world. Here we only consider cases where such a model is parameterized by a parameter vector θ of finite dimension. A Bayesian then focuses on the inference on θ (treated as a random variable) given the observed data Y (fixed), summarized in the posterior density $p(\theta|Y)$. The observations in Y define a mapping from the prior $p(\theta)$ into $p(\theta|Y)$. This posterior distribution can also be used to integrate out the parameters when we are interested in forecasting future values, say, $\tilde{Y}$, leading to the post-sample predictive density $p(\tilde{Y}|Y) = \int p(\tilde{Y}|Y,\theta)p(\theta|Y)d\theta$ where $p(\tilde{Y}|Y,\theta)$ is obtained from the sampling model.

In Economics as in many other disciplines, the Bayesian paradigm seems ideally suited to address the questions typically arising in applied work. Whereas statisticians are often mainly attracted by the theoretical coherence and mathematical elegance of the Bayesian approach, resulting in quotes like:

> "It is difficult for me to tone down the missionary zeal acquired in youth, but perhaps the good battle is justified since there are still many heathens."[I.J. Good (1976)]

> "Or perhaps most statisticians now *are* Bayesians (when it matters!), but they do not want to spoil the fun by admitting it."[A.F.M. Smith (1986)]

> "Every statistician would be a Bayesian if he took the trouble to read the literature thoroughly and was honest enough to admit he might have been wrong."[D.V. Lindley (1986)].

econometricians have primarily stressed the practical advantages of using the Bayesian paradigm, as opposed to the classical or sampling-theoretical methodology:

> "In other words, the traditional approach lacks flexibility in that it relies upon *too few* a priori restrictions that are, however, *too strict.*"[J.H. Drèze (1962)]

> "Necessarily, practicing economists have discarded the formal constraints of classical inference."[E.E. Leamer (1978)].

In particular, the Bayesian approach provides a formal way of incorporating any information we may have prior to observing the data, it fits perfectly with sequential learning and decision theory, and directly leads to exact small sample results. In addition, it naturally gives rise to predictive densities, where all parameters are integrated out, a perfect tool for missing data or forecasting. Of course, all this comes at a cost, which is typically of a computational nature. An analytical solution to the computational problem is provided by restricting ourselves to natural-conjugate prior densities

---

This is a preliminary draft of a chapter to appear in Hal Varian (editor), *Economic and Financial Modeling with Mathematica*, New York: Springer-Verlag, forthcoming. Some of the functions used here are standard *Mathematica* functions, while some others are implemented in the package **BayesE.m** developed by the authors and listed in Appendix B (available in computer media in the diskette accompanying the book, and from the authors upon request). All the computations shown in this paper were performed in a Macintosh SE/30 with 17Mb of RAM.

to summarize the prior information [see Raiffa and Schlaifer (1961) or Zellner (1971)]. A natural-conjugate prior shares the functional form of the likelihood and in exponential families this leads to posterior densities of the same form. The first part of this essay will consider this analysis in a Normal lineal regression model for health expenditure in OECD countries [see Newhouse (1977)].

A much more flexible tool is the rejection sampling method discussed in Smith and Gelfand (1992), which only requires that the maximum value of the sampling density

$$M = \max_{\theta} p(Y|\theta)$$

is finite. If we can then draw from the prior $p(\theta)$, we can generate drawings from the posterior $p(\theta|Y)$ simply by rejection. The second part of this essay will treat this method in some detail and illustrate it with an application to demand analysis.

2. Conjugate Analysis

The sampling model that we will focus on here is the linear regression model

$$y = X\beta + \varepsilon$$

where the observations $y' = (y_1, y_2, \ldots, y_T)$ are related to the k explanatory variables in $X = (x_1, x_2, \ldots, x_k)$, which are assumed to be strongly exogenous in the sense of Engle *et al.* (1983)—for a Bayesian treatment of exogeneity see Florens and Mouchart (1985) and Osiewalski and Steel (1990). The error vector ε is assumed to have a Normal distribution with mean zero and covariance matrix $\sigma^2 I_T$ $(\sigma^2 > 0)$.

In this application we are interested in the elasticity of the per capita health expenditure with respect to income (GDP) [Newhouse (1977)]. We use cross-country data on 24 OECD countries (the data that we use appeared in the *Health Care Financing Review* 1989 Annual Supplement). We explain per capita health expenditure, y, by GDP per capita, x_2, and the share of the public sector in total health expenses, x_3. Purchasing power parities are used to convert all magnitudes to common units. All variables are in natural logs and a constant, x_1 is included. Sample size is $T = 24$, and $k = 3$.

2.1. Read In the Data and Define Variables

We read the data corresponding to 1986 using standard *Mathematica* procedures.[1] We use the **SetDelayed** (':=') function instead of the **Set** ('=') function so that all the variables are automatically updated to their 1987 values when we later read the 1987 data into **data**.

```
In[2]:= data = ReadList[ "health86.dat", Table[Number, {5}] ];
        tothealthexp := Column[data, 1];
        pubhealthexp := Column[data, 2];
        population   := Column[data, 3];
        GDP          := Column[data, 4];
        PPP          := Column[data, 5];
```

Transformations are easily achieved using standard *Mathematica* functions,

```
In[3]:= y  := Log[tothealthexp/(population*PPP)];
        x2 := Log[GDP/(population*PPP)];
        x3 := Log[(pubhealthexp/tothealthexp)*100];
        X  := Table[{1, x2[[i]], x3[[i]]}, {i, 1, Length[data]}];
```

2.2. Bayesian Regression Analysis: Diffuse Prior

The parameter set in this problem is $\beta = (\beta_1, \beta_2, \beta_3)'$ and σ. We shall first assume that we possess no prior information at all which is translated into the improper prior density

$$p(\beta, \sigma) = p(\beta)p(\sigma) \propto \sigma^{-1}.$$

The latter prior density is obtained by applying Jeffreys' rule [see Jeffreys (1961) and Zellner (1971)] under prior independence of β and σ. Calling **BayesRegression[*ydata*, *Xdata*]** will return the posterior densities for σ and β into the Mathematica variables **postdistsigma** and **postdistbeta**.

```
In[4]:= BayesRegression[y, X]
```

[1] All data sets used in this paper are listed in Appendix B.

• *Sigma*

As this diffuse prior can, in fact, be considered as a limiting case of a natural-conjugate prior, the posterior analysis becomes quite straightforward; in particular, the posterior density of σ will be of the inverted gamma form (see Appendix A):

$$p(\sigma|y, X) = f_{i\gamma}(\sigma|T-k, s),$$

where $s^2 = y'M_x y/(T-k)$ and $M_x = I_T - X(X'X)^{-1}X'$. Typing **postdistsigma** we obtain σ's posterior density:[2]

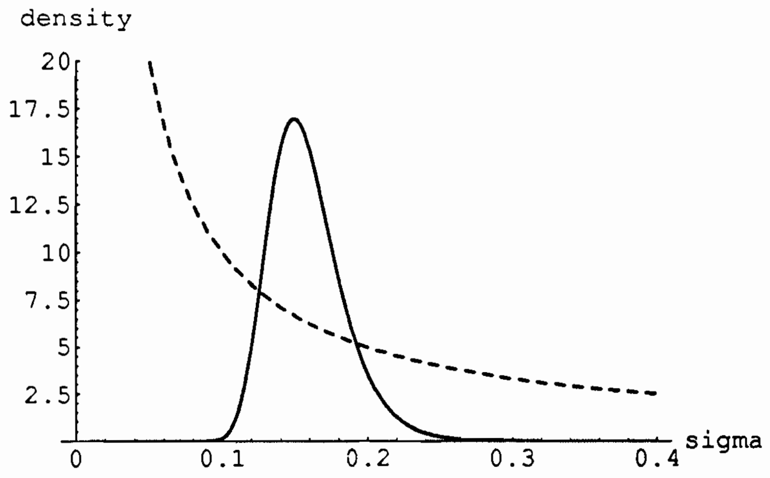
```
In[5]:= postdistsigma

Out[5]= IGammaDistribution[21, 0.152788]
```

Moments of this density are known analytically and can be readily computed:

```
In[6]:= Mean[postdistsigma]

Out[6]= 0.15853

In[7]:= Variance[postdistsigma]

Out[7]= 0.000669825
```

A plot shows clearly how the prior information (with arbitrary scaling), the dashed curve, is modified substantially through the observations.

```
In[8]:= Plot[{PDF[postdistsigma,s], s^(-1)}, {s, 0.01, 0.4},
          PlotRange -> {0, 20}, AxesLabel -> {"sigma", "density"},
           PlotStyle -> {{}, {Dashing[{0.015, 0.015}]}}]

Out[8]=
```



• *Beta*

For the coefficients in β, the resulting marginal posterior density is of the multivariate Student t form (see Appendix A):

$$p(\beta|y, X) = f_t^k(\beta|\hat{\beta}, s^{-2}X'X, T-k),$$

[where $\hat{\beta}$ is the least-squares estimate, $\hat{\beta} = (X'X)^{-1}X'y$] which, again, has known moments, up to the order $T-k$.

```
In[9]:= postdistbeta

Out[9]= MultivariateTDistribution[{-3.40481, 1.45723, -0.077314},
                                   {{1028.09,  2466.91,  4423.01},
                                    {2466.91,  6046.67, 10644.6},
```

[2] The distributions that this package adds to the existing ones in **ContinuousStatisticalDistributions** are the **IGammaDistribution**, the (non-standard) Student **UnivariateTDistribution**, the Student **MultivariateTDistribution** and the **MultivariateNormalDistribution**.

{4423.01, 10644.6, 19077.9}}, 21]

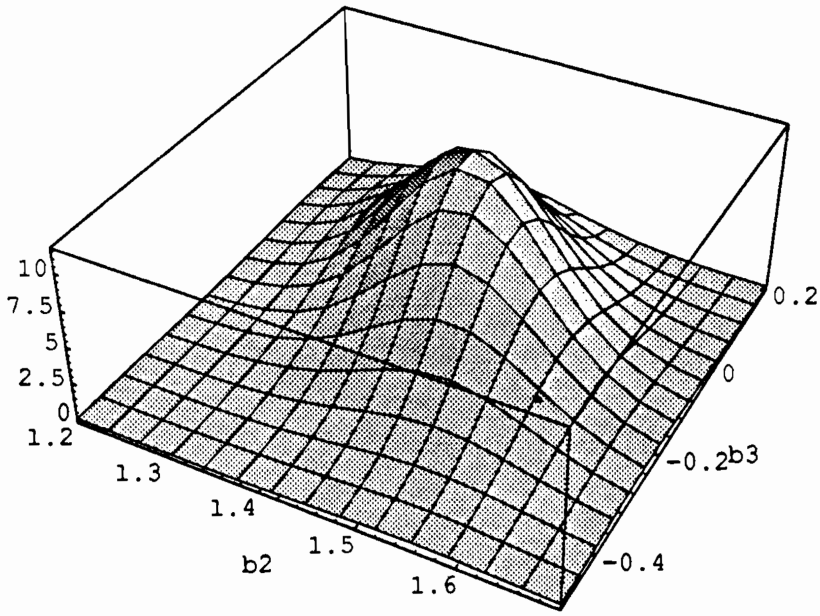
```
In[10]:= Mean[postdistbeta]

Out[10]= {-3.40481, 1.45723, -0.077314}

In[11]:= MatrixForm[Variance[postdistbeta]]

Out[11]=  0.415384      0.00355095    -0.0982834
          0.00355095    0.0103118     -0.00657677
         -0.0982834    -0.00657677     0.0265134
```

Marginal[*list*, *MultivariateDistribution*] returns the marginal distribution of the elements in *list*,

