Automating Bivariate Transformations

Jeff X. Yang John H. Drew Lawrence M. Leemis

Department of Mathematics

The College of William & Mary

Williamsburg, VA 23187–8795, USA

jxyang@wm.edu, jhdrew@math.wm.edu, leemis@math.wm.edu

July 29, 2010

Appendix

This appendix contains illustrations of the data structure and procedures that can be applied to bivariate distributions: `MarginalPDF`, `VerifyJointPDF`, `BiExpectedValue`, `Covariance`, and `Correlation`.

A. Data Structure Illustrations

The list-of-sublists data structure creates one major implementation issue for the transformation procedure. `BiTransform` stores the constraints in two arrays, one containing the intersection points between adjacent constraints and the other containing corresponding constraints that form the intersections. The algorithm will return an error if adjacent constraints intersect at more than one point in a particular partition. In some cases, the user can avoid this problem by manually partitioning the region so that adjacent constraints only intersect once. This is shown in Example 4 below.

Examples:

1. Consider the joint PDF of two independent U(0, 1) random variables

$$f_{X,Y}(x, y) = 1 \qquad 0 < x < 1, 0 < y < 1.$$

The following Maple statement creates a list-of-sublists that represents the PDF of X and Y.

```
> XY := [[(x, y) -> 1], [[x > 0, y > 0, x < 1, y < 1]],
         ["Continuous", "PDF"]];
```

2. Consider the joint PDF of a bivariate normal distribution

$$f_{X,Y}(x,y) = \frac{1}{2\pi\sigma_X\sigma_Y\sqrt{(1-\rho^2)}}\exp\left(-\frac{1}{2(1-\rho^2)}\left(\frac{(x-\mu_X)^2}{\sigma_X^2}+\frac{(y-\mu_Y)^2}{\sigma_Y^2}-\frac{2\rho(x-\mu_X)(y-\mu_Y)}{\sigma_X\sigma_Y}\right)\right)$$

with support $-\infty < x < \infty, -\infty < y < \infty$ and real parameters $\mu_X, \mu_Y, \sigma_X > 0, \sigma_Y > 0, -1 < \rho < 1$. The following statements define the bivariate normal distribution as a list-of-sublists. Unspecified parameters are entered symbolically.

```
> assume(sx > 0);
> assume(sx > 0);
> assume(cor > -1, cor < 1);
> XY := [[(x, y) -> exp(-((x - mx) ^ 2 / sx ^ 2 + (y - my) ^ 2 /
          sy ^ 2 - 2 * cor * (x - mx) * (y - my) / (sx * sy)) /
          (2 * (1 - cor ^ 2))) / (2 * Pi * sx * sy *
          sqrt(1 - cor ^ 2))],
         [[x > -infinity, y > -infinity, x < infinity,
           y < infinity]], ["Continuous", "PDF"]];
```

3. The joint PDF defined piecewise

$$f_{X,Y}(x,y) = \begin{cases} \frac{1}{4} & 0 < x < \frac{1}{2},\ 0 < y < 1 \\ \frac{7}{4} & \frac{1}{2} < x < 1,\ 0 < y < 1 \end{cases}$$

can be represented as a list-of-sublists with the Maple statement

```
> XY := [[(x, y) -> 1 / 4, (x, y) -> 7 / 4],
         [[x > 0, y > 0, x < 1 / 2, y < 1],
         [x > 1 / 2, y > 0, x < 1, y < 1]],
         ["Continuous", "PDF"]];
```

4. Consider the random variables X and Y with joint PDF

$$f_{X,Y}(x,y) = 3/4 \qquad y > 0, y < 1 - x^2.$$

Defining the data structure as

```
> XY := [[(x, y) -> 3 / 4], [[y > 0, y < 1 - x ^ 2]],
         ["Continuous", "PDF"]];
```

will create problems because the two constraints intersect at $(-1, 0)$ and $(1, 0)$. If we partition the region along the line $x = 0$, we can then represent the data structure in a form that is compatible with `BiTransform`. The partitioned data structure is correctly input by the statement

```
> XY := [[(x, y) -> 3 / 4, (x, y) -> 3 / 4], [[y > 0, x < 0,
          -x < sqrt(1 - y)], [y > 0, x > 0, x > sqrt(1 - y)]],
         ["Continuous", "PDF"]];
```

