



Revista UIS ingenierías

ISSN: 1657-4583

ISSN: 2145-8456

Universidad Industrial de Santander

Vargas-Sánchez, Jhon Jairo; Soto-Mejía, José Adalberto
Generador de variable aleatoria bivariado aplicado a DEA
Revista UIS ingenierías, vol. 20, núm. 2, 2021, Abril-Junio, pp. 139-150
Universidad Industrial de Santander

DOI: <https://doi.org/10.7440/res64.2018.03>

Disponible en: <https://www.redalyc.org/articulo.oa?id=553770380012>

- Cómo citar el artículo
- Número completo
- Más información del artículo
- Página de la revista en redalyc.org

UAEH  redalyc.org

Sistema de Información Científica Redalyc
Red de Revistas Científicas de América Latina y el Caribe, España y Portugal
Proyecto académico sin fines de lucro, desarrollado bajo la iniciativa de acceso
abierto

Generador de variable aleatoria bivariado aplicado a DEA

Generator of stochastic bivariate variable applied to stochastic Bayesian DEA

Jhon Jairo Vargas-Sánchez ¹, José Adalberto Soto-Mejía ²

¹ Grupo de Investigación Estadística y Métodos Cuantitativos, Facultad Ingeniería, Universidad del Magdalena, Colombia. Orcid: 0000-0003-3799-9850. Correo electrónico: jvargass@unimagdalena.edu.co

² Grupo de Investigación Análisis Envolvente de Datos, Facultad Ingeniería Industrial, Universidad Tecnológica de Pereira, Colombia. Orcid: 0000-0002-0205-6863. Correo electrónico: jomejia@utp.edu.co

Recibido: 10 octubre, 2020. Aceptado: 7 diciembre, 2020. Versión final: 10 febrero, 2021.

Resumen

La aplicación de modelos DEA estocásticos utilizando Estadística Bayesiana es una técnica que está surgiendo en los últimos años, para ser usada se debe hacer simulación de distribuciones de probabilidad multivariadas. En este artículo se presenta un generador bivariado para variable continua. Adicionalmente se ha creado una discretización de este para lograr simulaciones de distribuciones a posteriori de una manera fácil de aplicar. Se presentará la aplicación del generador a dos funciones de densidad de probabilidad bivariadas, una de ellas normal, con sus respectivas pruebas de bondad de ajuste. Se usaron datos del sector educativo en base de datos del DANE (Departamento Administrativo Nacional de Estadística) Colombia para resolver el problema de un modelo DEA estocástico Bayesiano. Los resultados muestran la utilidad, potencia y facilidad del uso del generador propuesto en este tipo de problema.

Palabras clave: análisis envolvente de datos; DEA Bayesiano; DEA estocástico; eficiencias; distribución normal multivariada; generador bivariado; sector educativo; simulación; funciones de densidad de probabilidad; programación no lineal; conos convexos; optimización.

Abstract


The application of stochastic DEA models using Bayesian Statistics is a technique that has been emerging in recent years, to be used it, simulation of multivariate probability distributions is necessary. A bivariate generator for continuous variable will be presented. Additionally, a discretization has been created on it to achieve simulations of a posteriori distributions with easy way to apply it. Application of the generator to two functions of bivariate probability density will be presented, one of them normal, with their respective tests of goodness of fit. Data from the education sector was used in the database of the DANE (National Administrative Department of Statistics) Colombia to solve the problem of a Bayesian stochastic DEA model. The results show the utility, power and easy implementation of the generator proposed in this type of problem.

Keywords: data envelopment analysis; Bayesian DEA; stochastic DEA; efficiencies; multivariate normal distribution; bivariate generator; education sector; simulation; probability density functions; nonlinear programming; convex cones; optimization.

1. Introducción

DEA (por sus siglas en inglés “Data Envelopment Analysis”) es una técnica que ha sido ampliamente usada para evaluar el desempeño de diferentes sistemas

organizacionales. DEA es una técnica no paramétrica usada para evaluar las eficiencias relativas de un conjunto de DMU's (Decision Making Units). Una variedad de modelos DEA han sido desarrollados para satisfacer

ISSN impreso: 1657 - 4583. ISSN en línea: 2145 - 8456, CC BY-ND 4.0 

Como citar: J. J. Vargas-Sánchez, J. A. Soto-Mejía, “Generador de variable aleatoria bivariado aplicado a DEA estocástico Bayesiano,” *Rev. UIS Ing.*, vol. 20, no. 2, pp. 139-150, 2021, doi: [10.18273/revuin.v20n2-2021012](https://doi.org/10.18273/revuin.v20n2-2021012)

ciertas situaciones y características de las variables [1]. Una característica estudiada es la variabilidad en los datos, dentro de los cuales se encuentran los modelos DEA estocásticos que buscan incorporar variabilidad e incertidumbre en los modelos DEA determinísticos [2].

Como una continuación de los trabajos estocásticos se considera el esfuerzo por realizar inferencia estadística en DEA. Históricamente aparece el concepto de Bootstrap aplicado a DEA en Simar y Wilson [3] y Simar y Wilson [4]. Esta técnica consiste en un procedimiento iterativo para generar, de un modelo dado, una simulación de un gran conjunto de datos que podrían haber sido observados. Una discusión de la inferencia estadística y eficiencia no paramétrica se puede ver en Dyson y Shale [5]. Otra comparación de DEA estocástico y Bootstrap en DEA lo realizan Ceyhan y Banneyan [6].

Una fuerte crítica al método Bootstrap en DEA se ha hecho en Tsionas y Papadakis [7] con el argumento de que en la técnica bootstrap no hay garantía de buen desempeño en la búsqueda de inferencia porque su utilización ha sido para DEA no estocástico y que las muestras generadas por bootstrap nunca fueron observadas.

Debido a lo anterior, Tsionas y Papadakis [7] han propuesto una metodología Bayesiana que permite que la simulación logre realizar inferencia en los puntajes de eficiencia (score). Siguiendo al artículo últimamente nombrado, Mitropoulos et al. [8] han propuesto un método que da nuevas oportunidades al DEA estocástico. Su inspiración se ha basado en el trabajo de Cooper et al. [9] quienes aseguran que la metodología Bayesiana en DEA estocástico puede ir más allá en las características de eficiencia usando dos indicadores de probabilidad. El primero indica si una DMU es completamente eficiente o suficientemente eficiente. Segundo, basado en la distribución marginal de los puntajes se construyen finitas muestras con intervalos de probabilidad Bayesianos. Otros trabajos recientes en DEA Bayesiano se pueden encontrar en Friesner et al. [10].

El presente artículo pretende aportar en las incursiones de DEA estocástico Bayesiano, mostrando formas prácticas de aplicarlo y que sean fácilmente adaptables. Se propone en el presente artículo un generador bivariado que se puede adaptar a esta metodología DEA, pues esta exige que una generación de números aleatorios sea llevada a cabo de una distribución a posteriori. El resto del artículo está organizado como sigue. En la sección 2 se revisan los conceptos de DEA y DEA estocástico. En la sección 3 el método Bayesiano de Tsionas y Papadakis [7]. En la sección 4 el generador bivariado propuesto. En la sección

5 aplicamos el generador al método Bayesiano usando datos del DANE Colombia y encontrando intervalos de confianza para las eficiencias, demostrando que las soluciones conseguidas son óptimos globales. En la sección 6 las conclusiones.