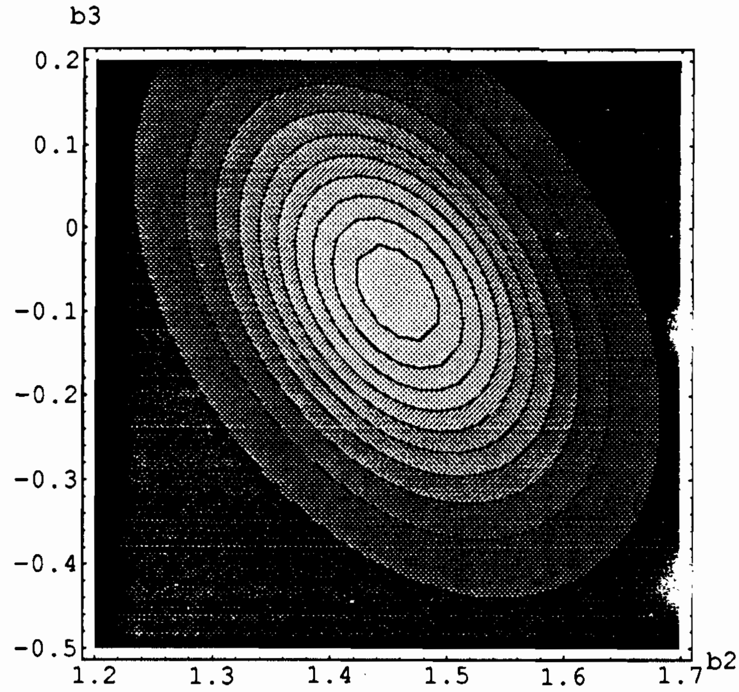
```
In[12]:= mar23 = Marginal[{2,3}, postdistbeta]

Out[12]= MultivariateTDistribution[{1.45723, -0.077314},
                        {{127.328, 31.5843}, {31.5843, 49.5216}}, 21]
```

We take advantage of *Mathematica*'s powerful plotting functions to visualize the posterior joint density of (β_2, β_3),

```
In[13]:= densityBetaDA =
         Plot3D[PDF[mar23, {b2, b3}], {b2, 1.2, 1.7}, {b3, -0.5, 0.2},
         PlotRange->All, AxesLabel -> {"b2   ", "   b3", ""}]

Out[13]=
```

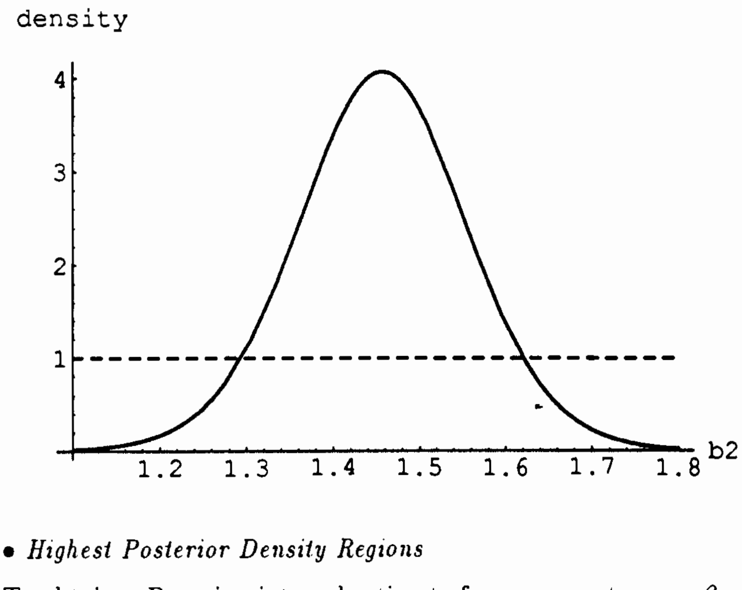



```
In[14]:= contourBetaDA =
         ContourPlot[PDF[mar23,{b2, b3}], {b2, 1.2, 1.7}, {b3, -0.5, 0.2}, PlotRange -> All,
         PlotPoints -> 30, Axes -> True, AxesLabel -> {"b2", "b3"}]

Out[14]=
```



Also, we can plot the prior (improper, therefore scaled arbitrarily) and posterior marginal densities of each of the β_i's in the same picture to graphically display the data information. For example, for β_2 we have

```
In[15]:= Plot[{PDF[Marginal[{2},postdistbeta], b2], 1}, {b2, 1.1, 1.8},
         AxesLabel -> {"b2", "density"}, PlotRange -> All,
         PlotStyle -> {{}, {Dashing[{0.015, 0.015}]}}]
Out[15]=
```



- *Highest Posterior Density Regions*

To obtain a Bayesian interval estimate for a parameter, say β_3, with a preassigned probability content q, say $q = 0.8$, we can use the *highest posterior density* (HPD) interval [see Judge *et al.* (1985)]. Since the marginal posterior density of β_3 is unimodal and symmetric, the interval will be symmetric around the posterior mean, say b_3, so we just need to find the z which solves,

$$\int_{b_3-z}^{b_3+z} p(\beta_3|y,X)d\beta_3 = 0.8$$

then,

```
In[16]:= m2 = Marginal[{2}, postdistbeta]

Out[16]= UnivariateTDistribution[1.45723, 107.184, 21]

In[17]:= q = .80;
         {Mean[m2] - z, Mean[m2] + z}/.FindRoot[
         Integrate[PDF[m2, b2], {b2, Mean[m2] - z, Mean[m2] + z}] - q, {z, 0.10}]

Out[17]= {1.32942, 1.58504}
```

Thus, there is a 80% posterior probability that β_3 belongs to the interval (1.33, 1.59). Note that the prior probability content of that interval (as of any other bounded interval) is zero due to the improper reference prior.

- *Probability of Subspaces*

The *Mathematica* function `NIntegrate` provides a straightforward method for computing the probabilities of certain regions in parameter space. Suppose that we are interested in the posterior probability that the elasticity of health expenditures with respect to income is greater than 1 ($\beta_2 > 1$) ***and*** that the elasticity with respect to the public sector participation in the health expenditures is negative ($\beta_3 < 0$). We can easily compute $Pr[\beta_2 > 1, \beta_3 < 0|y, X]$ integrating the joint density function of (β_2, β_3) over the appropiate range,

```
In[18]:= NIntegrate[PDF[mar23, {b2, b3}], {b2, 1, Infinity}, {b3, -Infinity, 0}]

Out[18]= 0.688572
```

so, the desired probability is 68.86%.

- *Prediction*

If we now wish to predict the per capita health expenses

$$\tilde{y}' = (\tilde{y}_{T+1}, \tilde{y}_{T+2}, \ldots, \tilde{y}_{T+n})$$

in other countries with explanatory variables $\tilde{X}$, we can simply use the post-sample predictive density

$$p(\tilde{y}|y, X, \tilde{X}) = f_t^n(\tilde{y}|\tilde{X}\hat{\beta}, s^{-2}(I_n + \tilde{X}(X'X)^{-1}\tilde{X}')^{-1}, T-k)$$

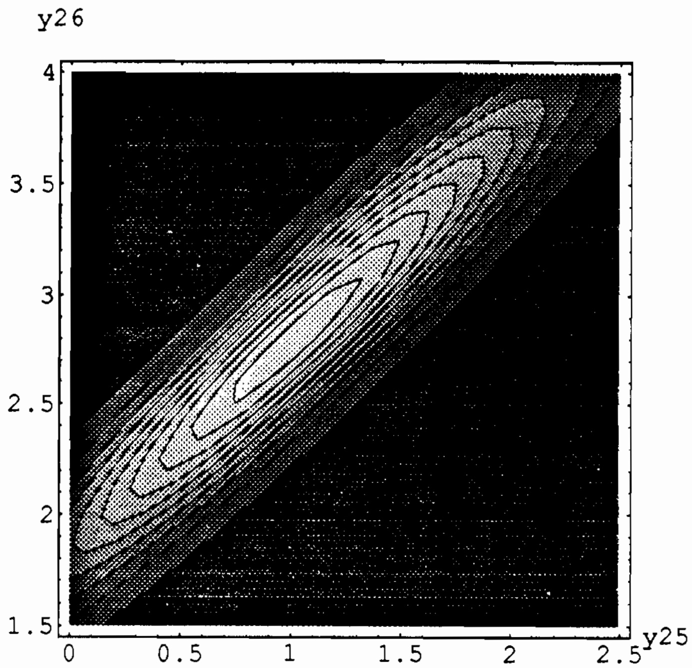
where all the parameters are integrated out using the posterior densities (i.e., incorporating the information on the actually observed countries).

If we take 2 unobserved countries (25, 26) with $\tilde{x}_2 = \log 20$, $\tilde{x}_3 = \log 0.5$ and $\tilde{x}_2 = \log 70$, $\tilde{x}_3 = \log 0.9$ respectively, we are led to the following post-sample predictive:

```
In[19]:= xf = {{1, Log[20.00], Log[.5]},
              {1, Log[70.00], Log[.9]}};
         BayesRegression[y, X, xf];
         preddisty

Out[19]= MultivariateTDistribution[{1.01424, 2.79436},
                                   {{16.4694, -16.2248}, {-16.2248, 17.5988}}, 21]
```

which can easily be plotted using the intrinsic *Mathematica* features.

```
In[20]:= ContourPlot[PDF[preddisty, {y25, y26}], {y25, 0, 2.5}, {y26, 1.5, 4},
                     PlotPoints -> 30, Axes -> True, AxesLabel -> {"y25", "y26"}]

Out[20]=
```



Note from the contour plot that y_{25} and y_{26} are not uncorrelated, even though they arise from an independent sampling process. Integrating out σ makes them dependent as it changes the Normal to a multivariate Student (where the elements can never be independent even if the correlation is zero) and integrating out the common uncertainty on β introduces correlation as well.

2.3. Natural Conjugate Analysis

When new data become available for 1987, we can use the posterior densities from the previous analysis as prior densities to study the behavior of the model. Reading the new data into **data** automatically updates the contents of all variables.

```
In[21]:= data = ReadList["health87.dat",Table[Number, {5}]];
```

Given that we are in a Normal linear context, the posterior distributions obtained from the previous diffuse analysis are natural conjugate priors.[3]

```
In[22]:= priordistbeta = postdistbeta;
         priordistsigma = postdistsigma;
         BayesRegression[y, X, priordistbeta, priordistsigma]
```

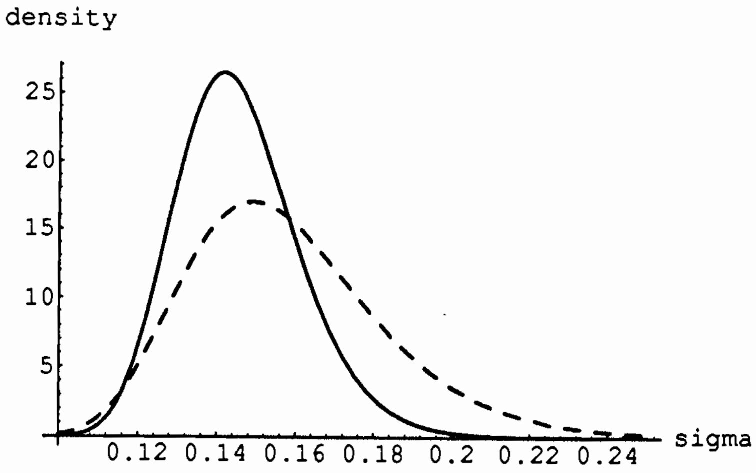
As before, **postdistsigma** contains the posterior distribution for σ,

```
In[23]:= postdistsigma

Out[23]= IGammaDistribution[45, 0.143222]
```

We can plot the prior (dashed line) and posterior densities in the same graph,

```
In[24]:= Plot[{PDF[postdistsigma,s], PDF[priordistsigma,s]}, {s, 0.10, 0.25},
              PlotRange -> All, AxesLabel -> {"sigma", "density"},
              PlotStyle -> {{},{Dashing[{0.025, 0.025}]}}]

Out[24]=
```

[3] When we want to use different prior information of a natural conjugate form, the function **BayesRegression** can alternatively be called with the hyperparameters of the prior distributions instead of the distributions themselves. Type **??BayesRegression** for details.