B. `JointPDF`

Syntax: The statement

```
JointPDF(X, Y, xp, yp);
```

returns the joint PDF of independent continuous univariate random variables X and Y in the list-of-sublists format. The third and fourth parameters, `xp` and `yp`, are optional arguments for the Maple variable names for X and Y in the joint PDF of X and Y. The default variables are "x" and "y."

Purpose: `JointPDF` finds the continuous joint PDF of two independent continuous univariate random variables, X and Y, so that bivariate procedures can be applied to the joint PDF.

Algorithm: The univariate random variables, X and Y, are given in the standard APPL list-of-sublists format. Let X and Y have distributions that are defined piecewise with m and n pieces respectively. The PDF of X, $f_i(x)$, over (x_i, x_{i+1}) for all $i = 1, 2, \ldots, m$ and the PDF of Y, $f_j(y)$, over (y_j, y_{j+1}) for all $j = 1, 2, \ldots, n$ are combined and appended to two arrays, `PDF` and `supp`. The joint PDF $f_i(x)f_j(y)$ is appended to `PDF` and the rectangular constraints $x_i < x < x_{i+1}$ and $y_j < y < y_{j+1}$ are appended to `supp`. A list-of-sublists is returned to represent the joint PDF of X and Y.

Examples:

1. The joint PDF of two independent U(0, 1) random variables is found by the Maple statements

   ```
   > X := [[x -> 1], [0, 1], ["Continuous", "PDF"]];
   > Y := [[y -> 1], [0, 1], ["Continuous", "PDF"]];
   > XY := JointPDF(X, Y);
   ```

   which correctly return

   $$[[(x, y) \rightarrow 1], [[x > 0, y > 0, x < 1, y < 1]], [''Continuous'', ''PDF'']]$$

   as the joint PDF of X and Y.

2. The joint PDF of the independent random variables $X \sim$ exponential(a) and $Y \sim$ exponential(b) is found by the Maple statements

   ```
   > X := [[x -> a * e ^ (- a * x)], [0, infinity], ["Continuous", "PDF"]];
   > Y := [[y -> b * e ^ (- b * x)], [0, infinity], ["Continuous", "PDF"]];
   > JointPDF(X, Y);
   ```

   which correctly return

   $$[[(x, y) \rightarrow abe^{-ax-bx}], [[x > 0, y > 0, x < \infty, y < \infty]], [''Continuous'', ''PDF'']]$$

   as the joint PDF of X and Y.

C. `MarginalPDF`

Although the bivariate transformation technique calculates the marginal PDF of U from the joint distribution of U and V, there are times when one wishes to find the marginal PDF of a distribution without using the transformation technique. For this reason, `MarginalPDF` was created as a stand-alone procedure.

To systematically integrate over the support of X and Y, the support of X and Y is partitioned into suitable components. Then the contribution of each component to the PDF is computed. Each component is partitioned with respect to x every time the constraints defining the component intersect. For a particular component, let n be the number of such intersections and $x_1, x_2, \ldots, x_n$ be the x-values of those intersections, where $x_1 \le x_2 \le \cdots \le x_n$. As before, in order to automate this computation, we must assume that each vertical line segment spanning the component is entirely contained therein, and that the endpoints of each such line segment are determined by two different constraints. Because of this assumption, x_1 and x_n are the infimum and supremum, respectively, of the x-values that can occur over the component. For each x between x_i and x_{i+1}, denote the supremum and infimum of the y-values on the vertical line segment spanning the component by $\overline{y}_i(x)$ and $\underline{y}_i(x)$ respectively. Then, denoting the joint PDF between x_i and x_{i+1} by $f_i(x, y)$, the contribution of this particular component to the marginal PDF of X from x_i to x_{i+1} is given by

$$f_i(x) = \int_{\underline{y}_i(x)}^{\overline{y}_i(x)} f_i(x, y)\, dy \qquad \text{for } i = 1, 2, \ldots, n-1 \text{ and } x_i \ne x_{i+1}.$$