2. DEA estocástico

DEA estima la medida de eficiencia de una DMU específica por comparación relativa de su desempeño con las otras. DEA asume que hay N DMUs ($j = 1, 2, \dots, N$). El desempeño de cada DMU es caracterizada por su proceso de producción de m entradas (x_{lj} para $l = 1, \dots, m$) para producir S salidas (y_{rj} para $r = 1, \dots, S$).

Uno de los métodos clásicos para tratar la incertidumbre en DEA es el CCP (por sus siglas en inglés “chance-constrained programming”), es ampliamente tratado en Land et al. [12, 13, 14, 15]. El modelo (1) es un CCP extendido del CCR (Charnes et al. [11]) en su forma primal.

$$\begin{aligned} & \max P \left(\frac{\sum_{r=1}^S u_r \widetilde{y_{ro}}}{\sum_{i=1}^m v_i \widetilde{x_{io}}} \geq \beta_0 \right) \\ & \text{s. t. } P \left(\frac{\sum_{r=1}^S u_r \widetilde{y_{rj}}}{\sum_{i=1}^m v_i \widetilde{x_{ij}}} \leq \beta_j \right) \geq 1 - \alpha_j, \\ & \quad j = 1, 2, \dots, n, \\ & \quad u_r, v_i \geq \forall r, i. \end{aligned} \quad (1)$$

El modelo (1) está basado en los multiplicadores u y v los cuales son variables que hacen que se encuentre la máxima $P \left(\frac{\sum_{r=1}^S u_r \widetilde{y_{ro}}}{\sum_{i=1}^m v_i \widetilde{x_{io}}} \geq \beta_0 \right)$, si una DMU es ineficiente no es porque se hayan asignado mal sus multiplicadores sino porque en verdad es ineficiente.

Se puede consultar en Cooper et al. [16] cómo el modelo (1) se transforma en su equivalente determinístico (2).

$$\begin{aligned} & \max \gamma_o \\ & \text{s. t. } u^T \bar{y}_o - \beta_0 v^T x_o \geq \phi^{-1}(\gamma_o) \sqrt{u^T \sum_j u} \\ & \quad \eta_j + u^T y_j - \beta_j v^T x_j \leq 0 \\ & \quad \eta_j^2 - K_{(1-\alpha_j)}^2 u^T \sum_j u \geq 0 \\ & \quad u, v \geq 0, \eta_j \geq 0, j = 1, 2, \dots, n. \\ & \quad 0 \leq \gamma_o \leq 1 \end{aligned} \quad (2)$$

En modelo (2), $K_{(1-\alpha_j)} = \Phi^{-1}(1 - \alpha_j)$ y Φ^{-1} es la función fractile asociada con la distribución normal estándar. Σ_j es la matriz de covarianza y las η_j son llamadas “splitting variables”.

Sin embargo, para la aplicación del generador propuesto a DEA estocástico Bayesiano, en el presente artículo la postura parte del modelo CCP dual definido en Subhash [17]:

$$\begin{aligned} & \text{máx} \phi \\ \text{s. t } & \Pr \left\{ \sum_{j=1}^N \lambda_j y_j \geq \phi y_0 \right\} \geq (1 - \alpha); \forall \\ & l = 1, \dots, m \\ & \sum_{j=1}^N \lambda_j x_j \leq x_0; \forall r = 1, \dots, S \\ & \sum_{j=1}^N \lambda_j = 1; \lambda_j \geq 0 \quad (j = 1, 2, \dots, N). \end{aligned} \quad (3)$$

En el modelo (3), ϕ es la medida de eficiencia o score, los x_j son los valores de la variable de entrada en la DMU j , y_j son los valores de la variable de salida en la DMU j ; los λ_j son valores que hacen que se maximice ϕ y son valores que ayudan a construir la frontera de eficiencia; x_0 y y_0 son los valores de entrada y salida respectivamente para la DMU observada; N simboliza el número de DMUs, así que este modelo se corre N veces para hallar las eficiencias relativas de todas las DMUs.

Este modelo se transforma a su equivalente determinístico a través de la creación de una nueva variable aleatoria u usando ecuación (4) la cual se distribuye normalmente con parámetros dados en ecuaciones (5) y (6). Dicha transformación se puede consultar en Subhash [17].

$$u = \sum_{j=1}^N \lambda_j y_j - \phi y_0 \quad (4)$$

$$E(u) = \sum_{j=1}^N \lambda_j \mu_j - \phi \mu_0 \equiv \mu_u \quad (5)$$

$$\text{Var}(u) = \sum_{j=1, j \neq 0}^N \lambda_j^2 \sigma_j^2 + (\lambda_0 - \phi)^2 \sigma_0^2 \equiv \sigma_u^2 \quad (6)$$

Continuando con la transformación y haciendo reemplazos correspondientes se llega al modelo (7).

$$\begin{aligned} & \text{máx} \phi \\ \text{s. t } & \Pr(u \geq 0) \geq 1 - \alpha \end{aligned} \quad (7)$$

$$\sum_{i=1}^N \lambda_i x_i \leq x_0.$$

$$\begin{aligned} & \text{máx} \phi \\ \text{s. t } & \Pr \left(z \geq \frac{\mu_u}{\sigma_u} \right) \geq 1 - \alpha \\ & \sum_{i=1}^N \lambda_i x_i \leq x_0 \end{aligned} \quad (8)$$

De esta forma el modelo CCP dado en (3), que es dual, se transforma en su equivalente determinístico (9).

$$\begin{aligned} & \text{máx} \phi \\ \text{s. t } & \sum_{j=1}^N \lambda_j \mu_j - \phi \mu_0 \\ & \geq 1.96 \sqrt{\sum_{j=1, j \neq 0}^N \lambda_j^2 \sigma_j^2 + (\lambda_0 - \phi)^2 \sigma_0^2}, \\ & s = 1, \dots, S \\ & \sum_{j=1}^N \lambda_j x_j \leq x_0; m = 1, \dots, M \\ & \sum_{j=1}^N \lambda_j = 1; \lambda_j \geq 0, j = 1, 2, \dots, N \end{aligned} \quad (9)$$

3. Métodos Bayesianos en DEA estocástico

En Tsionas y Papadakis [7] se clarifica la necesidad de un modelo estadístico en el modelo DEA estocástico a usar. En este artículo se menciona la metodología para resolver un DEA Bayesiano. Se parte de la naturaleza aleatoria de las variables dada en ecuación (10).

$$Z_i \equiv \begin{bmatrix} y_i \\ x_i \end{bmatrix} \sim N(\mu, \Sigma), i = 1, 2, \dots, N \quad (10)$$

Donde μ es un vector de medias de tamaño $(M+K) \times 1$ y Σ es la matriz de covarianza de tamaño $(M+K) \times (M+K)$ y los Z_i se asumen independientes. M el número de salidas, K el número de entradas y N es el número de DMUs. La función de verosimilitud está dada por (11).

$$L(\theta; Z) = \prod_{i=1}^N f(Z_i | \theta) \quad (11)$$

Al asumir una probabilidad a priori $p(\theta)$ para el vector de parámetros y usar el teorema de Bayes la distribución a posteriori queda determinada por la expresión (12).

$$p(\theta|Z) \propto L(\theta; Z) \cdot p(\theta) \quad (12)$$

Se define un operador $T : Z \rightarrow \Omega \subseteq \mathbb{R}^s$ el cual va del espacio muestral a un espacio s dimensional. El operador produce un conjunto de resultados $r(Z; \theta) \subseteq \mathbb{R}^s$ que puede abarcar muchas medidas, pero en este caso interesa que $s = 1$, y así contenga el espacio de las medidas de eficiencia.