Similarly, we obtain the posterior distribution for β,

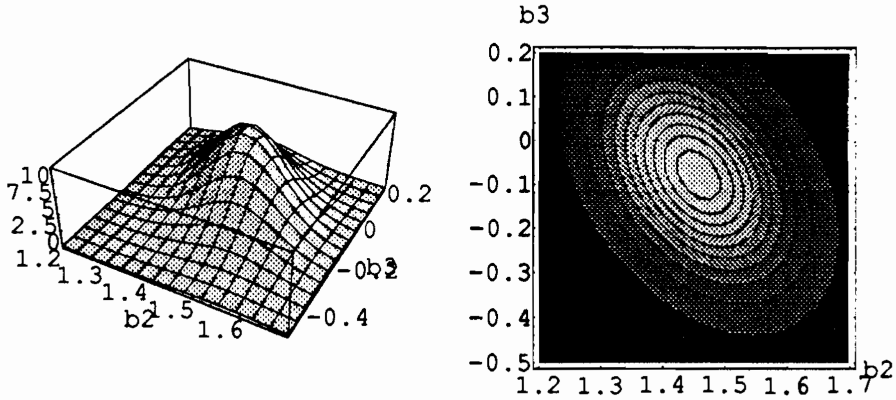
```
In[25]:= postdistbeta
```

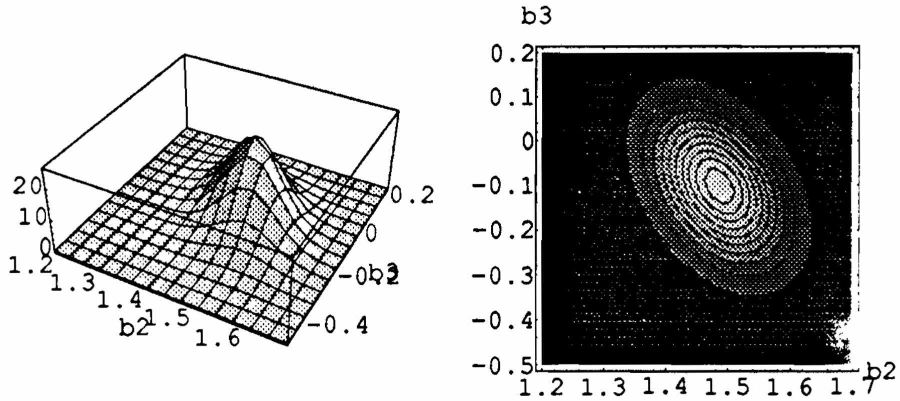
```
Out[25]= MultivariateTDistribution[{-3.37346, 1.48573, -0.102293},
                                   {{2340.04, 5677.02, 10087.1},
                                    {5677.02, 14062.8, 24542.9},
                                    {10087.1, 24542.9, 43590.8}}, 45]
```

The prior and posterior densities for β_2 and β_3 can be plotted side by side in order be more easily compared,

```
In[26]:= mar23 = Marginal[{2, 3}, postdistbeta];
         densityBetaNCA =
         Plot3D[PDF[mar23, {b2, b3}], {b2, 1.2, 1.7}, {b3, -0.5, 0.2},
         PlotRange -> All, AxesLabel -> {"b2   ", "   b3", ""}, DisplayFunction -> Identity];
         contourBetaNCA =
         ContourPlot[PDF[mar23, {b2, b3}], {b2, 1.2, 1.7}, {b3, -0.5, 0.2},
         PlotRange -> All, PlotPoints -> 30, Axes -> True, AxesLabel -> {"b2", "b3"},
         DisplayFunction -> Identity];
         Show[GraphicsArray[{{densityBetaDA, contourBetaDA}, {densityBetaNCA, contourBetaNCA}}],
         DisplayFunction -> $DisplayFunction]
```

```
Out[26]=
```



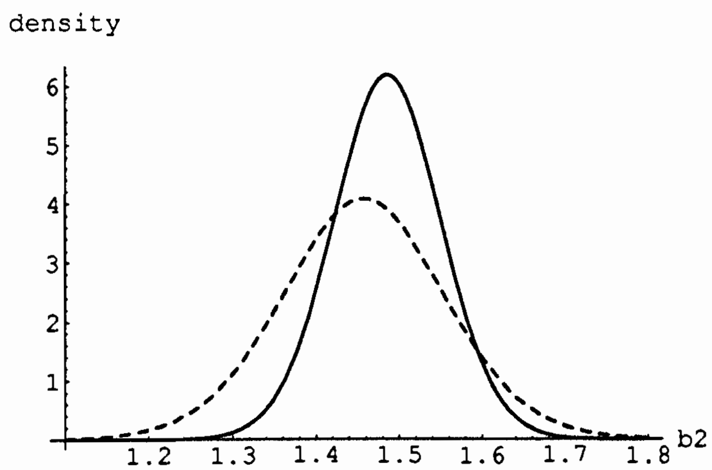


The new data information significantly concentrates the density around its mean. There is no real conflict of information between the data in both years.

Finally, we can plot the marginal posterior densities of any of the elements in β, for example β_2,

```
In[27]:= Plot[{PDF[Marginal[{2}, postdistbeta], b2], PDF[Marginal[{2}, priordistbeta], b2]},
        {b2, 1.1, 1.8}, PlotStyle -> {{}, {Dashing[{0.015, 0.015}]}},
        AxesLabel -> {"b2", "density"}, PlotRange -> All]
```

Out[27]=



3. A Rejection Sampling Approach to Demand Analysis

3.1. Introduction

The sampling model applies a suggestion of Varian (1990) to a demand system. Let the utility function be completely characterized by the functional form $u(\cdot)$ and the parameter α, $u = u(x;\alpha)$. If the *actual* expenditure (for observation t), $m_t = p_t x_t$, [where $p_t = (p_{1t}, p_{2t}, \ldots, p_{nt})$ is a row vector of prices, and $x_t = (x_{1t}, x_{2t}, \ldots, x_{nt})'$ is a column vector of a bundle of n goods] is compared to the *minimum* expenditure required to yield the same utility level, say, $e(p_t, x_t; \alpha)$, Ley (1992) proposes the model

$$\log m_t - \log e(p_t, x_t; \alpha) = \varepsilon_t$$

where $\varepsilon_t > 0$ has some distribution defined over the positive real line. With only three goods, in the simple Cobb-Douglas case where the utility function is

$$u(x;\alpha) = x_1^{\alpha_1} x_2^{\alpha_2} x_3^{1-\alpha_1-\alpha_2},$$

and with exponentially distributed error terms[4] ε_t (i.i.d. for $t = 1, 2, \ldots, T$),

$$p(\varepsilon_t|\mu) = f_\gamma(\varepsilon_t|1, \mu),$$

we obtain for all T observations:

$$p(\log m|e(p,x;\alpha),\mu) = \exp\Bigg\{-T\log\mu - \mu^{-1}\sum_{t=1}^{T}\Bigg[\log m_t - \alpha_1\log\left(\frac{p_{1t}x_{1t}}{\alpha_1}\right) - \alpha_2\log\left(\frac{p_{2t}x_{2t}}{\alpha_2}\right) - (1-\alpha_1-\alpha_2)\log\left(\frac{p_{3t}x_{3t}}{1-\alpha_1-\alpha_2}\right)\Bigg]\Bigg\}$$

where m, $e(p,x;\alpha)$, p and x are all straightforwardly extended to the case of T observations.

Clearly, this complicated likelihood function does not allow for a natural-conjugate analysis and a numerical approach would have to be followed. Here we adopt the rejection sampling technique, which can generally be used if we need to draw from a density $f(x)$ which is too complicated to draw from directly. If we can find a density $g(x)$ such that

$$M = \max_x \frac{f(x)}{g(x)} < \infty$$

and we can generate drawings from $g(x)$, then we can use the following algorithm [see Ripley (1986), Johnson (1987)]:

[S1] Generate x_d from $g(x)$ and compute $R_d = f(x_d)/Mg(x_d)$.

[S2] Generate u_d from a uniform on$(0,1)$.

[S3] If $u_d \leq R_d$ accept x_d; otherwise reject x_d.

The accepted drawings x_d will be distributed according to $f(x)$. As the probability of acceptance of x_d equals M^{-1}, we should attempt to make M not "too large". In practice, the function $f(x)$ will often only be the kernel of a density function, in which case the acceptance probability becomes

$$\frac{1}{M}\int f(x)dx$$

where the integral is over the support of x.

This general principle is applied to the prior-to-posterior mapping in Smith and Gelfand (1992). They observe that if we take the prior $p(\theta)$ as $g(\cdot)$, we can obtain drawings from the posterior with kernel $f(\cdot)$ provided that

$$M = \max_\theta p(Y|\theta)$$

is finite. The ratio R_d then becomes the ratio of the likelihood value at the drawn parameter vector and the maximum value of the likelihood, M.

[4] Note that an exponential distribution with mean μ is just a gamma distribution with shape parameter 1; see Appendix A for the form of the gamma distribution.

3.2. Applied Demand Analysis

We use the data in Varian (1990) which consists in U.S. aggregate consumption data of three groups of goods: durables, nondurables and services from 1947 to 1987 . We have $n = 3$ and $T = 41$. We read the data and define the variables,

```
In[1]:= data = ReadList["cons.dat", Table[Number, {6}]];
        Do[ p[i] = Column[data, i], {i, 1, 3} ];
        Do[ x[i] = Column[data, i+3], {i, 1, 3} ];
        Do[ m[i] = p[i]*x[i], {i, 1, 3} ];
        M = Sum[ m[i], {i, 1, 3} ];
```

- *ML Estimates via Calculus*

Ley (1992) shows how in this case the MLE have simple closed-form analytical solutions (otherwise, we could use the `FindMinimum` function). In particular, the first-order conditions imply that

$$\frac{\hat{\alpha}_i}{\hat{\alpha}_j} = \left(\prod_{t=1}^{T} \frac{p_{it}x_{it}}{p_{jt}x_{jt}}\right)^{1/T}$$

for $i, j = 1, 2, 3$. We can use *Mathematica* to find the expressions for the MLE's,

```
In[2]:= eq1 = a1/(1-a1-a2) - pr[1]/pr[3];
        eq2 = a2/(1-a1-a2) - pr[2]/pr[3];
        mlesol = Flatten[Simplify[Solve[{eq1 == 0 , eq2 == 0}, {a1, a2}]]]
Out[2]=
                    pr[1]                          pr[2]
        {a1 -> ---------------------, a2 -> ---------------------}
               pr[1] + pr[2] + pr[3]        pr[1] + pr[2] + pr[3]
```

We define the auxiliary variables,

```
In[3]:= Do[ pr[i] = (Product[m[i][[t]], {t, 1, 41}])^(1/41), {i, 1, 3} ];
```

The MLE for $(\alpha_1, \alpha_2, \alpha_3)$ are obtained by simply making the appropiate substitutions

```
In[4]:= a[1] = a1/.mlesol
Out[4]= 0.153783

In[5]:= a[2] = a2/.mlesol
Out[5]= 0.463645

In[6]:= a[3] = 1 - a[1] - a[2]
Out[6]= 0.382572
```

Next, we compute the MLE of the mean error, μ, which is given by the average of the logs of the ratios of observed to minimum expenditure,

```
In[7]:= Do[e[t] = Product[(m[i][[t]]/a[i])^a[i], {i,1,3}], {t,1,41}];
        mu = (1/41)Sum[Log[M[[t]]/e[t]],{t,1,41}]
Out[7]= 0.0195704
```

We store in `mle` the value of the likelihood function evaluated at the maximum likelihood estimates,

```
In[8]:= mle = ((1/mu)^41)*Exp[-41]
Out[8]=
                  70
        1.10756 10
        ------------
             41
            E
```

• *Rejection Sampling*

We postulate a Dirichlet distribution (see Appendix A) with parameter vector $\boldsymbol{k}$ for the prior distribution of α. We write a simple procedure to generate Dirichlet variables based on Devroye (1986), p. 594. First, independent gamma variables (with shape parameters given by the elements in $\boldsymbol{k}$ and scale parameters 1) are generated and, then, they are normalized by their sum.

```
In[9]:= RandomDirichlet[k_] := Module[{a,b},
                               a = Table[Random[GammaDistribution[k[[i]],1]], {i, 1, Length[k]}];
                               b = Apply[Plus, a];
                               a/b];
```

We choose a gamma prior for μ^{-1} [see van den Broeck *et al.* (1992), who also suggest a convenient elicitation process].