The algorithm begins at the minimum x-value, x_1, and calculates $f_1(x)$ using the integral above. The expressions $f_1(x)$, x_1, and x_2 are then stored in three arrays. Next, `MarginalPDF` finds $f_2(x)$ and appends the arrays with $f_2(x)$, x_2, and x_3. This process continues until $i = n - 1$. After the contribution of each component to the PDF has been calculated, the algorithm loops through the values in `supp`, the sorted array of all x_i's for all of the components. If the lower and upper x values of a particular interval from `supp[i]` to `supp[i + 1]` fall within some interval from x_j to x_{j+1} associated with one of the components, then the contribution to the PDF of that component is added to the PDF associated with the marginal distribution between `supp[i]` and `supp[i + 1]`. After this is done, `["Continuous", "PDF"]` is appended to the returned value and a univariate distribution in the standard APPL list-of-three-sublists format is returned.

The first three examples illustrate `MarginalPDF` in the context of bivariate transformations.

Examples:

1. **Independent X and Y**

   Let $X \sim$ U(0, 1) and $Y \sim$ U(0, 1) be independent random variables. Consider the joint distribution of $U = X + Y$ and $V = X - Y$. The PDF of U and V is

   $$f_{U,V}(u, v) = \frac{1}{2} \qquad |u - 1| + |v| < 1.$$

   The marginal distribution of U is

   $$f_U(u) = \begin{cases} \displaystyle\int_{-u}^{u} \frac{1}{2}\, dv = u & 0 < u < 1 \\ \displaystyle\int_{u-2}^{2-u} \frac{1}{2}\, dv = 2 - u & 1 < u < 2. \end{cases}$$

   The Maple commands

   ```
   > UV := [[(u, v) -> 1 / 2],[[v > -u, v < u, v < 2 - u, v > u - 2]],
            ["Continuous", "PDF"]];
   > MarginalPDF(UV, u);
   ```

   correctly return

   $$[[u \to u, u \to 2 - u], [0, 1, 2], [''Continuous'', ''PDF'']]$$

   as the marginal PDF of U.

2. **Dependent X and Y**

   Let X and Y have joint PDF

   $$f(x, y) = 2 \qquad x > 0, y > 0, x + y < 1.$$

Consider the joint distribution of $U = X + Y$ and $V = X - Y$. The PDF of U and V is

$$f(u, v) = 1 \qquad 0 < u < 1, -u < v < u.$$

The marginal distribution of U is

$$f_U(u) \;=\; \int_{-u}^{u} 1\; dv = 2u \qquad 0 < u < 1.$$

The Maple commands

```
> UV := [[(u, v) -> 1], [[v < u, u < 1, -u < v]], ["Continuous", "PDF"]];
> MarginalPDF(UV, u);
```

correctly return

$$[[u \to 2\,u], [0, 1], [''Continuous'', ''PDF'']]$$

as the marginal PDF of U.

3. **Infinite Support**

Let $X \sim$ exponential(a) and $Y \sim$ exponential(b) be independent random variables. Consider the transformation $U = X/Y$ and $V = Y$. The joint PDF of U and V is

$$f_{U,V}(u, v) = abe^{-auv-bv} \cdot |v| \qquad u > 0,\; v > 0.$$

The marginal distribution of U is

$$f_U(u) = \int_0^{\infty} abve^{-auv-bv} dv = \frac{ab}{(au+b)^2} \qquad u > 0.$$

The Maple commands

```
> UV := [[(u, v) -> a * b * v * exp(-a * u * v - b * v)],
         [[u > 0,v > 0, u < infinity, v < infinity]], ["Continuous", "PDF"]];
> MarginalPDF(UV, u);
```

return

$$[[u \rightarrow \lim_{v\rightarrow\infty} -\frac{ab\left(e^{-auv-bv}vau + e^{-auv-bv} - 1 + e^{-auv-bv}bv\right)}{(au+b)^2}], [0, \infty], [''Continuous'', ''PDF'']]$$

as the marginal PDF of U. This limit evaluates to the desired PDF $f_U(u)$ given above.