A diferencia del bootstrap, DEA estocástico Bayesiano no usa la variabilidad de la muestra para justificar la naturaleza aleatoria de $r(Z; \theta)$, para ello usa la distribución a posteriori $p(\theta|Z)$ para obtener $p(r|Z)$. Esto lo logra mediante tres pasos. Primero obtener un vector θ^j de la distribución posterior $p(\theta|Z)$. Segundo usar el operador T para obtener el respectivo r^j . Tercero, aproximar la distribución $p(r|Z)$ por técnicas de densidad de Kernel.

A continuación, se presenta la forma como se obtiene la distribución a posteriori para lograr el primer paso. Basados en la literatura DEA estocástico la matriz Σ está dada por

$$\Sigma = \begin{bmatrix} \Sigma_Y & 0 \\ 0 & \Sigma_X \end{bmatrix} \quad (13)$$

donde Σ_Y es de tamaño $M \times M$ y Σ_X es de tamaño $K \times K$. Se tiene que $y_i \sim N(\mu_Y, \Sigma_Y)$ y $x_i \sim N(\mu_X, \Sigma_X)$. La verosimilitud para (μ_Y, Σ_Y) está dada por

$$\begin{aligned} L(\mu_Y, \Sigma_Y) &\propto |\Sigma_Y|^{-\frac{N}{2}} \exp \left[-\frac{1}{2} \sum_{i=1}^N (y_i - \mu_Y)' \Sigma_Y^{-1} (y_i - \mu_Y) \right] \\ &= |\Sigma_Y|^{-\frac{N}{2}} \exp \left[-\frac{1}{2} \text{tr} \Sigma_Y^{-1} S_Y \right], \end{aligned} \quad (14)$$

donde tr es la traza y $S_Y = \sum_{i=1}^N (y_i - \mu_Y)(y_i - \mu_Y)'$. Una recomendación en Tsionas y Papadakis [7] es usar como a priori $p(\mu_Y, \Sigma_Y) \propto |\Sigma_Y|^{-\frac{M+1}{2}}$ lo cual permite llegar a la distribución a posteriori dada en

$$\begin{aligned} p(\mu_Y, \Sigma_Y|Y) &\propto |\Sigma_Y|^{-(N+M+1)/2} \exp \left[-\frac{1}{2} \text{tr} \Sigma_Y^{-1} S_Y \right] \end{aligned} \quad (15)$$

La marginal de μ_Y en ecuación (15) es de la forma Student multivariada y la de Σ_Y es de la forma Wishart. La posterior para $p(\mu_Y, \Sigma_X|X)$ es de la misma estructura y forma que ecuación (15) pero con variable X y parámetro K en lugar de M .

4. Metodología

En esta sección se presenta el planteamiento de un generador bivariado que permita simular una distribución bivariada dada. Este planteamiento se había iniciado en [18] para variable aleatoria continua, mas no para variable aleatoria discreta. La idea básica nace en los conceptos de gradiente y derivadas direccionales.

4.1. Fundamentos teóricos

El gradiente y la derivada direccional son los fundamentos matemáticos y base para la propuesta del generador bivariado. Se parte del hecho de que la derivada direccional de una distribución de probabilidad acumulada $F(x,y)$ nos indica la velocidad a la que se acumula probabilidad en un punto (x,y) en una dirección dada por un vector unitario U . Para el caso univariado, si en una función de densidad $f(x)$ un valor x_1 tiene mayor densidad de probabilidad que un valor x_2 entonces se cumple que $\frac{dF(x_1)}{dx} > \frac{dF(x_2)}{dx}$. Es claro que si se toma un incremento $\Delta x = x_2 - x_1$ y se calcula la pendiente $\frac{F(x_2) - F(x_1)}{\Delta x}$, estaríamos observando la velocidad a la que se acumula probabilidad en un intervalo Δx . Por ejemplo en la distribución normal, si se parte de la media μ , y formamos un intervalo $(\mu, \mu + \Delta x)$ se espera más tasa de acumulación de probabilidad en este intervalo que en otro intervalo, digamos $(P_{95}, P_{95} + \Delta x)$. Dicho de otra manera, $\frac{F(\mu + \Delta x) - F(\mu)}{\Delta x} > \frac{F(P_{95} + \Delta x) - F(P_{95})}{\Delta x}$, siendo P_{95} el percentil 95. Este concepto es llevado al caso bivariado y se propone un algoritmo a partir de esta idea. Dicho algoritmo será capaz de generar números aleatorios que sigan una distribución de probabilidad bivariada. Si f es una función en dos variables x y y , y las derivadas parciales f_x y f_y existen, entonces el gradiente de f denotado ∇f está definido por la ecuación (16).

$$\nabla f(x, y) = f_x(x, y)i + f_y(x, y)j, \quad (16)$$

Y la derivada direccional se puede escribir como el producto punto entre un vector unitario y el vector gradiente.

$$D_U f(x, y) = U \cdot \nabla f(x, y), \quad (17)$$

donde $f_x(x, y) = \frac{\partial f}{\partial x}$ es la derivada parcial de la función $f(x, y)$ y describe la variación de f en la dirección positiva del eje x ; $f_y(x, y) = \frac{\partial f}{\partial y}$ es la derivada parcial de la función $f(x, y)$ y describe la variación de f en la dirección positiva del eje y ; $U = \cos\theta i + \sin\theta j$ representa el vector unitario que forma un ángulo de θ radianes con la parte

positiva del eje x ; y $D_U f(x, y) = f_x(x, y)\cos\theta + f_y(x, y)\sin\theta$ representa la derivada direccional de f en la dirección U . Una derivada direccional de una función diferenciable puede obtenerse mediante el producto punto del gradiente y un vector unitario de la dirección deseada, ecuación (17). El valor máximo de $D_U f(x_0, y_0)$ ocurre cuando $\cos\alpha = 1$ o cuando $\alpha = 0$ en la ecuación (18).

$$\begin{aligned} D_U f(x_0, y_0) &= U \cdot \nabla f(x_0, y_0) \\ &= \|u\| \|\nabla f(x_0, y_0)\| \cos\alpha \\ &= \|\nabla f(x_0, y_0)\| \cos\alpha \end{aligned} \quad (18)$$