The following instructions implement the rejection algorithm. The accepted drawings are sent to different files in prevention of an interruption of the execution since the process can take a long time to finish (*e.g.*, about three days in a DECstation 5000). The acceptance ratio in this example was, approximately, 1:100, so close to 200,000 drawings from the prior were needed in order to achieve the target of 2,000 samples from the posterior distribution. Of course, more efficient generators can be coded either inside or outside *Mathematica* [e.g., using the algorithms in Devroye (1986)] but we wanted to illustrate the method using standard *Mathematica* functions.

```
In[10]:= accepteddrawings = 0;
         totnumberdrawings = 0;
         While[accepteddrawings < 2000,
         u  = Random[Real, {0,1}];
         da = RandomDirichlet[{30,90,80}];
         dm = 1/Random[GammaDistribution[1, -1/Log[.98]]];
         ratio = ((1/dm)^41)*Exp[-(1/dm)*
         Sum[Log[M[[t]]/((m[1][[t]]/da[[1]])^da[[1]]*
           (m[2][[t]]/da[[2]])^da[[2]]*
           (m[3][[t]]/da[[3]])^da[[3]])],{t,1,41}]]/mle;
         If[u <= ratio, (da>>>palpha.dat; dm>>>pmu.dat; accepteddrawings++;)];
         totnumberdrawings++;
         ]
```

We generate a uniform variable u, then we draw the vector of α's from a Dirichlet with parameters $(30, 90, 80)$ which are elicited postulating mean expenditure shares of 15%, 45%, and 40% with standard deviations of 2.5%, 3.5% and 3.5% for x_1, x_2 and x_3 respectively. The inverse μ's are drawn from a gamma distribution with shape parameter 1 and scale parameter $-1/\log 0.98$; the latter is elicited through specifying the prior median efficiency as 0.98 [see van den Broeck *et al.* (1992)]. Then, **ratio** is evaluated and compared to u in order to decide whether to accept the drawing or not. The files **palpha.dat** and **pmu.dat** will contain the accepted drawings. The variable **totnumberofdrawings** was 176,040 in our run, so the acceptance percentage was 1.14%.

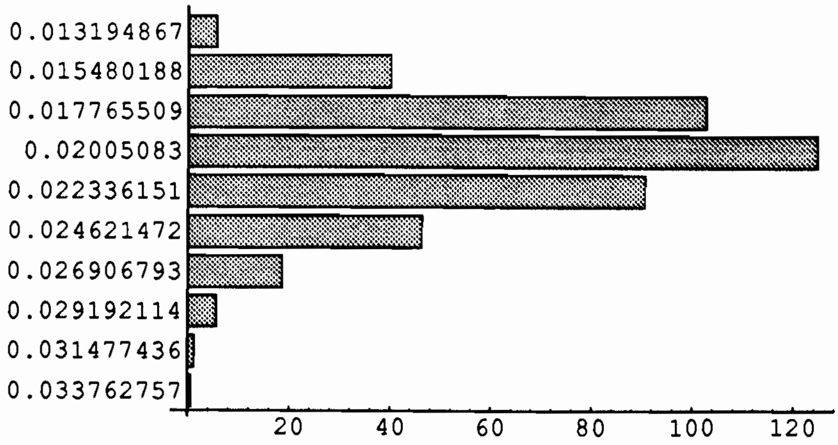
We read back into *Mathematica* the accepted drawings for μ and the α's,

```
In[11]:= pmu = ReadList["pmu.dat"];
         palpha  = ReadList["palpha.dat"];
         palpha1 = Column[palpha, 1];
         palpha2 = Column[palpha, 2];
         palpha3 = Column[palpha, 3];
```

• *Mu*

We plot a histogram of the accepted drawings of μ to visualize its posterior distribution

```
In[12]:= HHistogram[pmu]
Out[12]=
```



The **BayesE** function **HHistogram** returns a horizontal histogram with the default number of bins 10. This function was implemented because **VHistogram** doesn't allow to display too many decimals in the bin's coordinates without lumping the numbers together. Other functions in **BayesE** that produce histograms are **VHistogram** and **SmoothedVHistogram** which are illustrated below.

We can use the procedures in the package **DescriptiveStatistics** to analyze the sampled μ's,

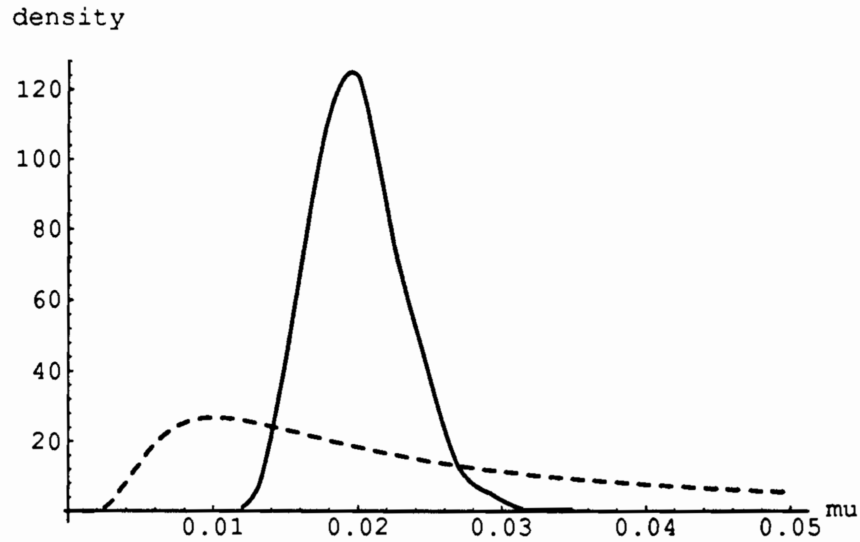
```
In[13]:= LocationReport[pmu]

Out[13]= {Mean -> 0.0204403, HarmonicMean -> 0.0199447, Median -> 0.0202037}
```

```
In[14]:= DispersionReport[pmu]

Out[14]= {Variance -> 0.0000104647, StandardDeviation -> 0.00323493,
          SampleRange -> 0.0228532, MeanDeviation -> 0.00254889,
          MedianDeviation -> 0.00207132, QuartileDeviation -> 0.00207876}
```

We could compare the prior and posterior density function of μ by plotting the PDF that we used to draw from (dashed) and the histogram of the accepted draws,[5]

```
In[15]:= mupdf[mu_] :=  -Log[0.98]*Exp[Log[0.98]/mu] / mu^2;
         graph1 = Plot[mupdf[mu], {mu, 0.001, 0.05},
         PlotRange -> All, PlotStyle -> {Dashing[{0.015, 0.015}]},
         DisplayFunction -> Identity];
         graph2 = SmoothedVHistogram[pmu, DisplayFunction -> Identity];
         Show[graph1, graph2, AxesLabel -> {"mu", "density"},
         DisplayFunction -> $DisplayFunction ]

Out[15]=
```

[5] If μ^{-1} is distributed as $f_\gamma(\mu^{-1}|1, -1/\log .98)$, making a change of variable we obtain the distribution of μ (which is an inverted gamma), $p(\mu) = -\log .98 \exp\{\log .98/\mu\}/\mu^2$.



The previous plot clearly shows how the data information is incorporated into μ's posterior density which concentrates a lot of mass around its mean, 0.02.

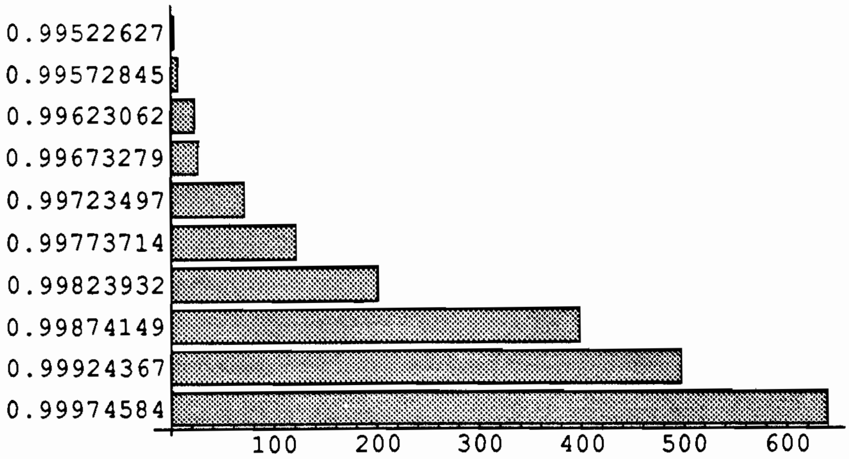
- *Efficiency*

We are probably more interested in the posterior distribution of the efficiency measure given by

$$\frac{e(p_t, x_t; \alpha)}{m_t} = e^{-\varepsilon_t} \in (0, 1]$$

which is a parametric generalization of Afriat's efficiency index [Varian (1990)]. We can easily look at the posterior density of this efficiency measure by simply evaluating the ratio of $e(p_t, x_t; \alpha)$ and m_t for a given year t (observed) at every drawn value from the posterior of α. If we take, *e.g.*, the t associated to the median m_t which corresponds to the year 1967, $t = 21$), we calculate

```
In[16]:= efficiency = Table[((m[1][[21]]/palpha1[[i]])^palpha1[[i]] *
                             (m[2][[21]]/palpha2[[i]])^palpha2[[i]] *
                             (m[3][[21]]/palpha3[[i]])^palpha3[[i]])/ M[[21]], {i, 1, Length[palpha1]}];
         HHistogram[efficiency]
Out[16]=
```

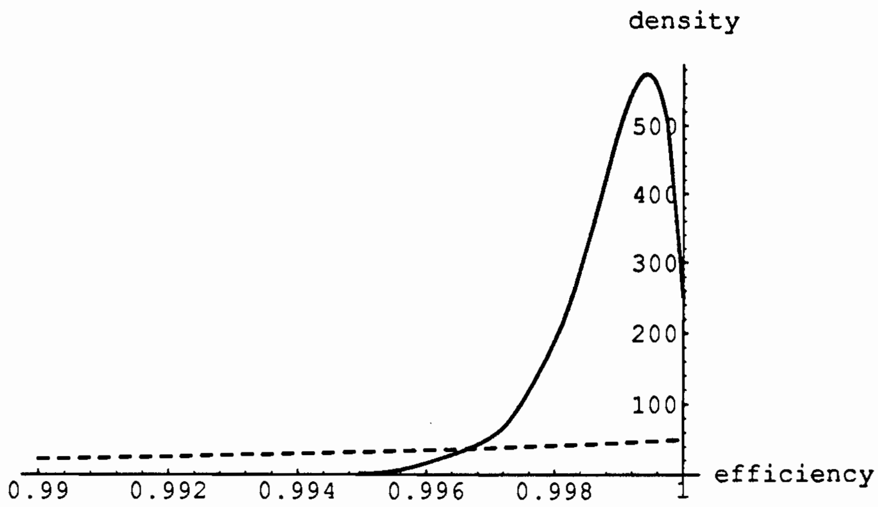


The corresponding prior efficiency can be found by integrating out μ from the implied joint density of (ε_t, μ) which leads to the following marginal prior density for $r_t = e^{-\varepsilon_t}$:

$$p(r_t) = \frac{-1}{r_t \log .98} \left(1 + \frac{\log r_t}{\log .98}\right)^{-2}$$

for all years. This is the dashed line in the graph,[6]

```
In[17]:= effpdf[r_] := -1/(r*Log[0.98])(1+(Log[r]/Log[0.98]))^(-2);
         graph1 = Plot[effpdf[r], {r, 0.99, 1}, DisplayFunction -> Identity,
         PlotRange -> All, PlotStyle -> {Dashing[{0.015, 0.015}]}];
         graph2 = SmoothedVHistogram[efficiency, DisplayFunction -> Identity];
         Show[graph1, graph2, AxesLabel -> {"efficiency", "density"},
         AxesOrigin -> {1, 0}, DisplayFunction -> $DisplayFunction ]
Out[17]=
```



As before, since **efficiency** is just a **List** of numbers, we can use any of the functions in **DescriptiveStatistics** to analyze it:

```
In[18]:= LocationReport[efficiency]
Out[18]= {Mean -> 0.998964, HarmonicMean -> 0.998963, Median -> 0.999167}
In[19]:= DispersionReport[efficiency]
Out[19]=
                                  -7
        {Variance -> 6.98323 10   , StandardDeviation -> 0.000835657,
         SampleRange -> 0.00502174, MeanDeviation -> 0.000651993,
         MedianDeviation -> 0.000504248, QuartileDeviation -> 0.000525945}
```

- *Alphas*

We use the auxiliary function **ListOfBinCounts** to get a rectangular list of the counts for each of the (default) 10×10 bins for the drawn α_1's and α_2's which we can use later to get a 3D histogram or a contour plot.[7]

```
In[20]:= palpha1and2 = Transpose[ColumnJoin[{Take[palpha1, 1000]}, {Take[palpha2, 1000]}]];
         bc = ListOfBinCounts[palpha1and2];
```

The auxiliary function **ListOfBinCoordinates** will return the coordinates at which the bins are centered. These can be used in conjuction with **ListContourPlot** and **BarChart3D** whose coordinates are just the bins' numbers.

