

FAC. CIENCIAS

49
2ej



METODOS NO PARAMETRICOS PARA OBTENER
LIMITES DE CONFIANZA DE TASAS DE
CRECIMIENTO POBLACIONAL ESTIMADAS CON
MODELOS MATRICIALES DE LESLIE

BIOLOGIA

ALVARO CHAOS CADOR

TESIS CON
FALLA DE ORIGEN

1993



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

RESUMEN.....	III
INTRODUCCIÓN.....	1
MODELOS PARA ANÁLISIS POBLACIONAL.....	4
LÍMITES DE CONFIANZA PARA λ	13
OBJETIVOS.....	20
MÉTODOS.....	21
RESULTADOS Y DISCUSIÓN.....	29
CONCLUSIONES.....	60
APÉNDICE A.....	62
APÉNDICE B.....	76
REFERENCIAS.....	121

RESUMEN

A pesar de que los estimados de la tasa de crecimiento poblacional sustentan las conclusiones de muchos trabajos biológicos, muy pocos estudios evalúan la precisión estadística de dichos estimados. Esto se debe a que estimar el error típico de la tasa de crecimiento poblacional es difícil porque esta tasa es una función no lineal de las tasas vitales, lo que origina una relación complicada entre la distribución de los errores de los estimadores de las tasas vitales y la distribución muestral de la tasa de crecimiento poblacional. Existen tres métodos para evaluar la precisión de estimados de esta tasa: 1) utilizando una aproximación analítica de la varianza de la tasa de crecimiento poblacional, 2) mediante simulación de Monte Carlo para obtener la distribución muestral de la tasa de crecimiento poblacional y 3) los métodos de remuestreo conocidos en inglés como "bootstrap" y "jackknife". Los dos primeros hacen suposiciones sobre el tamaño de los errores o sobre su distribución, los últimos no. Se han usado simulaciones de Monte Carlo para probar la estimación analítica de tasas de crecimiento poblacional finitas (λ) obtenidas a partir de matrices de Leslie y de Lefkovich y de la ecuación de Euler con datos de cohortes. Los métodos no paramétricos se han probado para tasas de crecimiento poblacional instantáneas (r) obtenidas de la ecuación de Euler con datos de cohortes de poblaciones con diferentes patrones demográficos, pero con un único tamaño de muestra.

Hasta ahora no se ha explorado el comportamiento estadístico de λ obtenida para poblaciones con generaciones sobrepuestas y estructuradas por edades o tamaños con base en los métodos de remuestreo. Tampoco se ha estudiado el efecto del tamaño de muestra, de los patrones demográficos y del número de entradas de la matriz sobre la precisión de los estimados de λ . Este trabajo estudia el comportamiento de tasas poblacionales finitas, obtenidas de matrices de Leslie (4×4) para poblaciones sobrepuestas con diferentes patrones demográficos. También analiza el efecto del tamaño de muestra sobre el estimador de λ y evalúa el desempeño de los métodos de remuestreo.

A partir de la combinación de 6 curvas de sobrevivencia con 4 de fecundidad, un programa de computadora creó 24 poblaciones diferentes de 1000 individuos cada una. Las curvas se obtuvieron de ajustar a 20 unidades de tiempo los diferentes tipos de curvas, tanto de sobrevivencia como de fecundidad, encontrados en una revisión de 33 estudios. De estas poblaciones se muestreó un total de 1000 veces para diferentes tamaños de muestra (10, 20, 40, 80, 200). De cada muestreo se calculó una λ a partir de la matriz de transiciones. La media de éstas se denominó λ real. Se obtuvo la distribución muestral de λ para cada tamaño de muestra y se comprobó su normalidad. Se aplicaron los métodos de remuestreo para analizar su desempeño. Los estimados de λ de ambos métodos se compararon con la λ real al 95% de confianza. Para comprobar si el método de aproximación de las entradas de la matriz afecta el comportamiento estadístico de λ , se construyeron las matrices de 3

comportamiento estadístico de λ , se construyeron las matrices de 3 poblaciones utilizando las fórmulas de reproducción discontinua. En la mayoría de los tamaños de muestra de todas las poblaciones las λ estimadas con los métodos de remuestreo fueron significativamente distintas de la real; sin embargo, a medida que aumentó el tamaño de muestra el valor de los estimados se acercó a la λ real. La distribución de λ de todas las poblaciones fue significativamente diferente de una distribución normal, a medida que aumentaba la mortalidad juvenil las colas se desplazaron de derecha a izquierda y llegaron a ser bimodales para curvas de tipo II y III. Para las poblaciones con valores de λ pequeños la bimodalidad fue marcada, en particular con la curva de fecundidad 1. El método de aproximación influyó sobre la distribución muestral de λ y sobre su media también. La sobrevivencia afectó la distribución muestral de λ , la fecundidad afectó la media.