El algoritmo consiste en conseguir direcciones aleatorias volviendo a U un vector aleatorio. Podemos generar aleatoriamente puntos (x, y) con $x \in X$ y $y \in Y$ y calcular las derivadas direccionales máximas aplicando $\|\nabla F(x, y)\|$ y obtener mediante procesos iterativos dicha derivada direccional máxima. Una vez obtenida D_U máx se puede encontrar la derivada direccional en un punto aleatorio (x, y) con la dirección del vector aleatorio U y hacer la relación $\frac{D_U F(x, y)}{D_U \text{ máx}}$ la cual es menor a 1 y x e y son valores aleatorios generados por ecuaciones (19) y (20), donde los respectivos R_i son variables aleatorias uniformes en el intervalo $(0, 1)$.

$$x = a + (b - a) * R_1 \quad (19)$$

$$y = c + (d - c) * R_2 \quad (20)$$

En las ecuaciones (19) y (20) a y b son, respectivamente, el límite inferior y superior de la variable aleatoria X ; c y d son, respectivamente, el límite inferior y superior de la variable aleatoria Y . Un vector aleatorio U será generado con ecuaciones (21), (22) y (23).

$$\cos\theta = R_3 \quad (21)$$

$$\sin\theta = \sqrt{1 - \cos^2\theta} = \sqrt{1 - R_3^2} \quad (22)$$

$$U = \cos\theta i + \sin\theta j \quad (23)$$

4.2. Fundamentos teóricos

- Paso 1. Encontrar la función de distribución mediante $F(x, y) = \int_{-\infty}^y \int_{-\infty}^x f(t, u) dt du$
- Paso 2. Calcular $D_U F(x, y)$ máxima usando $D_U F(x, y) = \|\nabla F(x, y)\|$ y ecuaciones (19) y (20).
- Paso 3. Hacer $G(x, y, R_3) = D_U F(x, y) = U \cdot \nabla F(x, y)$
- Paso 4. Hacer $R_4 = \frac{D_U F(x, y)}{D_U \text{ máx}} = \frac{G(x, y, R_3)}{D_U \text{ máx}}$

- Paso 5. En paso 4 sustituir $y = c + (d - c) * R_5$ y despejar x , donde $x = H(R_3, R_4, R_5)$
- Paso 6. Generar R_3, R_4 y R_5 , sustituir en $x = H(R_3, R_4, R_5)$ y obtener la pareja $(x, y) = (H(R_3, R_4, R_5), c + (d - c) * R_5)$

Al repetir este algoritmo se obtienen parejas aleatorias (x, y) que siguen una función de densidad de probabilidad $f(x, y)$

5. Resultados

5.1. Uso del algoritmo en una función bivariada

Supóngase que se van a generar números aleatorios que sigan la función dada por la ecuación (24).

$$f(x, y) = \begin{cases} \frac{x(1 + 3y^2)}{4} & \text{si } 0 < x < 2, 0 < y < 1 \\ 0 & \text{de otra forma} \end{cases} \quad (24)$$

A continuación, se presentan los pasos resueltos después de los desarrollos algebraicos.

- Paso 1. $F(x, y) = \frac{1}{8} x^2 y (y^2 + 1)$

- Paso 2.

$$\|\nabla F(x, y)\| = \sqrt{\frac{x^2 y^2 (1 + y^2)^2}{16} + \frac{9x^4 y^4}{64} + \frac{3}{32} x^4 y^2 + \frac{x^4}{64}}$$

Mediante un código construido en el software Matlab se encontró un valor máximo de $D_U \text{ máx} = 2,228737$

- Paso 3. Hacer $G(x, y, R_3)$

$$D_U F(x, y) = \left(\frac{R_3 x y (y^2 + 1)}{4} + \sqrt{1 - R_3^2} \left(\frac{3}{8} x^2 y^2 + \frac{1}{8} x^2 \right) \right)$$

- Paso 4. Hacer $R_4 = \frac{D_U F(x, y)}{D_U \text{ máx}} = \frac{G(x, y, R_3)}{D_U \text{ máx}}$

$$R_4 = \frac{\left(\frac{R_3 x y (y^2 + 1)}{4} + \sqrt{1 - R_3^2} \left(\frac{3}{8} x^2 y^2 + \frac{1}{8} x^2 \right) \right)}{D_U \text{ máx}}$$

- Paso 5.

$$x = \frac{\sqrt{D_U \text{ máx} R_4 + \frac{R_3^2 R_5^2 (R_5^2 + 1)^2}{64 \left(\frac{3}{8} R_5^2 + \frac{1}{8} \right) \sqrt{1 - R_3^2}}} - \frac{R_3 R_5 (R_5^2 + 1)}{8 \sqrt{1 - R_3^2 \left(\frac{3}{8} R_5^2 + \frac{1}{8} \right) \sqrt{1 - R_3^2}}}}{\sqrt{\left(\frac{3}{8} R_5^2 + \frac{1}{8} \right) \sqrt{1 - R_3^2}}}$$

- Paso 6. Generar R_3, R_4 y R_5 , sustituir en $x = H(R_3, R_4, R_5)$ y obtener la pareja $(x, y) = (H(R_3, R_4, R_5), c + (d - c) * R_5)$. Con este paso se

generaron 3586 parejas de números aleatorios. La forma como se distribuyeron se encuentra en tabla 1.

Las integrales dobles se consiguieron para revisar la frecuencia esperada a la luz de ecuación (24) y se usaron junto con las frecuencias de la tabla 1 para aplicar el estadístico $\Sigma \frac{(FO-FE)^2}{FE}$ el cual dio un resultado de 21.7831 que comparado con $\chi_{0,05,16}^2 = 26,296$ indica que los números generados se distribuyen como (24).

5.2. Uso del algoritmo en una función bivariada

En esta subsección se presenta una adaptación del generador mostrado en [18]. Debido a que en muchas ocasiones es difícil obtener la integral doble en el paso 1 del algoritmo y tampoco resulta factible despejar x en forma explícita en el paso 5, se ha extendido el generador propuesto al caso discreto. Esta discretización del algoritmo propuesto se ha aplicado a una distribución normal bivariada.

Simulamos 100 vectores bivariados aleatorios que sigan la distribución $X \sim N(\mu, \Sigma)$, donde $\mu = (3,9658, 7,7132)$ y $\Sigma = \begin{pmatrix} 0,6621 & 0,8 \\ 0,8 & 4,0494 \end{pmatrix}$.

Paso 1 y 2. Estos pasos se resolvieron de la siguiente forma. Lo que se hizo fue elegir valores posibles x_1 e x_2 que pertenezcan al dominio de la función. Se eligió un paso pequeño de 0.1, con este construimos una matriz de puntos bivariados, como una malla de puntos en una cuadrícula. Mediante una estructura de programación “for anidado” se fueron encontrando los volúmenes bajo la curva estableciendo las alturas de acuerdo con la función bivariada. Los volúmenes se acumularon para encontrar una función de distribución aproximada. Los gradientes se construyeron de acuerdo con el *paso* = 0,1 y a la nueva función acumulada discretizada. La tabla 2 presenta en las columnas 2 y 3 x_1 y x_2 respectivamente.

En dicha tabla se presentan los registros 1500 a 1510 solamente, ya que esta tiene 6222 registros. La columna 4 corresponde a las probabilidades acumuladas o los volúmenes bajo la curva acumulados.