4. **Piecewise definition of $f(x, y)$**

Consider the joint distribution of X and Y with PDF

$$f_{X,Y}(x,y) = \begin{cases} 4/13 & 0 < x < 1,\ 0 < y < 1 \\ 5/13 & 1/2 < x < 3/2,\ 1 < y < 2 \\ 3/13 & 1 < x < 2,\ 0 < y < 1 \\ 4/13 & x > 3/2,\ y > 1, y < -2x + 5. \end{cases}$$

Figure 6 shows the the support of X and Y.

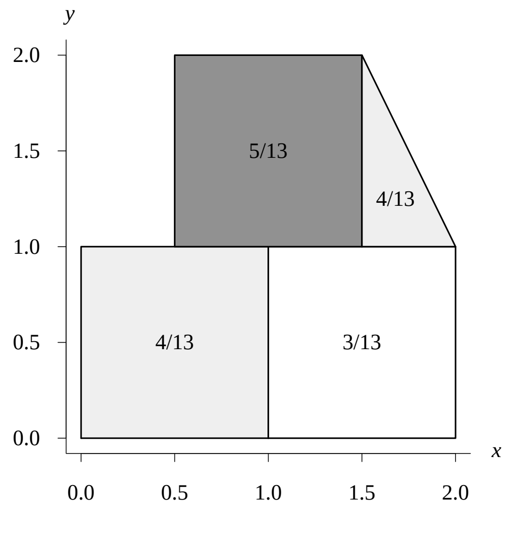

Figure 6: Support of X and Y

The Maple commands

```
> XY := [[(x, y) -> 4 / 13, (x, y) -> 5 / 13,
          (x, y) -> 3 / 13, (x, y) -> 4 / 13],
```

```
        [[y < 1, x > 0, y > 0, x < 1],
         [y > 1, x > 1 / 2, y < 2, x < 3 / 2],
         [x > 1, y > 0, x < 2, y < 1],
         [y > 1, x > 3 / 2, y < -2 * x + 5]],
        ["Continuous", "PDF"]];
> MarginalPDF(XY, x);
> MarginalPDF(XY, y);
```

return

$$[[x \to \frac{4}{13}, x \to \frac{9}{13}, x \to \frac{8}{13}, x \to \frac{19}{13} - \frac{8}{13}x], [0, 1/2, 1, 3/2, 2], [''Continuous'', \text{“}PDF''']]$$

as the marginal PDF of X and

$$[[y \to \frac{7}{13}, y \to \frac{9}{13} - \frac{2}{13}y], [0, 1, 2], [''Continuous'', \text{“}PDF''']]$$

as the marginal PDF of Y.

5. **Constraints with Multiple Intersections**

Let X and Y be uniformly distributed on the set defined by constraints $x > \sqrt{1+y^2}, x+y < 2, x-y < 2$. The marginal density of X is

$$f_X(x) = \begin{cases} \frac{2\sqrt{x^2-1}}{3/2-\sinh^{-1}(3/4)} & 1 \le x < 5/4 \\ \frac{4-2x}{3/2-\sinh^{-1}(3/4)} & 5/4 \le x < 2. \end{cases}$$