Queda por estudiar el efecto en λ del número de categorías de la matriz de transiciones, como también su comportamiento para diferentes patrones demográficos y tamaños de muestra de poblaciones estructuradas por tamaños.

INTRODUCCIÓN

El ciclo de vida de los seres vivos consta de una serie de estadios o fases por los que pasa un organismo desde su nacimiento hasta su muerte. El estudio de los ciclos de vida provee información valiosa en investigaciones de Ecología, Genética y Evolución. Las tasas vitales, como la tasa de crecimiento, la tasa de fecundidad y la tasa de sobrevivencia describen el desarrollo de los organismos a lo largo de sus ciclos de vida y constituyen la base de cualquier estudio demográfico. La manera en que éstas responden al ambiente determina la dinámica poblacional y la evolución de las historias de vida de las especies (Caswell, 1989).

De los atributos poblacionales, la tasa de crecimiento, que representa el número de individuos que cada individuo deja de una generación a otra (para tiempos discretos se simboliza con la letra griega lambda (λ)), es particularmente importante. La tasa de crecimiento poblacional engloba los efectos de factores ecológicos naturales o de perturbaciones artificiales en la capacidad de la población para crecer y expandir su rango de distribución (Begon et al., 1988). Es útil también para diseñar estrategias de manejo y conservación para especies en peligro de extinción y para evaluar el impacto ambiental, ya sea de condiciones naturales o artificiales, en especies indicadoras (Crouse et al., 1987; Caswell, 1989). Además Fisher (1930) demostró que la tasa de crecimiento poblacional estima la adecuación promedio de una población, y por lo tanto,

este parámetro también refleja el potencial evolutivo de cambios en las características del ciclo de vida de las especies.

La comparación de las tasas de crecimiento poblacional con valores específicos, entre especies, poblaciones, censos o condiciones artificiales, sustentan las conclusiones de muchos trabajos biológicos sobre demografía (Lande, 1988; Moloney, 1988), conservación (Crouse et al., 1987; Kinnaird y O'Brian, 1991), dinámica poblacional (Levin y Huggett, 1990; Lasker, 1991), control biológico (Birch, 1953a y 1953b) y evolución de historias de vida (Sarukhán y Gadgil, 1974). A pesar de la importancia que tiene evaluar la precisión estadística de estimados de tasas de crecimiento poblacional, muy pocos estudios proporcionan límites de confianza de dichos estimados (Rago y Dorazio, 1984; Meyer et al. 1989 y Álvarez-Buylla y Slatkin, 1991). La principal razón de la escasez de estimaciones de límites de confianza de λ , se debe a que la estimación del error típico de ésta es problemático. El hecho de que λ sea una función no lineal de los parámetros demográficos de fecundidad, sobrevivencia y crecimiento, acarrea una relación complicada entre la distribución de los errores de los estimadores de las tasas vitales y la distribución muestral de la tasa de crecimiento de la población.

Este trabajo es una contribución al estudio del comportamiento estadístico de estimados de tasas de crecimiento poblacional. En particular, se centra en el análisis de tasas de crecimiento finitas (λ) obtenidas para poblaciones con generaciones sobrepues-

¹Error estándar.

tas clasificadas por edades. Mediante simulaciones en computadora de la dinámica de poblaciones ideales, se analiza el desempeño de los métodos no paramétricos, conocidos en inglés como "bootstrap" y "jackknife"², para estimar límites de confianza de λ . Además se explora el efecto del tamaño de muestra utilizado al estimar las tasas vitales, en el comportamiento estadístico de λ para poblaciones con patrones demográficos contrastantes. Se discuten las implicaciones prácticas de los resultados con base en una revisión de 33 estudios empíricos.

² No se encontró la traducción de estos términos.

MODELOS PARA ANÁLISIS POBLACIONAL

Si suponemos que todos los individuos de una población son iguales, podemos calcular el número de individuos para el intervalo de tiempo siguiente con la ecuación de crecimiento exponencial:

$$n_{(t+1)} = \lambda n_{(t)} \quad (1)$$

y para cualquier tiempo:

$$n_{(t)} = \lambda^t n_0 \quad (2)$$

donde $\lambda = e^r$ y r es la tasa de crecimiento instantáneo de la población. Si la población está en equilibrio r es igual a cero, si r es negativa la población decrecerá y si es positiva la población aumentará. Este modelo no toma en cuenta la existencia de recursos finitos. Hay otros modelos que sí lo hacen incorporando un tope o capacidad de carga K , y son útiles para describir comportamientos poblacionales con densodependencia. No obstante, estos modelos también toman a la población como un grupo homogéneo, sin hacer distinciones de edad, sexo, tamaño, etc. Los modelos matriciales describen la dinámica de poblaciones con estructuras de edades y/o tamaños.