Tabla 1. Frecuencias observadas en la generación bivariada de la función de la ecuación (24)

Número	$0 < Y < 0,2$	$0,2 < Y < 0,4$	$0,4 < Y < 0,6$	$0,6 < Y < 0,8$	$0,8 < Y < 1$
$0 < X < 0,4$	16	13	23	28	45
$0,4 < X < 0,8$	56	47	69	105	141
$0,8 < X < 1,2$	74	90	118	165	262
$1,2 < X < 1,6$	120	128	170	256	332
$1,6 < X < 2$	129	165	218	366	450

Fuente: elaboración propia.

Tabla 2. Discretización del algoritmo propuesto

Registro	x_1	x_2	Prob Acum	Gradiente x_1	Gradiente x_2	Normas	$D_U F(x_1, x_2)$	$\frac{D_U F(x_1, x_2)}{D_U \text{máx}}$
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1500	2.7	5.1	0.0553	0.1238	0.0027	0.1238	0.0136	0.2653
1501	2.7	5.2	0.0556	0.1242	0.0028	0.1242	0.0137	0.2698
1502	2.7	5.3	0.0559	0.1246	0.0029	0.1247	0.0138	0.2622
1503	2.7	5.4	0.0562	0.1251	0.0030	0.1251	0.0139	0.2645
1504	2.7	5.5	0.0565	0.1256	0.0031	0.1256	0.0141	0.2667
1505	2.7	5.6	0.0568	0.1261	0.0031	0.1261	0.0142	0.2687
1506	2.7	5.7	0.0572	0.1266	0.0032	0.1267	0.0143	0.2707
1507	2.7	5.8	0.0575	0.1272	0.0032	0.1272	0.0144	0.2724
1508	2.7	5.9	0.0578	0.1278	0.0033	0.1278	0.0145	0.2740
1509	2.7	6.0	0.0581	0.1283	0.0033	0.1284	0.0145	0.2755
1510	2.7	6.1	0.0585	0.1289	0.0033	0.1290	0.0146	0.2767
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Fuente: elaboración propia.

Paso 3, 4, 5 y 6. Se continúa construyendo la tabla 2. Las columnas 5 y 6 corresponde al componente gradiente de x_1 y componente gradiente en x_2 respectivamente. La columna 7 es la norma del gradiente construido de las anteriores dos columnas. La columna 8 es la derivada direccional. La columna 9 el cociente entre derivada direccional y derivada direccional máxima. Una vez se tiene la tabla 2 lo que sigue es ordenar la tabla completa por la columna 9, de esta forma es fácil generar un número aleatorio que recorra la tabla hasta que este sea menor a algún valor de esta última columna ordenada de menor a mayor, de acuerdo con este resultado se selecciona el vector (x_1, x_2) respectivo.

Las figuras 1 y 2 muestran las pruebas Mardia y Henze-Zirkler de bondad de ajuste normal multivariadas corridas en el Rcomander [19]. Un diagrama Q-Q fue construido en Matlab en la figura 3, basado en la ecuación (25), donde se puede mostrar que si los Y_i siguen una distribución normal multivariada, u_i en ecuación (26) tiene distribución Beta con $\alpha = p/2$ y $\beta = (n - p - 1)/2$ con p el número de variables n las observaciones. La prueba de bondad de ajuste de u_i a una Beta se ve en la figura 4 que es una salida del software Infostat.

```
Salida
> mardiaTest(bivariado)
Mardia's Multivariate Normality Test
-----
data : bivariado

g1p      : 0.2334561
chi.skew  : 3.890935
p.value.skew : 0.4209674

g2p      : 6.532396
z.kurtosis : -1.834505
p.value.kurt : 0.06657911

chi.small.skew : 4.088626
p.value.small : 0.3941445

Result    : Data are multivariate normal.
```

Figura 1. Prueba de normalidad MardiaTest.

```
> hzTest(bivariado)
Henze-Zirkler's Multivariate Normality Test
-----
data : bivariado

H2      : 0.5488044
p-value : 0.4914625

Result   : Data are multivariate normal.
```

Figura 2. Prueba de normalidad Henze-Zirkler.

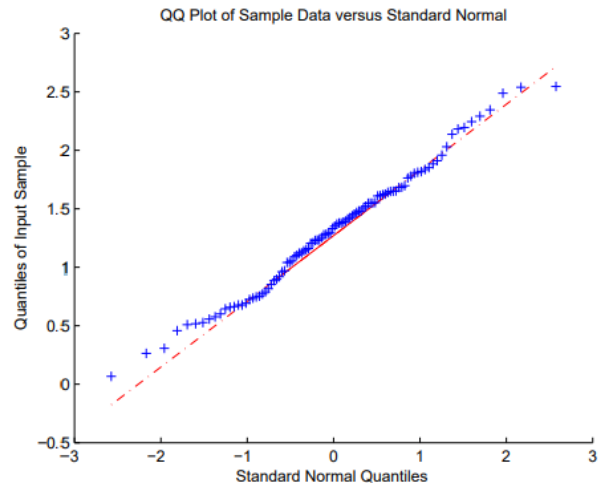


Figura 3. Diagrama Q-Q para los datos simulados.

$$D_1^2 = (y_i - \bar{y})' S^{-1} (y_i - \bar{y}) \quad (25)$$

$$u_i = \frac{n D_1^2}{(n - 1)^2} \quad (26)$$

Prueba de bondad de ajuste (Kolmogorov)

Variable	Ajuste	a	b	n	Estadístico D	p-valor
u	Beta(1,48,5)	1,60	78,21	100	0,11	0,1920

Figura 4. Prueba de bondad para los u_i .

5.3. Aplicación a DEA Bayesiano

Una vez logrado que el generador bivariado tanto discreto como continuo pasen las pruebas de bondad de ajuste, se puede poner en práctica al utilizarlo en un modelo DEA estocástico Bayesiano. Para correr dicho DEA Bayesiano se han tenido en cuenta varios aspectos. Primero se usaron datos de DANE Colombia (ver siguiente subsección “descripción de los datos”). Segundo se transformó un DEA estocástico expresándolo en forma de conjuntos convexos para garantizar que las soluciones de un problema no lineal son óptimos globales. Tercero se corrió el modelo estocástico transformado utilizando la distribución a posteriori de las variables de salida. Cuarto se obtuvieron, gracias a los aspectos anteriores, los determinados intervalos de confianza, las distribuciones de probabilidad de las eficiencias y las comparaciones con DEA clásico. En las siguientes subsecciones se amplía cada uno de los anteriores aspectos.

5.3.1. Aplicación a DEA Bayesiano

Se han usado datos del DANE Colombia, en un estudio sobre la investigación de educación formal en Colombia. Este estudio está enfocado a los niveles de preescolar, básica primaria, básica secundaria y educación media

[20]. En dicho estudio se cuenta con una base de datos llamada indicadores de eficiencia 2004-2014 (es la versión más actual encontrada que está organizada por departamentos) que contiene la información de porcentajes de alumnos aprobados, reprobados, desertores y transferidos de todos los municipios de Colombia.