The marginal PDF is calculated by dividing the support of (X, Y) into two regions using $\sqrt{y} = x-1$. The constraint equality $x = \sqrt{1+y^2}$ intersects both $x+y = 2$ and $x-y = 2$ twice. When there is more than one intersection for adjacent constraints, `MarginalPDF` identifies the correct intersection by testing which intersection satisfies all other constraints for a particular partition. If only one such intersection exists, then that intersection is used; if more than one such intersection exists, an error message is returned. The Maple commands

```
> XY := [[(x, y) -> 1 / (3 / 2 - arcsinh(3 / 4)),
          (x, y) -> 1 / (3 / 2 - arcsinh(3 / 4))],
         [[y > sqrt(x ^ 2 - 1), x + y < 2, sqrt(y) > x - 1],
          [y > -sqrt(x ^ 2 - 1), sqrt(y) < x - 1, x + y < 2, x - y < 2]],
         ["Continuous", "PDF"]];
> MarginalPDF(XY,x);
```

correctly return

$$[[x \mapsto -4\,\frac{\sqrt{x^2-1}}{2\,arcsinh\,(3/4)-3}, x \mapsto \frac{4\,x-8}{2\,arcsinh\,(3/4)-3}], [1.0, 1.250000000, 2.0], [\text{“}Continuous''\text{, “}PDF''\text{]]}$$

as the marginal PDF of X.

D. VerifyJointPDF

Syntax: The statement

```
VerifyJointPDF(XY);
```

displays whether the joint PDF of X and Y is a valid PDF.

Purpose: `VerifyJointPDF` ensures that $f_{X,Y}(x, y)$ is a valid PDF by checking that

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_{X,Y}(x, y)\, dy\, dx = 1$$

and $f_{X,Y}(x, y) \geq 0$ for all x, y in the support of X and Y.

Algorithm: `VerifyJointPDF` calculates

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_{X,Y}(x, y)\ dy\ dx \qquad \text{and} \qquad \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} |f_{X,Y}(x, y)|\ dy\ dx.$$

Equality of these integrals ensures that $f_{X,Y}(x, y)$ is positive for all values in the support of X and Y. If both integrals also evaluate to one, then the two conditions for a valid PDF are met, and $f_{X,Y}(x, y)$ is a valid PDF. If the integrals cannot be evaluated symbolically, `VerifyJointPDF` invokes Maple numerical procedures to approximate the values of the integrals.

Examples:

1. The joint PDF of two independent U(0, 1) random variables represented by

```
> XY := [[(x, y) -> 1], [[y > 0, x > 0, y < 1, x < 1]], ["Continuous", "PDF"]];
```

is verified as a valid PDF via `VerifyJointPDF(XY);`.

2. The PDF of the dependent random variables X and Y represented by

```
> XY := [[(x, y) -> x / y ^ 2 + 1],[[y > x, x > 0, y < 1]],
        ["Continuous", "PDF"]];
```

is verified as a valid PDF via `VerifyJointPDF(XY);`.

E. `BiExpectedValue`

Syntax: The Maple statement

```
BiExpectedValue(XY, g(x, y));
```

returns $E[g(X,Y)]$, the expected value of $g(X,Y)$.

Purpose: `BiExpectedValue` computes the expected value of $g(X,Y)$, where the joint PDF of X and Y is in the list-of-lists format and $g(X,Y)$ is a function of X and Y.

Algorithm: `BiExpectedValue` calculates

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x,y) f_{X,Y}(x,y)\ dy\ dx.$$

Example:

1. The joint PDF of two independent U(0, 1) random variables is represented by

```
XY := [[(x, y) -> 1], [[y > 0, x > 0, y < 1, x < 1]], ["Continuous", "PDF"]];
```

The Maple statement

```
> BiExpectedValue(XY, x + y);
```

returns $E[g(X, Y)] = E[X + Y] = 1$.

F. Covariance and Correlation

Syntax: The procedure `Covariance(XY)` returns the covariance of X and Y. The procedure `Correlation(XY)` returns the correlation of X and Y.

Purpose: `Covariance` and `Correlation` compute the covariance and correlation of X and Y, given that the joint distribution of X and Y is entered in the list-of-lists format.