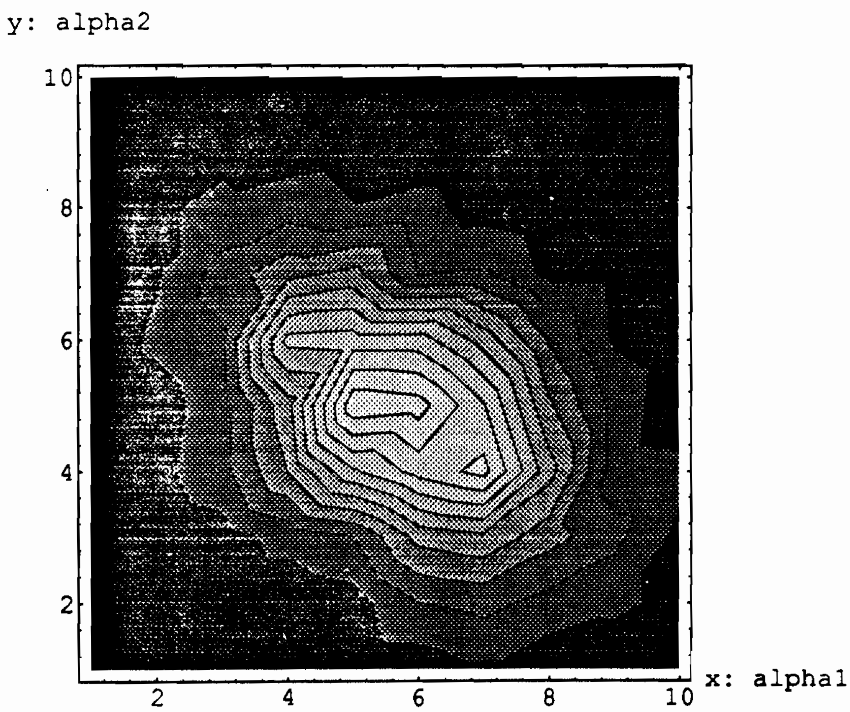
```
In[21]:= ListOfBinCoordinates[palpha1and2]
Out[21]=
```

[6] The package's function **SmoothedVHistogram** uses *Mathematica*'s function **Interpolation** to fit a third-order polynomial to the histogram's heights, and then **Plot** to plot this polynomial. Any of the valid **Plot** options can be specified; as an example, we set labels for the axes and set the origin at (1, 0).

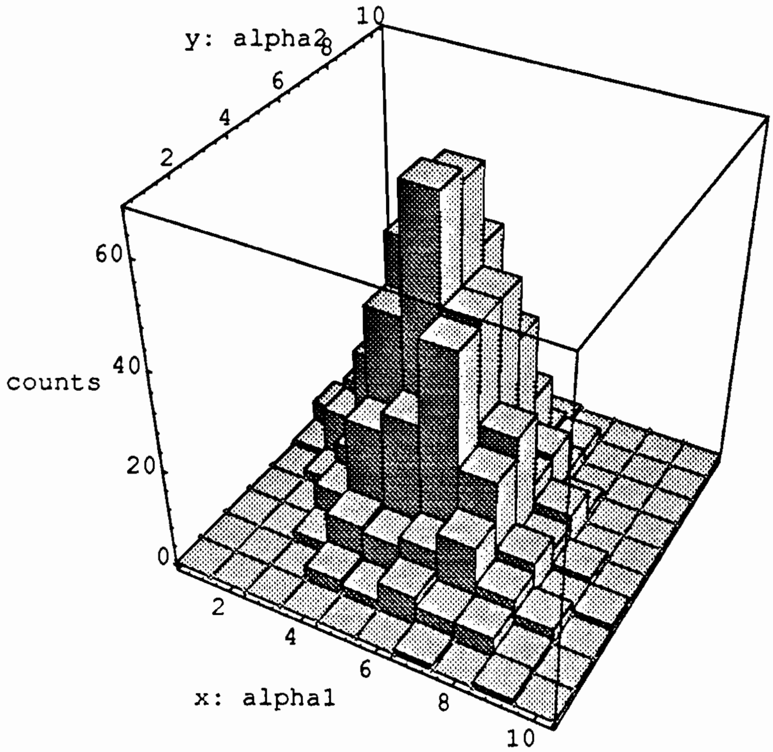
[7] Actually, only 1,000 values of the drawn α' s are used (notice the '**Take**' command) since **ListOfBinCounts** might crash in some systems if more values are read.

	x coord.	y coord.
1	0.1303	0.4294
2	0.1355	0.4356
3	0.1408	0.4417
4	0.146	0.4479
5	0.1512	0.4540
6	0.1564	0.4602
7	0.1617	0.4664
8	0.1669	0.4725
9	0.1721	0.4787
10	0.1774	0.4848

```
In[22]:= ListContourPlot[bc, Axes -> True, AxesLabel -> {"x: alpha1", "y: alpha2"}]

Out[22]=
```




```
In[23]:= BarChart3D[bc, AxesLabel -> {"x: alpha1 ", "y: alpha2  ", "counts"}]

Out[23]=
```

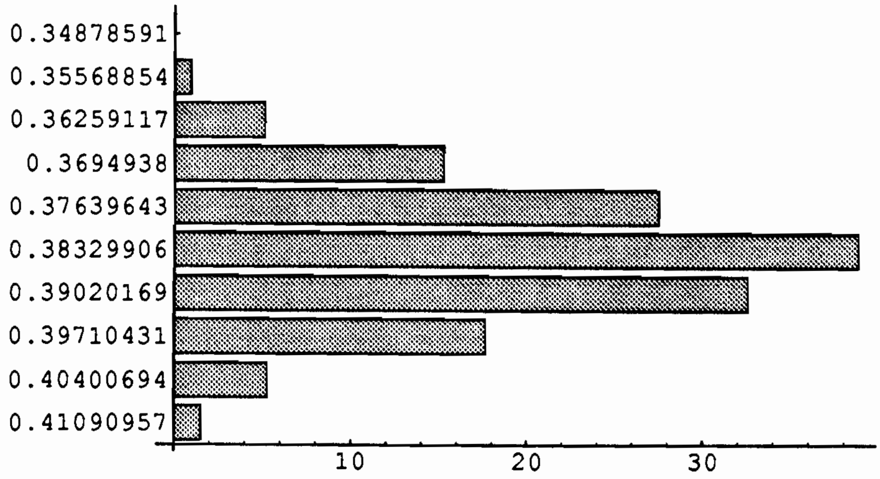


As before, we can easily get reports of the characteristics of the sampled parameters,

```
In[24]:= LocationReport[palpha1]

Out[24]= {Mean -> 0.153261, HarmonicMean -> 0.152872, Median -> 0.152999}

In[25]:= DispersionReport[palpha1]

Out[25]= {Variance -> 0.0000598379, StandardDeviation -> 0.00773549,
          SampleRange -> 0.0522568, MeanDeviation -> 0.00620629,
          MedianDeviation -> 0.00524782, QuartileDeviation -> 0.00527508}

In[26]:= LocationReport[palpha2]

Out[26]= {Mean -> 0.462792, HarmonicMean -> 0.462554, Median -> 0.462769}

In[27]:= DispersionReport[palpha2]

Out[27]= {Variance -> 0.000109767, StandardDeviation -> 0.010477,
          SampleRange -> 0.065284, MeanDeviation -> 0.00841525,
          MedianDeviation -> 0.00707425, QuartileDeviation -> 0.00713225}
```

We can also plot histograms of any of the drawn parameters, say, α_3,

```
In[28]:= HHistogram[palpha3]

Out[28]=
```



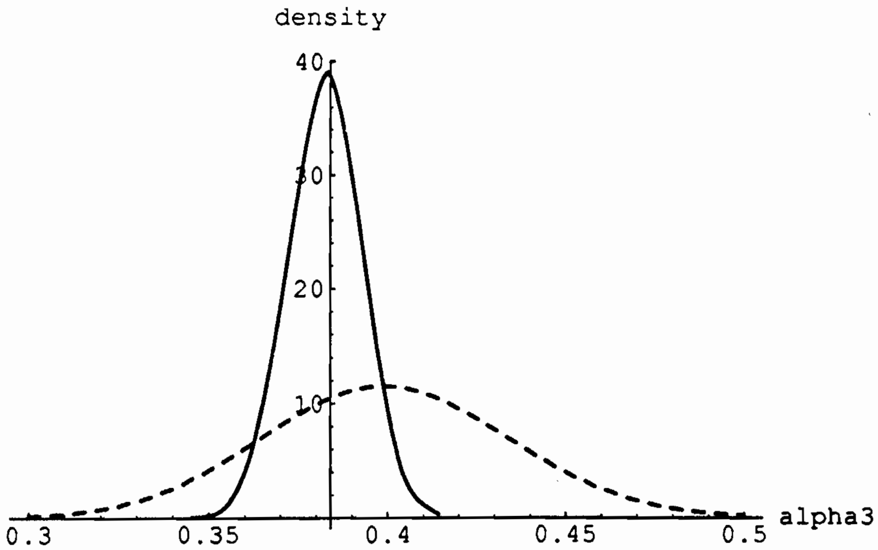
We can smooth the histogram and plot it in the same picture as the implied marginal prior (dashed) on α_3, which is a beta density on $(0,1)$:

$$p(\alpha_3) = f_\beta(\alpha_3|80, 120).$$

The corresponding density function is given in Appendix A.

```
In[29]:= graph1 = Plot[PDF[BetaDistribution[80, 120], a3], {a3, 0.30, 0.50},
         PlotRange -> All, PlotStyle -> {Dashing[{0.015, 0.015}]}, DisplayFunction -> Identity];
         graph2 = SmoothedVHistogram[palpha3, DisplayFunction -> Identity];
         Show[graph1, graph2, AxesLabel -> {"alpha3", "density"}, AxesOrigin -> {Mean[palpha3], 0},
         DisplayFunction -> $DisplayFunction ]
```

Out[29]=

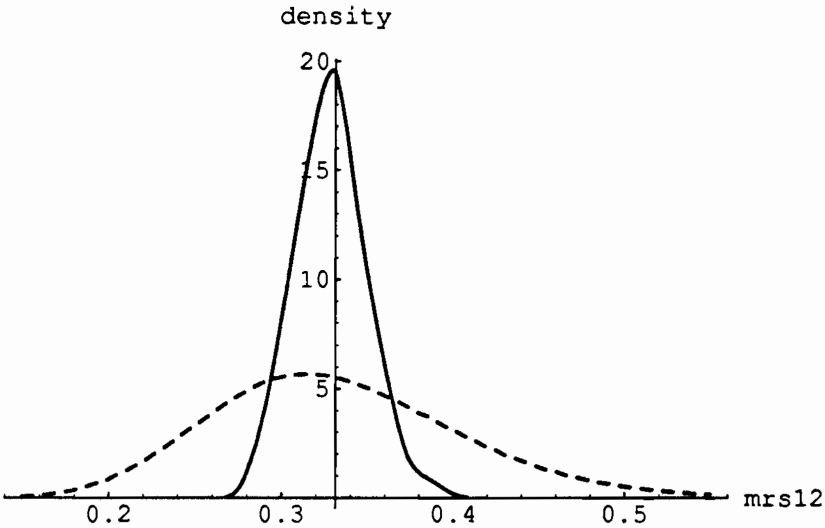


Finally, we can look at the posterior distribution of any transformation of the parameters. For instance, the marginal rate of substitution between x_1 and x_2 when equal amounts of both goods are consumed is given by α_1/α_2. Then,

```
In[30]:= mrs12 = palpha1/palpha2;
         LocationReport[mrs12]

Out[30]= {Mean -> 0.331492, HarmonicMean -> 0.33017, Median -> 0.330918}
```

By saving the partition of all drawn prior values of α_1/α_2 over prespecified bins (remember that there are close to 200,000 drawings here so it would not be efficient to store all of them), we can contrast the posterior density with its prior counterpart (in a dashed line).[8] Generally, the set of *all* drawn values corresponds to the prior, whereas the subset of *accepted* drawings characterizes the posterior densities.

```
In[31]:= mrspart = ReadList["mrspart.dat"];
        graph1 = Plot[Interpolation[mrspart][x], {x, 0.15, 0.55},
        PlotRange->All, PlotStyle -> {Dashing[{0.015, 0.015}]}, DisplayFunction -> Identity];
        graph2 = SmoothedVHistogram[mrsl2,  DisplayFunction -> Identity];
        Show[graph1, graph2, AxesLabel -> {"mrsl2", "density"}, AxesOrigin -> {Mean[mrsl2], 0},
        DisplayFunction -> $DisplayFunction ]
Out[31]=
```



4. Concluding Remarks

Mathematica provides an excellent environment for doing Bayesian econometrics because of the ease with which it can be programmed and its outstanding capabilities in graphics, symbolic analysis and data manipulation. In addition, its interactive flavour, specially in a Notebook front end, is ideally suited to the subjectivist perspective. Beliefs keep evolving while different models and representations are tried.