Para probar el generador aplicado al DEA estocástico Bayesiano, se seleccionaron 4 variables medidas en el año 2014, dos de entrada y dos de salida. Como variables de entradas se definieron el porcentaje de estudiantes reprobados y porcentaje de estudiantes desertores. Como variables de salida el porcentaje de estudiantes aprobados y porcentaje de transferidos. Como DMUs se tuvieron en cuenta los 19 municipios de mayor tamaño de Colombia. Con estas 4 variables se desea crear una medida comparativa de resultados para los municipios. La lógica es que las salidas mientras más altas mejor, es decir mientras más estudiantes aprobados y transferidos mejor. Las entradas mientras menores mejor, así que mejor será que haya menor reprobados y menos desertores para la educación en Colombia. Se tienen en cuenta las siguientes definiciones de acuerdo con el Ministerio de Educación Nacional [21]:

- Reprobados. Son los alumnos que durante el período electivo no logran cumplir con todos los requisitos académicos para ingresar al siguiente período escolar.
- Desertores. Son los estudiantes que abandonan el sistema escolar durante el período académico, o que finalizan el período lectivo, pero no regresan en el período siguiente.
- Aprobados. Son alumnos que durante el período lectivo cumplen con todos los requisitos académicos para ingresar al siguiente período escolar.
- Transferidos. Son los alumnos que se encontraban matriculados en un establecimiento educativo y que cambiaron de institución educativa durante el año escolar, por lo tanto, siguen siendo cubiertos por el sistema educativo.

5.3.2. Problema no lineal convexo

En [22] se puede consultar el problema de optimización como compuesto como sigue:

$$\begin{aligned} & \min c^T x + c^f \\ & \text{s.t } l^c \leq Ax \leq u^c \\ & \quad l^x \leq x \leq u^x \\ & \quad x \in C, \end{aligned} \quad (27)$$

donde C debe satisfacer los siguientes requerimientos. Sea $x^t \in R^{n^t}$, $t = 1, \dots, k$ vectores incluidos en las variables de decisión, entonces se define: $C := \{x \in R^n : x^t \in C_t, t = 1, 2, \dots, k\}$ o donde C_t debe tener una de las siguientes formas:

• Conjunto R :

$$C_t = \{x \in R^{n^t}\}$$

• Cono cuadrático:

$$C_t = \left\{x \in R^{n^t} : Cx_1 \geq \sqrt{\sum_{j=2}^{n^t} x_j^2}\right\}$$

• Cono cuadrático rotado:

$$C_t = \left\{x \in R^{n^t} : 2x_1 x_2 \geq \sum_{j=3}^{n^t} x_j^2, x_2 \geq 0\right\}$$

Basado en los conceptos anteriores, se transformó el modelo (9) en un problema convexo usando un cono cuadrático y restricciones lineales para garantizar de que, a pesar de tener un problema no lineal, se puedan lograr soluciones óptimas globales. Se puede cambiar el primer conjunto de restricciones de la salida creando nuevas variables y obteniendo conos cuadráticos. Entonces el primer conjunto de restricciones de (9) queda así:

$$\phi y_0 \leq \sum_j^N \lambda_j \mu_j - 1,96 \sqrt{\sum_{j=1, j \neq 0}^N \lambda_j^2 \sigma_j^2 + (\lambda_0 - \phi)^2 \sigma_0^2},$$

que se puede expresar como:

$$\frac{1}{1,96}(-\phi y_0 + \sum_j^N \lambda_j \mu_j) \geq \sqrt{\sum_{j=1, j \neq 0}^N \lambda_j^2 \sigma_j^2 + (\lambda_0 - \phi)^2 \sigma_0^2}$$

Reemplazando el lado izquierdo por una variable x_1 como el cono cuadrático tenemos:

$$x_1 \geq \sqrt{\sum_{j=1, j \neq 0}^N \lambda_j^2 \sigma_j^2 + (\lambda_0 - \phi)^2 \sigma_0^2} \quad (28)$$

Si por ejemplo la DMU observada es la 19, entonces se pueden hacer los siguientes reemplazamientos:

$$\begin{aligned} x_2^2 &= \lambda_1^2 \sigma_1^2; x_3^2 = \lambda_2^2 \sigma_2^2; \dots; x_{19}^2 = \lambda_{18}^2 \sigma_{18}^2; x_{20}^2 \\ &= (\lambda_{19} - \phi)^2 \sigma_{19}^2 \end{aligned} \quad (29)$$

Ahora, para la DMU 19, se puede expresar (28) así:

$$x_1 \geq \sqrt{\sum_{j=2}^{20} x_j^2}, \quad (30)$$

que es un cono cuadrático convexo. Las restricciones del modelo se completan con las siguientes restricciones lineales que parten de (29).

$$\begin{aligned} x_2 &= \lambda_1 \sigma_1; x_3 = \lambda_2 \sigma_2; \dots; x_{19} = \lambda_{18} \sigma_{18}; x_{20} \\ &= (\lambda_{19} - \phi) \sigma_{19} \end{aligned} \quad (31)$$

Las variables x_1, x_2, \dots, x_{20} son todas expresiones de tipo lineal. Un procedimiento muy similar se realiza para las restricciones de la segunda salida. De esta forma se demuestra que el problema es de programación no lineal con restricciones convexas.

5.3.3. Uso de la distribución a posteriori

Se programó en el software Matlab el generador bivariado discretizado propuesto en el presente artículo para simular valores que siguieran la distribución a posteriori (15) cuya marginal para μ_Y es de Student multivariada la cual está definida por ecuaciones (32) y (33) de acuerdo con siguiente definición:

Sea Y un vector aleatorio continuo $K \times 1$. Que μ sea un vector $K \times 1$, V una matriz simétrica definida positiva y $n \in \mathbb{R}_{e++}$. Se dice que, Y tiene distribución Student multivariada con media μ , matriz de escala V y n grados de libertad si su función de densidad de probabilidad conjunta es:

$$f_x(x) = c \left(1 + \frac{1}{n} (y - \mu)' V^{-1} (y - \mu) \right)^{-(n+k)} \quad (32)$$

Donde:

$$c = (n\pi)^{-k/2} \frac{\Gamma(n/2 + K/2)}{\Gamma(n/2)} |det(V)|^{-1/2} \quad (33)$$

Para las matrices de varianzas y covarianzas se usó la simulación de Wishart de Matlab. Al obtener las medias y varianzas necesarias que provienen de distribuciones a posteriori, se alimenta el modelo DEA estocástico (9).

Para evaluar la incertidumbre de los parámetros de la distribución a posteriori, se tuvo en cuenta que, como era de esperarse, la parte derecha de la expresión (15) es normal multivariada ya que la marginal de μ_Y es Student multivariada. Se hizo la prueba de multinormalidad de Henze-Zirkler a los datos obtenidos del generador discretizado que sigue la distribución a posteriori en expresión (15) y dicha prueba fue superada de acuerdo con la figura 5.