Algorithm: The covariance of X and Y is given by the formula $\text{Cov}(X, Y) = E[XY] - E[X]E[Y]$. The procedure `Covariance(XY)` finds the means of the marginal distributions X and Y and subtracts their product from $E[XY]$. `Covariance` uses the procedures `BiExpectedValue` and `MarginalPDF`. The correlation of X and Y is equal to $\frac{\text{Cov}(X,Y)}{\sigma_X \sigma_Y}$. `Correlation` uses the APPL procedure `Variance` to calculate σ_X and σ_Y.

Examples:

1. The joint PDF $f_{X,Y}(x, y)$ of two independent U(0, 1) random variables is represented by

   ```
   > XY := [[(x, y) -> 1], [[y > 0, x > 0, y < 1, x < 1]], ["Continuous", "PDF"]];
   ```

   The Maple statements

   ```
   > Covariance(XY);
   > Correlation(XY);
   ```

   both return 0.

2. The joint PDF $f_{X,Y}(x, y)$ of dependent random variables X and Y is represented by

   ```
   > XY := [[(x, y) -> 2], [[x > 0, y > 0, x + y < 1]], ["Continuous", "PDF"]];
   ```

   The Maple statements

   ```
   > Covariance(XY);
   > Correlation(XY);
   ```

return $-1/36$ and $-1/2$ respectively.

3. We apply the bivariate transformation technique to a simple example from queueing theory. Consider an M/M/1 queue that consists of a single server, exponential interarrival times with rate λ, and exponential service times with rate μ. Let T_1 and T_2 denote the sojourn times (times in the system) of the first two customers arriving to an empty system. Kaczynski et al. (2010) give the joint PDF of T_1 and T_2 as

$$f_{T_1,T_2}(t_1,t_2) = \begin{cases} \dfrac{\mu^2\left(\lambda e^{-\mu t_2} + \mu e^{-\lambda t_1 - \mu t_1 - \mu t_2}\right)}{\lambda+\mu} & 0 < t_1 \le t_2 \\ \dfrac{\mu^2\left(\lambda e^{-\lambda t_1 - \mu t_1 + \lambda t_2} + \mu e^{-\lambda t_1 - \mu t_1 - \mu t_2}\right)}{\lambda+\mu} & 0 < t_2 < t_1. \end{cases}$$

Our goal is to calculate the covariance between T_1 and T_2. The statements to establish the joint PDF of T_1 and T_2 and calculate their covariance are

```
> assume(lambda > 0);
> assume(mu > 0);
> top := mu ^ 2 * (lambda * exp(-mu * t2) + mu *
         exp(-lambda * t1 - mu * t1 - mu * t2)) / (lambda + mu);
> bot := mu ^ 2 * (lambda * exp(-lambda * t1 - mu * t1 + lambda * t2) + mu *
         exp(-lambda * t1 - mu * t1 - mu * t2)) / (lambda + mu);
> TT  := [[(t1, t2) -> top, (t1, t2) -> bot],
          [[t1 > 0, t1 < t2, t2 < infinity], [t1 > t2, t1 < infinity, t2 > 0]],
          ["Continuous", "PDF"]];
> Covariance(TT);
```

which return

$$\text{Cov}(T_1, T_2) = \frac{\lambda(\lambda+2\mu)}{(\lambda+\mu)^2\mu^2}.$$

References

Glen, A. G., D. L. Evans, L. M. Leemis. 2001. APPL: A Probability Programming Language. *The American Statistician* **55** 156–166.

Glen, A. G., J. H. Drew, L. M. Leemis. 1997. A Generalized Univariate Change-of-Variable Transformation Technique. *INFORMS Journal on Computing* **9** 288–295.

Hogg, R. V., J. W. McKean, A. T. Craig. 2005. *Introduction to Mathematical Statistics*, Sixth Edition. Prentice Hall, New Jersey.

Kaczynski, W. H., L. M. Leemis, J. H. Drew. 2010. Transient Queueing Analysis. Technical Report, Department of Mathematics, The College of William & Mary.

Rose, C., M. D. Smith. 2002. *Mathematical Statistics With Mathematica*. Springer–Verlag, New York.