We have illustrated the use of *Mathematica* functions in a traditional conjugate analysis of the linear regression model. In our second example, we have shown how *Mathematica* could be used to succesfully carry out the analysis of a completely nonstandard model.

We don't provide a *comprehensive* Bayesian package because that seems too ambitious a task. Different analyses and analysts will require different functions and methods that, when needed, can easily be accommodated into *Mathematica* with little programming effort. However, the method of rejection sampling seems an extremely flexible and easily understood mechanism for the application of Bayesian analysis in a very general context. Its main cost is in terms of computer time, but we feel this drawback can seriously (at least a factor of 10 and more likely a factor of 100) be reduced by performing the actual rejection sampling in a faster (compiled) computing language (such as FORTRAN) and embedding this through *MathLink* in a *Mathematica* environment to take full advantage of the graphical, symbolic and data handling features of *Mathematica*.

[8] The pairs of counts and bin coordinates were saved into the file **mrspart** which we read back into *Mathematica* now to construct the histogram.

Appendix A

Since some of the densities used in this paper often appear in the literature with different parametrizations, we include here the forms used in the text and implemented in this package [see, *e.g.*, Zellner (1971) for more details].

The random variable $z > 0$ has a gamma distribution with parameters α and β if its density function is given by

$$f_\gamma(z|\alpha,\beta) = \frac{z^{\alpha-1}e^{-z/\beta}}{\beta^\alpha \Gamma[\alpha]},$$

with $E[z] = \alpha\beta$, and $Var[z] = \alpha\beta^2$. (A gamma distribution with shape parameter $\alpha = 1$ is also known as an exponential distribution.)

The random variable $\sigma > 0$ has an inverted gamma distribution if its density function is given by

$$f_{i\gamma}(\sigma|\nu,s) = \frac{2}{\Gamma(\nu/2)}\left(\frac{\nu s^2}{2}\right)^{\nu/2}\frac{1}{\sigma^{\nu+1}}\exp\left\{-\frac{\nu s^2}{2\sigma^2}\right\}$$

we have (for $\nu > 2$)

$$E[\sigma] = \frac{\Gamma((\nu-1)/2)}{\Gamma(\nu/2)}\left(\frac{\nu}{2}\right)^{1/2}s, \quad E[\sigma^2] = \frac{\nu}{\nu-2}s^2, \quad \text{mode}[\sigma] = \left(\frac{\nu}{\nu+1}\right)^{1/2}s.$$

The k-dimensional random vector x has a multivariate-t distribution with mean μ, precision matrix H, and degrees of freedom ν, if its density is given by

$$f_t^k(x|\mu,H,\nu) = \frac{\Gamma((k+\nu)/2)}{\pi^{k/2}\Gamma(\nu/2)}\sqrt{\frac{|H|}{\nu^k}}\left[1+\frac{1}{\nu}(x-\mu)'H(x-\mu)\right]^{-(k+\nu)/2},$$

which implies

$$E[x] = \mu, \qquad Var[x] = \frac{\nu}{\nu-2}H^{-1}$$

when $\nu > 2$.

The random variable $v \in (0,1)$ is beta distributed if its density function is given by

$$f_\beta(v|a,b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}v^{a-1}(1-v)^{b-1},$$

where $a, b > 0$. Its mean is $E[v] = a/(a+b)$ and its variance $Var[v] = ab(a+b)^{-2}(a+b+1)^{-1}$.

The vector $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n)'$ with $\alpha_i > 0$, $\forall i$, and $\sum_{i=1}^n \alpha_i = 1$ has a Dirichlet distribution with parameters $\gamma = (\gamma_1, \gamma_2, \dots, \gamma_n)'$ if its density function is given by

$$f_D^n(\alpha|\gamma) = \frac{\Gamma(g)}{\prod_{i=1}^n \Gamma(\gamma_i)}\prod_{i=1}^n \alpha_i^{\gamma_i - 1},$$

where $\gamma_i > 0$, $\forall i$, and $g = \sum_{i=1}^n \gamma_i$. The first two moments are given by

$$E[\alpha_i] = \frac{\gamma_i}{g}, \quad \text{and } Var[\alpha_i] = \frac{\gamma_i(g-\gamma_i)}{g^2(g+1)}.$$