\$MultivariateNormality										
	Test	HZ	p value	MVN						
1	Henze-Zirkler	0.345192	0.5841939	YES						
\$UnivariateNormality										
	Test	Variable	Statistic	p value Normality						
1	Shapiro-Wilk	V1	0.9629	0.6301 YES						
2	Shapiro-Wilk	V2	0.9640	0.6523 YES						
\$Descriptives										
	n	Mean	Std.Dev	Median	Min	Max	25th	75th	Skew	Kurtosis
V1	19	88.68947	2.797200	88.9	83.1	92.9	87.00	91.30	-0.32627725	-0.9632211
V2	19	4.700000	2.691757	4.9	0.4	9.3	2.65	6.65	0.05317326	-1.3134169

Figura 5. Prueba de normalidad Henze-Zirkler a posteriori.

De acuerdo con la figura 5 se puede obtener que $P((84,09 < \mu_{Y1} < 93,29) | Y) = 0,90$ y $P((0,27 < \mu_{Y2} < 9,13) | Y) = 0,90$

5.3.4. Obtención de eficiencias

En las subsecciones anteriores se logró obtener $p(\theta|Z)$, lo que permite llegar al cálculo de las eficiencias $p(r|Z)$. De esta forma se ha podido llegar al cálculo de las eficiencias de los principales municipios de Colombia teniendo en cuenta las medidas del DANE Colombia. Se usaron rutinas del programa Mosek [22], específicamente el toolbox Mosekopt, para correrse en Matlab. Estas rutinas usadas son diseñadas por Mosek para resolver un problema de optimización cónica a través de de conos convexos. Para cada municipio (DMU) se corrió el modelo estocástico Bayesiano 30 veces. La tabla 3 muestra los puntajes de DEA clásico orientado a las salidas con retorno a escala contante (CRS por sus siglas en inglés) comparado con del DEA Bayesiano estocástico. De este último se presenta también la media, desviación estándar y los intervalos de confianza del 95% construidos con el estadístico t student (usando el teorema del límite central), donde L.I significa límite inferior y L.S el límite superior. También se presentan el ranking alcanzado por DEA clásico y Bayesiano.

Las pruebas de bondad de ajuste para tres distribuciones de probabilidad adecuadas de acuerdo con los histogramas que presentaron las eficiencias de los municipios son presentadas en la tabla 4.

La figura 6 muestra una de las corridas que se usó en la DMU 19 usando software MOSEK. En la figura 7 se muestra una de las corridas hechas, para la DMU 19, se puede apreciar que la solución es óptima debido a que el problema primal y dual llegaron al mismo resultado.

Tabla 3. Resultados empíricos

DMU	Municipio	DEA-O CRS		DEA-O estocástico Bayesiano				
		DEA	Ranking	Media	D.E	L.I	L.S	Ranking
1	Medellín	1.308	8	0.791	0.303	0.678	0.904	12
2	Pasto	1.192	12	0.721	0.190	0.650	0.792	16
3	Bar/lla	1.038	16	0.762	0.206	0.685	0.839	13
4	Bogotá	1.000	18	0.667	0.274	0.564	0.769	18
5	Cartagena	1.289	9	0.833	0.168	0.770	0.896	10
6	Manizales	1.168	13	0.673	0.200	0.599	0.748	17
7	Popayán	1.389	5	0.959	0.260	0.862	1.056	5
8	Valle/par	1.000	19	0.738	0.241	0.648	0.828	15
9	Montería	1.251	10	0.879	0.309	0.763	0.994	8
10	Neiva	1.689	1	1.093	0.282	0.988	1.198	1
11	Riohacha	1.527	2	1.000	0.303	0.887	1.113	4
12	Santa Marta	1.067	15	0.747	0.223	0.664	0.830	14
13	Cúcuta	1.373	6	1.090	0.454	0.921	1.260	2
14	Armenia	1.000	17	0.587	0.235	0.499	0.675	19
15	Pereira	1.396	4	1.023	0.384	0.879	1.166	3
16	Buc/manga	1.224	11	0.828	0.218	0.747	0.909	11
17	Sincelejo	1.480	3	0.914	0.244	0.823	1.005	7
18	Ibagué	1.327	7	0.958	0.384	0.815	1.101	6

Fuente: elaboración propia.

Tabla 4. Pruebas de bondad de ajuste Kolmogorov

Municipio	Gamma			Lognormal			Normal		
	α	β	P-valor	μ	σ	P-valor	μ	σ	P-valor
Medellín	8.059	10.187	0.729	0.791	0.289	0.9126	0.791	0.303	0.323
Pasto	17.152	23.797	0.767	0.721	0.176	0.827	0.7208	0.190	0.497
Bar/lla	15.203	19.956	0.435	0.762	0.200	0.571	0.7618	0.206	0.247
Bogotá	1.213	3.867	0.936	0.665	0.255	0.487	0.667	0.274	0.2709
Cartagena	26.115	313.600	0.669	0.833	0.167	0.779	0.833	0.168	0.540
Manizales	12.709	18.873	0.793	0.674	0.195	0.834	0.673	0.200	0.842
Popayán	15.321	15.983	0.482	0.959	0.251	0.587	0.959	0.260	0.362
Valle/par	10.889	14.750	0.491	0.738	0.230	0.645	0.738	0.241	0.205
Montería	10.289	11.712	0.172	0.876	0.275	0.265	0.879	0.309	0.088
Neiva	16.717	15.294	0.585	1.093	0.275	0.712	1.093	0.282	0.450
Riohacha	13.291	13.289	0.391	0.999	0.278	0.508	1.000	0.303	0.293
Santa Marta	11.358	15.204	0.652	0.749	0.236	0.553	0.747	0.223	0.670
Cúcuta	7.422	6.806	0.664	1.087	0.408	0.746	1.090	0.454	0.454
Armenia	8.723	14.854	0.335	0.585	0.197	0.538	0.587	0.235	0.083
Pereira	8.107	7.927	0.766	1.023	0.375	0.769	1.023	0.384	0.567
Buc/manga	16.254	19.632	0.743	0.828	0.211	0.861	0.828	0.218	0.437
Sincelejo	15.866	17.359	0.864	0.914	0.235	0.926	0.914	0.244	0.701
Ibagué	7.986	8.335	0.428	0.955	0.343	0.646	0.958	0.384	0.154
Cali	12.577	14.844	0.553	0.847	0.246	0.726	0.847	0.254	0.286

Fuente: elaboración propia.

Las pruebas de bondad de ajuste para tres distribuciones de probabilidad adecuadas de acuerdo con los histogramas que presentaron las eficiencias de los municipios son presentadas en la tabla 4.