Desde que Leslie (1945) introdujo los modelos matriciales de

proyección poblacional, se utilizan frecuentemente en proyectar el estado futuro de una población. El modelo matricial básico y más simple de crecimiento poblacional es:

$$n_{(t+1)} = A n_{(t)} \quad (3)$$

Donde n es un vector de n renglones cuyos elementos son el número de individuos en cada categoría al tiempo t y A es una matriz cuadrada y no negativa de tamaño $n \times n$. Los elementos a_{ij} están determinados por la sobrevivencia, la fecundidad y el crecimiento de los individuos durante el intervalo de tiempo escogido y representan la contribución de los individuos de la categoría j a la categoría i del tiempo t al tiempo $t+1$.

De esta forma, si tenemos una población inicial de composición n , podemos saber la composición de ésta en cualquier tiempo posterior. Las propiedades de los modelos matriciales originan que para una t grande la proporción de individuos en las diferentes categorías se vuelva constante y que la población crezca a una tasa exponencial: λ , que es el mayor autovalor positivo de A , solución de la ecuación:

$$\det (A - \lambda I) = 0 \quad (4)$$

Como $\lambda = e^r$ entonces si $\lambda = 1$ ($r=0$) la población se mantiene en equilibrio, si es mayor que uno ($r > 0$) aumentará y si es menor que

uno ($r < 0$) disminuirá. λ se puede calcular también a partir de las tasas de fecundidad y sobrevivencia tomadas de una tabla de vida con la ecuación de Euler:

$$1 = \sum_{x=0}^{\Omega} \lambda^{-x} l_x m_x \quad (5)$$

En donde λ es la tasa de crecimiento poblacional, x es la categoría de edad, Ω es la última categoría, l_x es la probabilidad de sobrevivir hasta la edad x y m_x es la fecundidad asociada a la edad x .

La tabla I presenta la interpretación demográfica de algunas propiedades algebraicas de las matrices de proyección poblacional.

Para construir estas matrices es necesario tener claras las divisiones del ciclo de vida del organismo. Los modelos de Leslie se basan en intervalos de edad, todos de igual duración entre sí e iguales al intervalo de iteración de la matriz. El ciclo de vida del individuo se divide en diferentes categorías y cada una tiene asociada una fecundidad y una probabilidad de pasar a la siguiente categoría. La fecundidad de cada categoría se coloca en el primer renglón de la matriz de acuerdo a fórmulas de aproximación según la fenología y censos poblacionales (Caswell, 1989). Las probabilidades de transición se acomodan en la subdiagonal; lo demás se llena con ceros.

Sin embargo, la edad no es siempre un buen indicador del destino (propiedades demográficas) de un individuo. En muchos casos

Tabla I Interpretación demográfica de algunas propiedades algebraicas de las matrices de proyección poblacional.

Símbolo	Interpretación demográfica	Definición
A	Matriz de proyección poblacional	Una matriz cuadrada con coeficientes que representan tasas de lx , mx , crecimiento, etc.
a_{ij}	Fecundidades y probabilidades de transición.	Entrada en el renglón i y en la columna j .
λ	Tasa de crecimiento poblacional.	El mayor de los autovalores positivos de la matriz A.
w	Distribución estable de edades.	El autovector derecho de A.
v	Valor reproductivo.	El autovector izquierdo de A.
S	Matriz de sensibilidades.	Matriz cuadrada cuyos elementos son las s_{ij} .
s_{ij}	Elemento de la matriz S.	Sensibilidad de λ a cambios en el elemento a_{ij} .
E	Matriz de elasticidades.	Matriz cuadrada cuyos elementos son las e_{ij} .
e_{ij}	Elemento de la matriz E.	Sensibilidad proporcional de λ a cambios en el elemento a_{ij} .

el tamaño, la fase o el estadio nos indica mejor su fecundidad y sobrevivencia, especialmente en plantas. Para este tipo de especies, Lefkovitch (1965) propuso un modelo general en el que se clasifican o dividen las partes del ciclo vital en estadios, tamaños o fases de duración arbitraria. En realidad, el modelo de Leslie es un caso particular del modelo de Lefkovitch. Un par de problemas importantes que surgen al aplicar este modelo son:

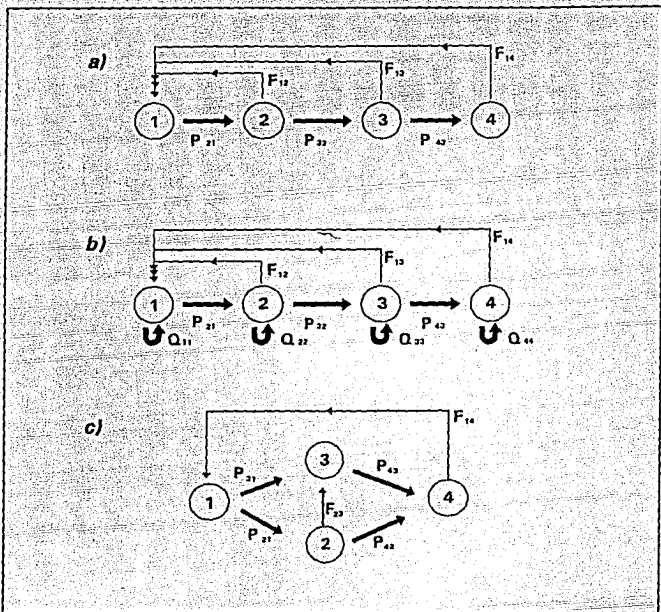


Figura 1 ciclo de vida clasificado por edad (a), por fase (b) y por fase con reproducción vegetativa. Las flechas indican las transiciones entre las categorías, ya sea por fecundidad (F) o sobrevivencia (P y Q).