Appendix B

Listing of BayesE.m:

```
(* file version: 9/21/92 *)

(*:Version: Mathematica 2.0 *)
(*:Name:  BayesianEmetrics *)
(*:Title:  Tools for Bayesian Econometrics *)
(*:Author:  Eduardo Ley and Mark F.J. Steel, July 1992 *)
(*:Legal:  Copyright (c) 1992, Eduardo Ley and Mark F.J. Steel *)
(*:Summary: *)
(*:Keywords: Bayesian, Regression *)
(*:Requirements: No special system requirements. *)
(*:Warning: None. *)
(*:Sources:
A. Zellner (1971): "An Introduction to Bayesian Inference in Econometrics,"
NY: Wiley.
G.G. Judge, R.C. Hill, W.E. Griffiths, H. Lutkepohl, T.-C. Lee (1982):
"Introduction to the Theory and Practice of Econometrics," NY: Wiley.
*)

BeginPackage["BayesianEmetrics`",
 "Statistics`ContinuousDistributions`",
    "Statistics`NormalDistribution`",
    "Statistics`InverseStatisticalFunctions`",
    "Statistics`Common`DistributionsCommon`",
    "Statistics`DescriptiveStatistics`",
    "Statistics`DataManipulation`",
    "Graphics`Graphics`",
    "Graphics`Graphics3D`"
    ]

IGammaDistribution::usage =
"IGammaDistribution[alpha,beta] represents the Inverted Gamma (Type-2)
Distribution as in
H. Raiffa and R. Schlaifer (1961):
Applied Statistical Decision Theory;
Boston: Harvard Business School; or
Arnold Zellner (1971):
Bayesian Inference in Econometrics,
pp. 369-373; New York: Wiley."

MultivariateTDistribution::usage =
"MultivariateTDistribution[m, H, v] represents a multivariate t
distribution with mean vector m, and covariance matrix (v/(v-2))Inverse[H].
Arnold Zellner (1971):
Bayesian Inference in Econometrics,
pp. 383-389; New York: Wiley."

UnivariateTDistribution::usage =
"UnivariateTDistribution[m, h, v] represents a univariate t
distribution with mean vector m, and variance (v/(v-2))(1/h).
Arnold Zellner (1971):
Bayesian Inference in Econometrics,
pp. 366-367; New York: Wiley."

MultivariateNormalDistribution::usage =
```

```
"MultivariateNormalDistribution[mu, sigma] represents a multivariate
Normal distribution with mean mu and covariance matrix sigma."

Likelihood::usage =
"Likelihood[density,data] calculates the
 likelihood function of the data using the specified
 density---ie, the PDF of the density is evaluated
 at each data point and the product computed."

LogLikelihood::usage =
"LogLikelihood[density,data] calculates the
 log-likelihood function of the data using the
 specified density---ie, the log[PDF] of the density
 is evaluated at each data point and the sum computed."

Marginal::usage =
"Marginal[ list, MultivariateDistribution[ parameters ]], where list is a
list of indices and MultivariateDistribution is either
MultivariateTDistribution or MultivariateNormal, gives the marginal
distribution of the variables whose indices are listed in list."

BayesRegression::usage =
"BayesRegression[y, x, priormeanbeta, A, priormeansigma, n, xf] computes the
posterior distribution for beta, sigma (and future values of y if xf is
provided).  If prior means for beta and sigma are provided along with the
precision matrix A, and n, then it assumed that the priors are natural
conjugates.  Otherwise, diffuse priors are assumed."

postdistbeta::usage =
"It is returned by BayesRegression and contains the
posterior distribution of the regression vector, beta."

postdistsigma::usage =
"It is returned by BayesRegression and contains the
posterior distribution of the error standard deviation, sigma."

preddisty::usage=
"It is returned by BayesRegression when xf is specified
and contains the posterior distribution of the predicted values of y."

HHistogram::usage=
"Histogram[xdata, <numberofbins>] returns a HORIZONTAL Histogram of xdata.
*** WARNING *** there's a bug in BinCounts and sometimes the number
of bins is not what Histogram expects, try multiplying your data by
some constant like 100 or 1000."

VHistogram::usage=
"Histogram[xdata, <numberofbins>] returns a VERTICAL Histogram of xdata.
Only two decimals will be shown in the x axis.  Multiply the data by some
constant or use VHistogram if that's not enough precision.
*** WARNING *** there's a bug in BinCounts and sometimes the number
of bins is not what Histogram expects, try multiplying your data by
some constant like 100 or 1000."

SmoothedVHistogram::usage=
"Histogram[xdata, <numberofbins>] returns a SMOOTHED VERTICAL Histogram of xdata.
*** WARNING *** there's a bug in BinCounts and sometimes the number
of bins is not what Histogram expects, try multiplying your data by
```

```
some constant like 100 or 1000."

Histogram3D::usage=
"Histogram[xydata, <numberofbins>, opts] uses BarChart3D and ListOfBinCounts
to construct a 3D Histogram.
*** WARNING *** there's a bug in BinCounts and sometimes the number
of bins is not what Histogram3D expects, try multiplying your data by
some constant like 100 or 1000."

ListOfBinCoordinates::usage=
"It returns the x and y coordinates at the bins' center.  It is used in
conjunction with Histogram3D since it displays only the bin number and not
its coordinates."

ListOfBinCounts::usage=
"ListOfBinCounts[xy_List, n_Integer] is an auxiliary function that calls
BinCounts and returns a list of n bin counts of xy."

(*******************************************************************
Begin Private Context
*******************************************************************)

Begin["'Private'"]

(*******************************************************************
Inverted Gamma Distribution
Commonly used in Bayesian Statistics,
see H. Raiffa and R. Schlaifer (1961):
"Applied Statistical Decision Theory;"
Boston: Harvard Business School; or
Arnold Zellner (1971), pp. 369-373.
********************************************************************)

IGammaDistribution/:
Domain[IGammaDistribution[___]] := {0,Infinity}

IGammaDistribution/:
PDF[IGammaDistribution[v_, s_], x_] :=
With[{result = (2*(1/2)^(v/2)*x^(-1-v)*(s^2*v)^(v/2))/
   (E^((s^2*v)/(2*x^2))*Gamma[v/2])},
If[NumberQ[N[result]],N[result],result]]

IGammaDistribution/:
CDF[IGammaDistribution[v_, s_], x_] :=
With[{result = (Gamma[v/2] -
Gamma[v/2, 0, (v*s^2)/(2*x^2)])/Gamma[v/2]},
If[NumberQ[N[result]],N[result],result]]

IGammaDistribution/:
Mean[IGammaDistribution[v_, s_]] :=
With[{result = Sqrt[v/2]Gamma[(v-1)/2]s/Gamma[v/2]},
If[NumberQ[N[result]],N[result],result]]

IGammaDistribution/:
Variance[IGammaDistribution[v_, s_]] :=
With[{result = ((v*s^2)/(v-2))-
(Mean[IGammaDistribution[v,s]])^2},
If[NumberQ[N[result]],N[result],result]]
```

```
IGammaDistribution/:
StandardDeviation[IGammaDistribution[v_, s_]] :=
With[{result = Sqrt[Variance[IGammaDistribution[v,s]]]},
If[NumberQ[N[result]],N[result],result]]

IGammaDistribution/:
Skewness[IGammaDistribution[n_,s_]] :=
With[{result = (n^(3/2)*s^3*Gamma[(-1 + n)/2]^3)/
(2^(1/2)*Gamma[n/2]^3) + (n^(3/2)*s^3*Gamma[(-3 + n)/2])/
    (2^(3/2)*Gamma[n/2]) - (3*2^(3/2)*n^(3/2)*s*Gamma[(-1 + n)/2])/
   ((-2 + n)*Gamma[n/2])},
If[NumberQ[N[result]],N[result],result]]

IGammaDistribution/:
Kurtosis[IGammaDistribution[v_, s_]] :=
With[{result =(s^4*v^2)/(8 - 6*v + v^2) -
(3*n^2*s^4*Gamma[(-1 + n)/2]^4)/(4*Gamma[n/2]^4) -
    (n^2*s^4*Gamma[(-3 + n)/2]*Gamma[(-1 + n)/2])/
    Gamma[n/2]^2 + (12*n^2*s^2*Gamma[(-1 + n)/2]^2)/((-2 + n)*Gamma[n/2]^2)},
    If[NumberQ[N[result]],N[result],result]]

IGammaDistribution/:
KurtosisExcess[IGammaDistribution[v_, s_]] :=
NotImplemented/; False;

IGammaDistribution/:
CharacteristicFunction[IGammaDistribution[v_, s_]] :=
NotImplemented/; False;

IGammaDistribution/:
Quantile[IGammaDistribution[v_, s_], q_] :=
x/.FindRoot[q-CDF[IGammaDistribution[v,s],x],
{x,1,0,v*s*1000}]

IGammaDistribution/:
Random[IGammaDistribution[v_, s_]] :=
With[{result = Quantile[IGammaDistribution[v,s],Random[]]},
If[NumberQ[N[result]],N[result],result]]

(**********************************************************************
Univariate Student t distribution. See
Arnold Zellner (1971), pp. 366-367.
**********************************************************************)

UnivariateTDistribution/:
Domain[UnivariateTDistribution[___]] := {-Infinity, Infinity}

UnivariateTDistribution/:
PDF[UnivariateTDistribution[m_, h_, v_], x_] :=
With[{result = (Gamma[(v+1)/2]*Sqrt[h/v])/
(Sqrt[Pi]*Gamma[v/2]*(1+(h/v)*(x-m)^2)^((v+1)/2))},
If[NumberQ[N[result]],N[result],result]]

UnivariateTDistribution/:
CDF[UnivariateTDistribution[m_, h_, v_], x_] :=
With[{result = CDF[StudentTDistribution[v], Sqrt[h]*(x-m)]},
If[NumberQ[N[result]],N[result],result]]
```

```
UnivariateTDistribution/:
Mean[UnivariateTDistribution[m_, h_, v_]] :=
With[{result = m},
If[NumberQ[N[result]],N[result],result]]

UnivariateTDistribution/:
Variance[UnivariateTDistribution[m_, h_, v_]] :=
With[{result = (v/(v-2))*(1/h)},
If[NumberQ[N[result]],N[result],result]]

UnivariateTDistribution/:
StandardDeviation[UnivariateTDistribution[m_, h_, v_]] :=
With[{result = Sqrt[(v/(v-2))*(1/h)]},
If[NumberQ[N[result]],N[result],result]]

UnivariateTDistribution/:
Skewness[UnivariateTDistribution[ n_, s_]] := 0

UnivariateTDistribution/:
Kurtosis[UnivariateTDistribution[m_, h_, v_]] :=
With[{result =(3/((v-2)(v-4)))*(v/h)^2},
    If[NumberQ[N[result]],N[result],result]]

UnivariateTDistribution/:
KurtosisExcess[UnivariateTDistribution[m_, h_, v_]] :=
With[{result =6/(v-4)},
    If[NumberQ[N[result]],N[result],result]]

UnivariateTDistribution/:
CharacteristicFunction[UnivariateTDistribution[m_, h_, v_]] :=
NotImplemented/; False;

UnivariateTDistribution/:
Quantile[UnivariateTDistribution[m_, h_, v_], q_] :=
With[{result = (Quantile[StudentTDistribution[v], q]/Sqrt[h])+ m},
If[NumberQ[N[result]],N[result],result]]

UnivariateTDistribution/:
Random[UnivariateTDistribution[m_, h_, v_]] :=
Quantile[UnivariateTDistribution[m, h, v], Random[]]

(****************************************************************
Multivariate t Density
  As defined in A. Zellner (1971), pp. 383-389.
 ****************************************************************)

MultivariateTDistribution/:
Domain[MultivariateTDistribution[___]] := {-Infinity,Infinity}

MultivariateTDistribution/:
PDF[MultivariateTDistribution[mean_, H_, v_], x_] :=
Module[{k, qform},
k = Max[Length[mean],TensorRank[H]];
qform = Transpose[x-mean].H.(x-mean);
With[{result = (Gamma[(v+k)/2]*Sqrt[Det[H]/v^k])/
(Sqrt[Pi^k]*Gamma[v/2]*Sqrt[(1+qform/v)^(v+k)])},
If[NumberQ[N[result]],N[result],result]]
```

```
]

MultivariateTDistribution/:
Mean[MultivariateTDistribution[mean_, H_, v_]] :=
With[{result = mean},
If[NumberQ[N[result]],N[result],result]]

MultivariateTDistribution/:
Variance[MultivariateTDistribution[mean_, H_, v_]] :=
With[{result = (v/(v-2))*Inverse[H]},
If[NumberQ[N[result]],N[result],result]]

MultivariateTDistribution/:
Marginal[ijk_List, MultivariateTDistribution[mean_, H_, v_]]:=
Module[{inH, Hijk,lenz},
inH = Inverse[H];
Hijk = Inverse[inH[[ijk,ijk]]];
Length[ijk];
If[Length[ijk]==1,
UnivariateTDistribution[Flatten[mean[[ijk]]][[1]],
Flatten[1/inH[[ijk,ijk]]][[1]], v],
MultivariateTDistribution[mean[[ijk]], Hijk, v]
]
]

(**********************************************************************
Multivariate Normal
**********************************************************************)

MultivariateNormalDistribution/:
Domain[MultivariateNormalDistribution[___]] := {-Infinity,Infinity}

MultivariateNormalDistribution/:
PDF[MultivariateNormalDistribution[mu_,A_], x_] :=
Module[{k},
k = Max[Length[mean],TensorRank[A]];
With[{result = 1/Sqrt[(2Pi)^k Det[A]]*
Exp[0.5*Transpose[x-mu].Inverse[A].(x-mu)]},
If[NumberQ[N[result]],N[result],result]]
]

MultivariateNormalDistribution/:
Mean[MultivariateNormalDistribution[mu_, A_]] :=
With[{result = mu},
If[NumberQ[N[result]],N[result],result]]

MultivariateNormalDistribution/:
Variance[MultivariateNormalDistribution[mu_, A_]] :=
With[{result = A},
If[NumberQ[N[result]],N[result],result]]

MultivariateNormalDistribution/:
Marginal[ijk_List, MultivariateNormalDistribution[mu_, A_]]:=
If[Length[ijk]==1,
NormalDistribution[Flatten[mu[[ijk]]][[1]], Flatten[A[[ijk,ijk]]][[1]]],
MultivariateNormalDistribution[mu[[ijk]], A[[ijk,ijk]]]
]
```

```
(*******************************************************************
Likelihood Function
*******************************************************************)

Likelihood[density_, data_] :=
  Module[{i}, Product[PDF[density,data[[i]]],{i,1,Length[data]}]]

LogLikelihood[density_,data_] := Log[LikelihoodFunction[density, data]]

(*******************************************************************
Bayesian Regression
*******************************************************************)

BayesRegression[y_, x_] :=
Module[{betahat, nu, ehat, sigma2hat, ivar},
betahat  = N[Inverse[Transpose[x].x].Transpose[x].y];
nu = Length[y]-Length[betahat];
ehat = y - x.betahat;
sigma2hat  = N[(Transpose[ehat].ehat)/nu];
ivar  = N[Transpose[x].x/sigma2hat];
postdistbeta = MultivariateTDistribution[betahat, ivar, nu];
postdistsigma = IGammaDistribution[nu,Sqrt[sigma2hat]];
]

BayesRegression[y_, x_, xf_] :=
Module[{mean, ivar, ivaryf, betahat, sigma2hat, nu},
betahat  = N[Inverse[Transpose[x].x].Transpose[x].y];
mean  = N[xf.betahat];
nu = Length[y]-Length[betahat];
ehat = y - x.betahat;
sigma2hat  = N[(Transpose[ehat].ehat)/nu];
ivar  = N[Transpose[x].x/sigma2hat];
ivaryf  = N[(1/sigma2hat)*
Inverse[IdentityMatrix[Part[Dimensions[xf],1]] +
xf.Inverse[Transpose[x].x].Transpose[xf]]];
postdistbeta = MultivariateTDistribution[betahat, ivar, nu];
postdistsigma = IGammaDistribution[nu,Sqrt[sigma2hat]];
preddisty = MultivariateTDistribution[mean, ivaryf, nu];
]

BayesRegression[y_, x_, priorbeta_, A_, priorsigma_, priorn_]:=
Module[{postn, postbeta, postsigma, ivar},
postn = priorn + Length[y];
postbeta  = N[Inverse[A+Transpose[x].x].(A.priorbeta+Transpose[x].y)];
    postsigma = N[Sqrt[(priorn*priorsigma^2 + Transpose[y].y -
            Transpose[postbeta].(A+Transpose[x].x).postbeta +
            Transpose[priorbeta].A.priorbeta)/postn]];
    ivar  = N[(A+Transpose[x].x)*postsigma^(-2)];
postdistbeta = MultivariateTDistribution[postbeta, ivar, postn];
postdistsigma = IGammaDistribution[postn, postsigma];
]

BayesRegression[y_, x_, MultivariateTDistribution[m_, H_, n_],
IGammaDistribution[n_, s_]] := BayesRegression[y, x, m, H*s^2, s, n];

BayesRegression[y_, x_, priorbeta_, A_, priorsigma_,  priorn_, xf_]:=
Module[{mean, ivar, ivaryf, postn, postbeta, postsigma},
    postn = priorn + Length[y];
```

```
postbeta  = N[Inverse[A+Transpose[x].x].(A.priorbeta+Transpose[x].y)];
    postsigma = N[Sqrt[(priorn*priorsigma^2 + Transpose[y].y -
            Transpose[postbeta].(A+Transpose[x].x).postbeta +
            Transpose[priorbeta].A.priorbeta)/postn]];
mean = N[xf.postbeta];
ivar  = N[(A+Transpose[x].x)*postsigma^(-2)];
ivaryf = N[postsigma^(-2)*
Inverse[IdentityMatrix[Part[Dimensions[xf],1]] +
xf.Inverse[A+Transpose[x].x].Transpose[xf]]];
postdistbeta  = MultivariateTDistribution[postbeta, ivar, postn];
postdistsigma = IGammaDistribution[postn, postsigma];
preddisty     = MultivariateTDistribution[mean, ivaryf, postn];
]

BayesRegression[y_, x_, MultivariateTDistribution[m_, H_, n_],
IGammaDistribution[n_, s_], xf] := BayesRegression[y, x, m, H*s^2, s, n, xf]

(**********************************************************************
       Histograms
**********************************************************************)

HHistogram[x_List, n_Integer:10, opts___Rule] :=
Module[{dx, maxx, minx, area},
maxx = Max[x];
minx = Min[x];
dx = (maxx - minx)/n;
area = dx * Length[x];
BarChart[Transpose[{BinCounts[x, {minx, maxx, dx}]/area,
Table[N[i, 8], {i, maxx - dx/2, minx + dx/2, -dx}]}],
BarOrientation->Horizontal, opts]]

HHistogram[x_List, opts___Rule] := HHistogram[x, 10, opts]

VHistogram[x_List, n_Integer:10, opts___Rule] :=
Module[{dx, maxx, minx, area},
maxx = Max[x];
minx = Min[x];
dx = (maxx - minx)/n;
area = dx * Length[x];
BarChart[Transpose[{BinCounts[x, {minx, maxx, dx}]/area,
Table[N[i, 2], {i, minx + dx/2, maxx - dx/2, dx}]}],
opts]]

VHistogram[x_List, opts___Rule] := VHistogram[x, 10, opts]

SmoothedVHistogram[x_List, n_Integer:10, opts___Rule] :=
Module[{dx, maxx, minx, bc, xc, z, hist, area},
maxx = Max[x];
minx = Min[x];
dx = (maxx-minx)/n;
area = dx * Length[x];
(*we create two extra bins at the extremes*)
(*we rescale so that it integrates to 1*)
bc = BinCounts[x, {minx - 2*0.9*dx, maxx + 2*0.9*dx, dx}]/area;
xc = Range[minx - (3/2)*dx, maxx + (3/2)*dx, dx];
z = Transpose[ColumnJoin[{xc}, {bc}]];
hist = Interpolation[z];
Plot[hist[y], {y, minx, maxx}, opts]
```

```
]

SmoothedVHistogram[x_List, opts___Rule] := SmoothedVHistogram[x, 10, opts]

ListOfBinCounts[xy_List, n_Integer:10] :=
Module[{x, y, dx, dy, maxx, minx, maxy, miny},
x = Transpose[xy][[1]];
y = Transpose[xy][[2]];
maxx = Max[x];
minx = Min[x];
maxy = Max[y];
miny = Min[y];
dx = (maxx-minx)/n;
dy = (maxy-miny)/n;
BinCounts[xy, {minx, maxx, dx}, {miny, maxy, dy}]
]

Histogram3D[xy_List, n_Integer:10, opts___Rule] :=
BarChart3D[ListOfBinCounts[xy, n], opts]

Histogram3D[xy_List, opts___] := Histogram3D[xy, 10, opts]

ListOfBinCoordinates[xy_List, n_Integer:10] :=
Module[{x, y, dx, xx, yy, maxx, minx, maxy, miny},
x = Transpose[xy][[1]];
y = Transpose[xy][[2]];
maxx = Max[x];
minx = Min[x];
maxy = Max[y];
miny = Min[y];
dx = (maxx-minx)/n;
dy = (maxy-miny)/n;
xx = Range[minx, maxx, dx];
yy = Range[miny, maxy, dy];
TableForm[Table[{N[xx[[i]], 4], N[yy[[i]], 4]}, {i, 1, n}],
TableHeadings->{Automatic, {"x coord.", "y coord."}}]
]

(*********************************************************************
End of Private Context
*********************************************************************)

End[]

(*********************************************************************
HouseKeeping
*********************************************************************)

SetAttributes[IGammaDistribution, ReadProtected];
SetAttributes[UnivariateTDistribution, ReadProtected];
SetAttributes[MultivariateTDistribution, ReadProtected];
SetAttributes[MultivariateNormalDistribution, ReadProtected];
SetAttributes[BayesRegression, ReadProtected];
SetAttributes[Marginal, ReadProtected];
SetAttributes[Likelihood, ReadProtected];
SetAttributes[LogLikelihood, ReadProtected];
SetAttributes[VHistogram, ReadProtected];
SetAttributes[HHistogram, ReadProtected];
```

```
SetAttributes[SmoothedVHistogram, ReadProtected];
SetAttributes[Histogram3D, ReadProtected];
SetAttributes[ListOfBinCoordinates, ReadProtected];
SetAttributes[ListOfBinCounts, ReadProtected];

Protect[IGammaDistribution, UnivariateTDistribution,
MultivariateTDistribution, MultivariateNormalDistribution,
BayesRegression, Marginal,
Likelihood, LogLikelihood,
VHistogram, SmoothedVHistogram,
HHistogram, Histogram3D,
ListOfBinCoordinates, ListOfBinCounts
];

(*********************************************************************
End of Package
*********************************************************************)

EndPackage[]
```