La figura 6 muestra una de las corridas que se usó en la DMU 19 usando software MOSEK. En la figura 7 se muestra una de las corridas hechas, para la DMU 19, se puede apreciar que la solución es óptima debido a que el problema primal y dual llegaron al mismo resultado.

```
MOSEK Version 7.1.0.60 (Build date: 2016-10-7 14:10:42)
Copyright (c) 1998-2016 MOSEK ApS, Denmark. WWW: http://mosek.com
Platform: Windows/64-X86

Computer
Platform      : Windows/64-X86

Problem
Name          :
Objective sense : min
Type          : CONIC (conic optimization problem)
Constraints    : 42
Cones         : 2
Scalar variables : 60
Matrix variables : 0
Integer variables : 0
```

Figura 6. Salida Mosek.

```
ITE PFEAS DFEAS GFEAS PRSTATUS POBJ DOBJ
0 1.2e+000 2.0e+000 0.0e+000 0.00e+000 -1.0000000000e+000 0.000000
1 2.2e-001 3.7e-001 2.2e-016 9.43e-001 -1.256594295e+000 -9.55204
2 3.2e-002 5.2e-002 4.4e-016 1.02e+000 -1.211905640e+000 -1.18382
3 7.0e-003 1.1e-002 1.3e-015 1.52e+000 -1.186312596e+000 -1.18359
4 5.3e-004 8.7e-004 4.7e-015 1.22e+000 -1.191064599e+000 -1.19081
5 3.0e-005 5.0e-005 1.0e-013 8.67e-001 -1.190656396e+000 -1.19063
6 1.5e-006 2.5e-006 9.5e-015 9.92e-001 -1.190626577e+000 -1.19062
7 4.7e-010 7.7e-010 1.1e-013 1.00e+000 -1.190625001e+000 -1.19062
Interior-point optimizer terminated. Time: 0.09.

Optimizer terminated. Time: 0.14

Interior-point solution summary
Problem status : PRIMAL_AND_DUAL_FEASIBLE
Solution status : OPTIMAL
Primal. obj: -1.1906250013e+000 Viol. con: 1e-008 var: 4e-010
Dual. obj: -1.1906250012e+000 Viol. con: 8e-011 var: 6e-010
```

Figura 7. Solución óptima usando software Mosek.

6. Conclusiones

El propósito de este artículo fue desarrollar un generador bivariado basado en gradientes y derivadas direccionales. Este es aplicado a variables continuas cuando es posible lograr expresar una variable en forma explícita en términos de la otra variable y de variables aleatorias uniformes. Conociendo las dificultades de la expresión explícita para algunas funciones, se logró adaptar el generador propuesto para simular variables continuas con una estrategia de discretización. Su utilidad se ha puesto a prueba en un problema DEA estocástico Bayesiano que permite establecer intervalos de confianza en los puntajes de eficiencia. Mediante conversiones a conos convexos y restricciones lineales, se logró

demostrar que el problema DEA tratado presentó soluciones óptimas. La metodología presentada permitirá trabajar de manera más sencilla las técnicas de solución que, a través de estimación Bayesiana, están surgiendo en la literatura DEA.

Referencias

- [1] W. Cook, L. Seiford, "Data envelopment analysis (dea) - thirty years on", *European Journal of Operational Research*, vol. 192, no. 1, pp. 1-17, 2009, doi: 10.1016/j.ejor.2008.01.032
- [2] M. Bruni, D. Conforti, P. Beraldi, E. Tundis, "Probabilistically constrained models for efficiency and dominance in dea", *International Journal Production Economics*, vol. 117, no. 1, pp. 219-228, 2009, doi: 10.1016/j.ijpe.2008.10.011
- [3] L. Simar, P. Wilson, "Sensitivity analysis of efficiency scores: How to bootstrap in nonparametric frontier models", *Management Science.*, vol. 44, no. 1, pp. 49-61, 1998, doi: 10.1287/mnsc.44.1.49
- [4] L. Simar, P. Wilson, "Statistical inference in non-parametric frontier models: the state of the art", *Journal of Productivity Analysis*, vol. 13, pp. 49-78, 2000, doi: 10.1023/A:1007864806704
- [5] R. Dyson, E. Shale, "Data envelopment analysis, operational research and uncertainty", *Journal of Operational Research Society*, vol. 61, pp. 25-34, 2010, doi: 10.1057/jors.2009.145
- [6] M. Ceyhan, J. Banneyan, "Handling estimated proportions in public sector data envelopment analysis", *Annals of Operational Research*, vol. 221, pp. 107-132, 2014, doi: 10.1007/s10479-011-1007-z
- [7] E. Tsionas, E. Papadakis, "A bayesian approach to statistical inference in stochastic dea", *Omega*, vol. 38, no. 5, pp. 309-314, 2010, doi: 10.1016/j.omega.2009.02.003
- [8] P. Mitropoulos, M. Talias, I. Mitropoulos, "Combining stochastic dea with bayesian analysis to obtain statistical properties of the efficiency scores: An application to greek public", *Mathematical and Computer Modelling hospitals*, vol. 243, no. 1, pp. 302-311, 2015, doi: 10.1016/j.ejor.2014.11.012
- [9] W. Cooper, Z. Huang, S. Li, "Change constrained DEA", en *Handbook on data envelopment analysis*, 2nd ed. Boston: Springer US, 2011, pp. 211-240.

- [10] D. Friesner, R. Mittelhammer, R. Rosenman, "Inferring the incidence of industry inefficiency from dea estimates", *European Journal of Operational Research*, vol. 224, no. 2, pp. 414-424, 2013, doi: 10.1016/j.ejor.2012.08.003
- [11] A. Charnes, W. Cooper, E. Rhodes, "Measuring the efficiency of decision making units", *European Journal of Operation Research*, vol. 2, no. 6, pp. 429-444, 1978.
- [12] K. Land, C. Lovell, S. Thore, "Productive efficiency under capitalism and state socialism: An empirical inquiry using chance-constrained data envelopment analysis", *Technological Forecasting and Social Change*, vol. 46, no. 2, pp. 139-152, 1994, doi: 10.1016/0040-1625(94)90022-1
- [13] K. Land, C. Lovell, S. Thore, "Chance-constrained data envelopment analysis", *Managerial and Decision Economics*, vol. 14, pp. 541-554, 1993, doi: 10.1002/mde.4090140607
- [14] K. Land, C. Lovell, S. Thore, "Productive efficiency under capitalism and state socialism: the chance constrained programming approach", *Supplement to Public Finance*, vol. 47, pp. 109-121, 1992.
- [15] K. Land, C. Lovell, S. Thore, "Chance constrained data envelopment analysis", *Managerial and Decision Economics*, vol. 14, no. 6, pp. 541-554, 1993, doi: 10.1002/mde.4090140607
- [16] W. Cooper, L. Seiford, K. Tone, *Data envelopment analysis: a comprehensive text with models, applications, references and DEA-solver software, Second edition*. New York, NY, USA: Springer, 2007.
- [17] C. Subhash, *Data envelopment analysis: theory and techniques for economics and operations research*. New York, NY, USA: Cambridge University Press, 2004.
- [18] J. Vargas, *Simulación Basada en Probabilidad*. Manizales: Universidad Autónoma de Manizales, 2013.
- [19] S. Korkmaz, D. Goksuluk, G. Zararsiz, "MVN: An r package for assessing multivariate normality", *The R Journal*, vol. 6, no. 2, pp. 151-162, 2014.
- [20] DANE, "Educación formal educ," [En línea]. Disponible en: <http://www.dane.gov.co/index.php/estadisticas-por-tema/educacion/poblacion-escolarizada/educacion-formal>
- [21] R. d. C. Ministerio de Educación Nacional, "Educación de calidad," [En línea]. Disponible en: <http://www.mineducacion.gov.co/1621/article-123912.html>
- [22] A. Mosek, "The mosek optimization toolbox for matlab manual", Mosek ApS, Denmark, Tech. Rep. version 7.1, revision 6.0.