definir sin ambigüedad las categorías en las que se dividirá el ciclo de vida y calcular el número de categorías que se usará para tratar de minimizar la suma del error de muestreo y el error de distribución (Moloney, 1986). La figura 1a muestra el ciclo de vida de un organismo clasificado por edad. Consta de 4 categorías de

edades representadas por las circunferencias. Las flechas indican las transiciones entre las categorías, ya sea por fecundidad o por sobrevivencia. Se toman estas fecundidades (F) y probabilidades de transición (P) y se construye la matriz de Leslie:

$$A = \begin{pmatrix} 0 & F_{12} & F_{13} & F_{14} \\ P_{21} & 0 & 0 & 0 \\ 0 & P_{32} & 0 & 0 \\ 0 & 0 & P_{43} & 0 \end{pmatrix}$$

Para los organismos clasificados por estadios las cosas cambian; ahora existe la posibilidad de que un individuo permanezca en la misma categoría del tiempo t a al tiempo t+1 (Q), es decir, que se quede (por eso se pone una Q). Al pasar el ciclo de vida de la figura 1b a una matriz de Lefkovich se obtiene:

$$B = \begin{pmatrix} Q_{11} & F_{12} & F_{13} & F_{14} \\ P_{21} & Q_{22} & 0 & 0 \\ 0 & P_{32} & Q_{33} & 0 \\ 0 & 0 & P_{43} & Q_{44} \end{pmatrix}$$

El modelo se puede complicar un poco más al tomar un ejemplo con reproducción vegetativa, como lo ilustra la figura 1c. La matriz de Lefkovich es:

$$C = \begin{bmatrix} 0 & 0 & 0 & F_{14} \\ P_{21} & 0 & F_{23} & 0 \\ P_{31} & 0 & 0 & 0 \\ 0 & P_{42} & P_{43} & 0 \end{bmatrix}$$

Como ya se dijo, al iterar la ecuación 3 la dinámica poblacional converge a una población con una estructura estable, la que crece o decrece a una tasa constante λ . Ésta y otros parámetros poblacionales coinciden con algunas propiedades algebraicas de los modelos matriciales de proyección poblacional.

Cálculo de las P_i y F_i

Las entradas de la matriz de proyección se calculan a partir de la sobrevivencia y fecundidad tomadas de una tabla de vida. La manera en que éstas se obtienen varía de acuerdo al tipo de reproducción de la población: continua o discontinua. Las poblaciones con reproducción continua son aquellas en las que los nacimientos tienen lugar a lo largo de todo el intervalo de tiempo, por otro lado, si los nacimientos se concentran en un período corto de tiempo (una estación del año, por ejemplo) se trata de una población de reproducción discontinua o a pulsos. Las poblaciones humanas representan un buen ejemplo de la primera y muchas poblaciones de insectos y plantas presentan el segundo tipo de reproducción.

Para obtener las P_i y las F_i de una población con reproducción

continua aplicamos las siguientes fórmulas (Caswell, 1989):

$$P_j = \frac{l_{(j)} + l_{(j+1)}}{l_{(j-1)} + l_{(j)}} \quad (6)$$

$$F_j = l_{(0.5)} \left(\frac{m_j + P_j m_{j+1}}{2} \right) \quad (7)$$

Si $l_{(0.5)}$ se desconoce es posible calcularla utilizando interpolación lineal (Keyfitz en Caswell, 1989):

$$l_{(0.5)} = \frac{l_{(0)} + l_{(1)}}{2} \quad (8)$$

La mayoría de los organismos tienen una mortalidad neonatal alta, para este caso una aproximación logarítmica es más precisa:

$$l_{(0.5)} = [l_{(0)} l_{(1)}]^{1/2} \quad (9)$$

La distribución de edades para poblaciones con reproducción discontinua depende del momento en que se realice el censo, ya sea después del periodo de nacimientos (censo postreproductivo) o antes del mismo (censo prerreproductivo). Definamos p como el intervalo de tiempo entre la reproducción y el censo. En gran cantidad de

estudios los censos se realizan justo antes o justo después de la reproducción, lo que corresponde a los límites de p tendiendo hacia 1 o hacia 0 respectivamente. Las fórmulas generales son (Caswell, 1989):