Listing of **health86.dat**:

18613	13312	16018	260400	1.30
118344	79880	7566	1423050	16.83
366750	282200	9862	5121000	45.07
44276	32944	25374	502200	1.22
39778	34109	5121	667141	10.01
26328	20710	4918	360300	6.02
429967	326990	55393	5035000	7.41
156690	122040	61066	1931220	2.49
296180	222809	9966	5543000	89.68
12218	10729	243	158167	46.05
1450	1278	3541	18540	0.74
61310000	47400000	57246	896320000	1370.28
22428000	16274000	121490	333236800	220.01
14945	13367	370	220500	43.10
35730	26265	14572	429900	2.50
3670	3169	3279	53079	1.54
36308	34988	4169	514600	8.28
291753	172000	10230	4403000	76.07
1933499	1382000	38668	31955000	103.23
84972	77213	8370	931784	8.51
18613	12703	6573	243400	2.44
1400000	525000	51731	39290000	196.10
22677	19640	56763	376500	0.57
455700	188900	241625	4191000	1.00

Listing of **health87.dat**:

21052	14835	16263	291887	1.35
124271	83340	7576	1481560	16.8
386000	297000	9868	5337800	44.5
47807	35335	25652	544900	1.23
41400	35400	5130	693028	10.2
29110	22894	4932	393600	6.21
450348	336750	55627	5289000	7.43

161800	126930	61077	2009090	2.47
336000	253000	9998	6390000	100
16273	14413	246	207640	54.2
1470	1279	3542	19780	0.74
70890000	56110000	57345	979680000	1399
23853000	17402000	122091	348911300	213
16680	15284	372	223500	41
36550	26995	14671	431800	2.4
4100	3382	3309	59257	1.69
41527	40549	4184	556900	8.64
333000	202000	10280	5169000	84.1
2145600	1533600	38830	35710000	106
92211	83752	8399	1005226	8.69
19700	18256	6619	255180	2.43
2012855	831449	52893	57770000	262
24798	21433	56930	409900	0.58
500300	207300	243934	4473000	1.00

Listing of **cons.dat**:

27.40	26.88	16.85	20.43	90.88	50.60
28.77	29.18	17.77	22.85	96.60	55.48
29.95	27.85	18.45	25.05	94.85	58.42
31.23	27.73	18.90	30.73	98.22	63.15
31.85	29.73	19.40	29.85	109.15	69.03
31.57	29.85	20.30	29.23	114.72	75.13
31.77	29.38	21.40	32.67	117.83	82.13
32.10	30.02	22.18	32.10	119.67	88.05
33.38	30.60	23.02	38.88	124.70	94.30
34.73	30.95	24.02	38.20	130.78	101.63
36.50	31.90	25.02	39.65	137.10	108.55
36.92	32.85	26.10	37.17	141.75	115.67
38.50	33.17	26.93	42.80	148.47	125.00
38.77	33.63	27.80	43.42	153.20	134.00
38.88	34.08	28.40	41.90	157.40	141.80
39.75	34.58	29.13	47.02	163.82	151.05
40.15	34.90	29.85	51.80	169.35	160.63
40.35	35.25	30.60	56.85	179.68	172.78
40.92	36.10	31.43	63.48	191.85	185.40
41.55	37.35	32.60	68.53	208.45	200.30
42.33	38.23	33.80	70.63	216.90	216.00
44.55	39.98	35.65	81.00	235.00	236.43
46.10	41.98	37.70	86.22	252.18	259.43
47.80	44.05	40.38	85.67	270.32	284.02
50.25	45.92	43.08	97.58	283.27	310.65
51.05	47.90	45.58	111.22	305.10	341.27
50.83	53.75	48.10	124.72	339.55	372.98
54.08	59.58	51.77	123.75	380.90	411.90
61.15	65.13	56.38	135.35	416.20	461.23
65.88	67.60	60.65	161.45	451.95	515.92
69.15	71.03	65.45	184.50	490.45	582.25
72.78	76.00	70.30	205.57	541.80	656.10
78.25	83.55	75.88	218.95	613.25	734.55
85.55	88.85	83.72	219.28	681.35	831.95
93.65	96.78	92.30	239.88	740.58	934.70
100.00	100.00	100.03	252.65	771.00	1026.97
101.60	102.50	106.13	289.10	816.70	1128.75

102.97	106.17	111.60	335.55	867.30	1227.63
102.35	108.50	117.25	368.70	913.13	1347.52
101.38	110.13	122.25	402.43	939.35	1458.05
100.40	114.05	127.42	413.73	982.88	1571.22

Contents of **mrs.part**:

```
{0.1074438108891304, 0.0006830961368485605}
{0.1440418805478707, 0.03142242229503378}
{0.1806399502066110, 0.3463297413822202}
{0.2172380198653513, 1.517156519940653}
{0.2538360895240916, 3.525459162275421}
{0.2904341591828319, 5.32405129059768}
{0.3270322288415722, 5.598655937610802}
{0.3636302985003125, 4.56991315551687}
{0.4002283681590528, 3.078714288776462}
{0.4368264378177931, 1.76990209057462}
{0.4734245074765334, 0.908517862008585}
{0.5100225771352737, 0.4037098168774993}
{0.5466206467940139, 0.1625768805699574}
{0.5832187164527542, 0.05396459481103629}
{0.6198167861114944, 0.02049288410545681}
{0.6564148557702346, 0.005464769094788485}
{0.6930129254289749, 0.003415480684242802}
{0.7296109950877151, 0.002049288410545681}
{0.7662090647464554, 0}
{0.8028071344051960, 0.0006830961368485605}
{0.8394052040639360, 0.0006830961368485605}
{0.8760032737226760, 0}
```

References

Drèze J.H. (1962): "The Bayesian Approach to Simultaneous Equation Estimation," O.N.R. Research Memorandum no. 67, Northwestern University.

Engle, R.F., D.F. Hendry and J.F. Richard (1983): "Exogeneity," *Econometrica*, 51, 277-304.

Florens, J,-P. and M. Mouchart (1985): "Conditioning in Dynamic Models," *Journal of Time Series Analysis*, 6, 15-34.

Good, I.J. (1976): "The Bayesian Influence, or How to Sweep Subjectivism Under the Carpet," in *Foundations of Probability Theory, Statistical Inference, and Statistical Theories of Science*, vol. II, W. Harper and C. Hooker (eds.), Dordrecht: Reidel, 125-174.

Jeffreys, H. (1961): *Theory of Probability*, Oxford: Oxford University Press.

Judge, G.C., W.E. Griffiths, R. Carter-Hill, H. Lütkepohl, T.-C. Lee (1985): *The Theory and Practice of Econometrics*, New York: Wiley.

Johnson, M.E. (1987): *Multivariate Statistical Simulation*, New York: Wiley.

Leamer, E.E. (1978): *Specification Searches*, New York: Wiley.

Ley, E. (1992): "Applied Demand Analysis in Utility Space," mimeo.

Lindley, D.V. (1986): "Comment," *American Statistician*, 40, 6-7.

Newhouse, J.P. (1977): "Medical Care Expenditure: A Cross-National Survey, "*Journal of Human Resources*, 12, 115-125.

Osiewalski J. and M.F.J. Steel (1990): "A Bayesian Analysis of Exogeneity in Models Pooling Time-Series and Cross-Section Data," mimeo.

Poirier, D.J. (1988): "Frequentist and Subjectivist Perspectives on the Problems of Model Building in Economics,"*Economic Perspectives*, 2, 121-144.

Raiffa, H.A. and R.S. Schlaifer (1961): *Applied Statistical Decision Theory*, Boston: Harvard University Press.

Ripley, B. (1986): *Stochastic Simulation*, New York: Wiley.

Smith, A.F.M. (1986): "Comment," *American Statistician*, 40, 10-11.

Smith, A.F.M. and A.E. Gelfand (1992): "Bayesian Statistics Without Tears: A Sampling-Resampling Perspective," *American Statistician*, 46, 84-88.

van den Broeck, J., G. Koop, J. Osiewalski and M.F.J. Steel (1992): "Stochastic Frontier Models: A Bayesian Perspective," División de Economía WP 92-12, Universidad Carlos III de Madrid.

Varian, H.R. (1990): "Goodness-of-Fit in Optimizing Models," *Journal of Econometrics*, 46, 125-140.

Zellner, A. (1971): *An Introduction to Bayesian Inference in Econometrics*, New York: Wiley.