$$P_i = P \text{ (sobrevivencia de la edad } i-1 \text{ a } i+p) \quad (10)$$

$$P_i = \frac{l(i+p)}{l(i-1+p)} \quad (11)$$

$$F_i = l(p) P_i^{i-p} m_i \quad (12)$$

Substituyendo para un censo postreproductivo ($p=0$):

$$P_i = \frac{l(i)}{l(i-1)} \quad (13)$$

$$F_i = P_i m_i \quad (14)$$

y para un censo prerreproductivo ($p=1$):

$$P_i = \frac{l(i+1)}{l(i)+1} \quad (15)$$

LÍMITES DE CONFIANZA PARA λ

Existen tres tipos de métodos para obtener los límites de confianza de tasas de crecimiento poblacional. El primero, es un método analítico que consiste en obtener la varianza de λ a partir de un polinomio que aproxima a la λ como función de los estimados de las tasas vitales. El segundo método consiste en obtener la distribución muestral de λ mediante el método de Monte Carlo en la computadora. Y el tercer método obtiene la distribución muestral de λ mediante alguno de los métodos no paramétricos de remuestreo de bootstrap y jackknife.

Aproximación Analítica

La tasa de crecimiento poblacional, como función de las tasas vitales, se aproxima mediante el método de expansión de Taylor hasta los términos de segundo orden. La esperanza y varianza de este polinomio es una aproximación analítica de la esperanza y varianza de λ como función de los errores de estimación de las tasas vitales (ver Álvarez-Buylla y Slatkin 1991). Suponiendo que los errores de estimación de las tasas vitales son pequeños y que la distribución muestral de λ es aproximadamente normal, se pueden obtener límites de confianza de λ al 95% sumando y restando a la λ promedio aproximadamente dos veces el error típico de λ (el área situada debajo de la curva normal que está entre $Z=1.96$ y $Z=-1.96$ es el 95% del área total, Freund y Smith, 1989). Éste se obtiene

como la raíz cuadrada de la varianza de λ . La derivación analítica se explica en Álvarez-Buylla y Slatkin (1991). Es importante resaltar que la correlación en los errores de los estimadores de las tasas vitales afectan la varianza de λ . Esto ocurre si éstas se obtienen a partir de los mismos individuos o de individuos emparentados genéticamente (Álvarez-Buylla y Slatkin, 1991).

El método analítico es muy sencillo de usar, pero puede proporcionar estimaciones poco confiables si los errores de estimación de las tasas vitales no son pequeños y si la distribución muestral de λ no es normal (Álvarez-Buylla & Slatkin, en prensa). Para detectar y evitar estos problemas se recurre a métodos computacionales de Monte Carlo o a métodos de remuestreo no paramétricos.

Método de Monte Carlo

Este método se puede aplicar a cualquier situación, pero es necesario hacer suposiciones específicas acerca de la distribución de los errores. Las simulaciones de Monte Carlo parten de la suposición que se conoce la distribución conjunta de las tasas demográficas. De esta distribución se muestrea un conjunto de tasas demográficas y se calcula λ . Este proceso se repite un número elevado de veces (generalmente 1000) y se infiere la media, la varianza y la distribución muestral de λ .

Utilizando el método de Monte Carlo con poblaciones de *Daphnia pulex*, Rago y Dorazio (1984) encontraron para tasas de crecimiento

poblacional estimadas con la ecuación de Euler, que cuando el tamaño de la cohorte es chico, se produce un sesgo en la distribución de λ , que el tipo de sobrevivencia y el tamaño de la cohorte afectan la forma de la distribución muestral de λ y que la fecundidad sólo afecta el valor medio de λ .

Houllier et al. (1986) demostraron que si el tiempo de generación es largo, la tasa de crecimiento poblacional es más sensible a errores cometidos al estimar la sobrevivencia en etapas adultas que a errores al estimar la fecundidad. Aunque si las varianzas de las fecundidades son desproporcionadamente grandes, pueden contribuir más que los otros parámetros en la distribución muestral de λ . También, para tres de las cuatro matrices que usaron, encontraron que no existe un sesgo significativo en el estimador de λ (para la cuarta no existe sesgo). En cuanto a la distribución se puede decir que casi es normal, con un ligero sesgo hacia la izquierda. La varianza de λ se subestimó para 3 de las 4 matrices. A diferencia de Rago y Dorazio (1984), λ se calculó a partir de matrices de Leslie y Lefkovitch.

Álvarez-Buylla y Slatkin (en prep.) encontraron que los sesgos reportados por Houllier y colaboradores se debían a errores en los métodos de truncamiento de las distribuciones generadas para las tasas vitales, más que a sesgos reales. Demostraron que, tanto para matrices de Leslie, como para matrices de Lefkovitch, el método analítico proporciona estimaciones de la varianza de λ no sesgadas y estimados de límites de confianza de λ muy parecidos a los obtenidos con simulaciones de Monte Carlo, siempre y cuando los

coeficientes de variación de los estimados de las tasas demográficas sean menores al 50%. Aun cuando éstos son mayores al 50%, los límites de confianza de λ obtenidos mediante el método analítico son muy parecidos a los obtenidos por el método de Monte Carlo.

Métodos No Paramétricos

Al utilizar el modelo analítico o el método de Monte Carlo se debe asumir que λ se distribuye normalmente o que el tamaño de los errores de los estimadores de los parámetros demográficos es chico. La alternativa es recurrir a los métodos de remuestreo de jackknife y de bootstrap que no hacen ninguna de estas suposiciones (Meyer et al. 1989). También gracias a estos métodos otras propiedades de la muestra se pueden explorar numéricamente, a pesar de que su análisis exacto sea imposible (Diaconis y Efron, 1983).

Ambos métodos se basan en la idea de usar a la muestra como un universo surtido³; muestreando varias veces de este universo se obtienen medias, varianzas, sesgos e intervalos de confianza para el parámetro deseado, en este caso λ .

El jackknife calcula una λ a partir de todos los individuos de la muestra. Se le llama λ^1 . Después excluye al primer individuo de la muestra y calcula otra vez λ . Esta es la primera λ^i (donde $i=1$). Continúa calculando las demás λ^i omitiendo sucesivamente un individuo de la población. De esta forma habrá tantas λ^i como individuos en la muestra. Cuando todos los individuos han sido

³ Traducción de "sorted universe".

excluidos, o lo que es igual, cuando el número de λ^i sea igual al tamaño de la muestra ($i=n$), se obtiene λ^j , que no es más que el promedio de estas λ^i . Las fórmulas para obtener el sesgo, la λ jackknife y la varianza son, de acuerdo con Caswell (1989), las siguientes:

$$\text{Sesgo} = (n-1) (\lambda^P - \lambda^j) \quad (17)$$

$$\lambda^j = \lambda^T - \text{sesgo} \quad (18)$$

$$V(\lambda^j) = \frac{n-1}{n} \sum_i (\lambda^i - \lambda^P)^2 \quad (19)$$

El bootstrap obtiene la λ^T de la misma forma que el jackknife. Después remuestrea al azar y con reemplazo y produce una muestra del mismo tamaño que la original, por lo que cada individuo tiene una probabilidad $1/n$ de ser elegido durante cada evento muestral. Como el remuestreo se hace con reemplazo, es posible que en una remuestra cualquiera haya individuos representados más de una vez y que haya otros que no están representados. Este procedimiento de remuestreo se repite b veces; b es el tamaño de muestra del bootstrap. Así produciremos b λ^i . El promedio de éstas se llamará λ^P . Los estimadores del sesgo, la λ bootstrap y la varianza son según Caswell (1989):

$$\text{Sesgo} = (\lambda^P - \lambda^T) \quad (20)$$

$$\lambda^B = \lambda^T - \text{sesgo} \quad (21)$$

$$V(\lambda^B) = \frac{1}{b-1} \sum_i^b (\lambda^i - \lambda^B)^2 \quad (22)$$

El tamaño de muestra del bootstrap afecta la precisión de los estimadores, 1000 es un tamaño suficientemente bueno en la mayoría de los casos (Meyer et al., 1986). El estimador de λ converge después de unos pocos cientos de bootstrap, sin embargo, para obtener estimadores más confiables o reales de la desviación típica y los intervalos de confianza es necesario incrementar el tamaño de muestra del bootstrap entre 500 y 1000 (Caswell, 1989).

Para el jackknife la i toma valores desde 1 hasta n (tamaño de la muestra) y para el bootstrap la i va de 1 a b (tamaño de muestra del bootstrap).

Después de comparar la precisión del jackknife con el bootstrap, Meyer et al. (1986) encontraron que, para estimadores de r , los estimados del jackknife son casi tan precisos y confiables como los del bootstrap y que los errores al estimar la desviación típica de r también son casi iguales. Como el jackknife consume sólo el 1% del tiempo que tarda el bootstrap en hacer los cálculos, recomiendan, para poblaciones con fecundidad y sobrevivencia parecidas a las que se reportan en dicho estudio, usar el jackknife

cuando los recursos computacionales sean limitados o simplemente por economía y rapidez.

En resumen, se han usado simulaciones de Monte Carlo para probar la estimación analítica para lambdas obtenidas a partir de matrices de Leslie y Lefkovitch y a partir de la ecuación de Euler con datos de cohortes. En el último caso se probó el efecto del tamaño de muestra. Los métodos no paramétricos de bootstrap y jackknife se han probado para tasas instantáneas (r) obtenidas de la ecuación de Euler a partir de datos de cohortes, pero usando un único tamaño de muestra. Hasta ahora no se ha explorado el comportamiento estadístico de λ obtenida para poblaciones con generaciones superpuestas y estructuradas por edades (matrices de Leslie) o tamaños (matrices de Lefkovitch) con base en los métodos no paramétricos de bootstrap y jackknife. Para estos casos tampoco se ha estudiado el efecto de los tamaños de muestra y de los patrones demográficos de supervivencia, crecimiento y fecundidad de los individuos.

OBJETIVOS

Los objetivos de esta tesis son:

- 1) Simular en la computadora poblaciones ideales clasificadas por edades y con diferentes curvas de sobrevivencia y fecundidad, cuya dinámica se describe con base en matrices de Leslie.
- 2) Probar para estas poblaciones el desempeño de los métodos no paramétricos de bootstrap y jackknife para obtener límites de confianza de λ .
- 3) Estudiar el efecto del tamaño muestral utilizado para estimar las entradas de la matriz en el desempeño del bootstrap y jackknife para estimar los límites de confianza de λ .
- 4) Discutir las implicaciones de los resultados de las simulaciones en el diseño de estudios empíricos.

MÉTODOS

Se diseñó un programa en computadora con el fin de que creara poblaciones a partir de diferentes combinaciones de tasas vitales. Los detalles del programa, el diagrama de flujo y el listado aparecen en los apéndices A y B.

Las funciones de lx y mx

Para simular los patrones de sobrevivencia y fecundidad de poblaciones naturales se realizó una revisión bibliográfica cuyo resultado se muestra en la tabla II. También se proporciona el tamaño de muestra.

Tabla II

Fuente	Especie	Tam. de muestra	lx	mx
Protozoarios				
Kent, 1981	<i>Tokophrya lemnae</i>	15-24	1-2	2-4
Cnidarios				
Grigg, 1977	<i>Muricea fruticosa</i>	cr	3	1
	<i>Muricea californica</i>	cr	3	1
Rotíferos				
Robertson y Salt, 1981	<i>Asplanchna girodi</i>	40	1	4
Arácnidos				
Polis y Farley, 1980	<i>Paruroctonus mesaensis</i>	145-169	1-2	2
Crustáceos				
Frank, 1960	<i>Daphnia pulex</i>	8-32	1	4
Goulden et al, 1982	<i>Daphnia magna</i>	34-44	1	2-4
	<i>Daphnia galeata mendotae</i>	36-46	1-2	1-4
	<i>Bosmina longirostris</i>	17-58	1-2	2

cr=captura-recaptura

Tabla II Continuación

Fuente	Especie	Tam. de muestra	lx	mx
Crustáceos				
Neill, 1975	<i>Alonella globulosa</i>	45	1	1
	<i>Ceriodaphnia quadrangula</i>	45	1	1
	<i>Simocephalus vetulus</i>	45	2	1
	<i>Hyalella azteca</i>	45	1	3
	<i>Pseudosida bidentata</i>	45	1	1
	<i>Daphnia magna</i>	45	2	4
	<i>Diaptomus conipedalus</i>	45	2	4
Skadsheim, 1990	<i>Gammarus salinus</i>	91	3-2	1
Vanni, 1986	<i>Diaphanosoma birgei</i>	30-35	1-2	4
	<i>Ceriodaphnia lacustris</i>	30-35	1	4
Insectos				
Culliney y Pimentel, 1986	<i>Myzus persicae</i>	27-28	1	4
	<i>Aspidomorfa miliaris</i>	26	1	4
Nakamura et al., 1989	<i>Aspidomorfa sanctaerucis</i>	18	1	4
	<i>Calandra orezae</i>		1	4
Birch, 1948	<i>Callosobruchus chinensis</i>	65-90	1-2	4
Tanaka, 1990	<i>Callosobruchus maculatus</i>	26-28	1	4
Messina, 1991	<i>Epilachna niponica</i>	cr	1-2	4
Ohgushi, 1991	<i>Lucilia cuprina</i>	250-4000	1	4
	<i>Oncopeltus fasciatus</i>	30	1-2	4
Readshaw y Gerwen, 1983	<i>Oncopeltus unifasciatus</i>	40	1-2	4
Landahl y Root, 1969				
Moluscos				
Perron, 1983	<i>Conus pennaceus</i>	cr	3	4
Hughes y Roberts, 1981	<i>Littorina rudis</i>	cr	3	4
Aves				
Ainley, 1980	<i>Pygoscelis adeliae</i>	cr	2	2
	<i>Megadyptes antipodes</i>	cr	2	2
Baker et al., 1981	<i>Zonotrichia leucophrys nuttalli</i>	43-141	2	1
cr=captura-recaptura				

Las funciones de sobrevivencia que usa el programa se definieron ajustando las curvas de sobrevivencia tipo I, tipo II y tipo III para un lapso de tiempo de 20 unidades. Se muestran en la figura 2. Las funciones 1, 2, 3 y 4 se comportan, en diferente medida, como una curva de tipo I, característica de los humanos en países desarrollados (Krebs, 1985), de *Alonella Globulosa*, de

Tabla II Continuación

Fuente	Especie	Tam. de muestra	lx	mx
Grant y Grant, 1992	<i>Gospiza fortis</i>	67-166	2	4
	<i>Gospiza scandens</i>	17-120	2	4
Reptiles				
Tinkle et al., 1981	<i>Chrysemis picta</i>	cr	3-2	2
Wilbur, 1975	<i>Chrysemis picta</i>	cr	3-2	2
Frazer et al., 1991	<i>Kinosternon subrubrum</i>	cr	3-2	2
Hasegawa, 1990	<i>Eumeces okadae</i>	cr	2	2
Ballinger, 1979	<i>Sceloporus jarrovi</i>	cr	2	2
Tinkle y Ballinger 1972	<i>Sceloporus undulatus</i>	cr	2	1
Vinegar, 1975	<i>Sceloporus virgatus</i>	cr	2	2
Maníferos				
Choquenot, 1991	<i>Equus asinus</i>	150	2-1	2
Slade y Balph, 1974	<i>Spermophilus armatus</i>	cr	2	2
Bronson, 1979	<i>Spermophilus lateralis</i>	85-197	2	2
Rickart, 1988	<i>Spermophilus mollis</i>	1-116	2	2

cr=captura-recaptura

Ceriodaphnia quadrangula (Neil, 1975); la función 5 es del tipo II y se presenta en muchas aves como *Megadyptes antipodes*, *Pygoscelis adeliae* (Ainley, 1980) y *Zonotrichia leucophrys nuttalli* (Baker et al., 1981) y en *Hydra* (Price, 1984). La 6 es característica de insectos parasitoides e invertebrados marinos (Price, 1984) y se comporta como el tipo III. Las funciones 5 y 6 no se definieron, sólo se proporcionó el valor de la lx para cada uno de los 20 intervalos.

Ecuaciones de las funciones de sobrevivencia:

$$1) l(x) = -0.05x + 1$$

$$2) l(x) = -\sqrt{0.05x} + 1$$

$$3) l(x) = \left(\frac{-x^2}{400} \right) + 1$$

$$4) l(x) = \sqrt{1 - \left(\frac{x^2}{400} \right)}$$

Las funciones de fecundidad se definieron para simular diferentes estrategias de reproducción reportadas en la literatura (tabla II). Todas las funciones se ajustaron para que los individuos se reprodujeran a partir de $t=5$. Estas se muestran en la figura 3.

Ecuaciones de las funciones de fecundidad:

$$1) m(x) = \frac{0.8x-4}{10}$$

$$2) m(x) = \frac{4\sqrt{(x-5)}+3}{10}$$

$$3) m(x) = \frac{50 - \sqrt{2500 \left(1 - \left(\frac{x^2}{400}\right)\right)}}{10}$$

$$4) m(x) = \frac{50e^{-\frac{(x-15)^2}{20}}}{10}$$

La función 1 presenta un aumento constante y pequeño en la fecundidad. La función 3 también aumenta; aunque no es lineal, sino exponencial. La función 2 presenta un aumento en la fecundidad con respecto del tiempo; sin embargo, cada vez este aumento es menor, es decir, se estabiliza. La última función presenta un aumento exponencial de la fecundidad hasta cierto punto en donde decae de la misma forma en que aumentó.

Después el programa realizó lo siguiente: creó las poblaciones reales o ideales (1000 individuos a lo largo de 20 intervalos de tiempo) a partir de las seis funciones de sobrevivencia y de las cuatro de fecundidad. Cada población tuvo una combinación diferente

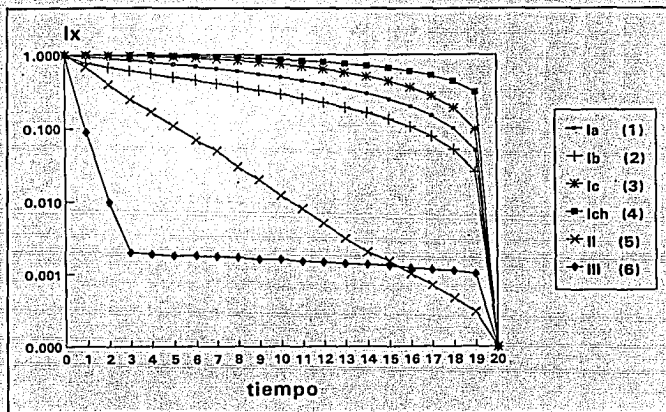


Figura 2. Curvas de sobrevivencia. El número entre paréntesis corresponde al identificador de la función en el programa.

de sobrevivencia y fecundidad, por lo que el total de poblaciones que se produjeron fueron 24. La nomenclatura que se usó para distinguir las 24 poblaciones consistió en poner el número de la función de sobrevivencia (1,2,3,4,5 6 6) y el número de la función de fecundidad (1,2,3 6 4) separados por un acento circunflejo.

λ : poblacional, real y estimada

Las lambdas poblacionales se obtuvieron a partir de las l_x y m_x a través del tiempo de los mil individuos de cada población utilizando el modelo matricial.

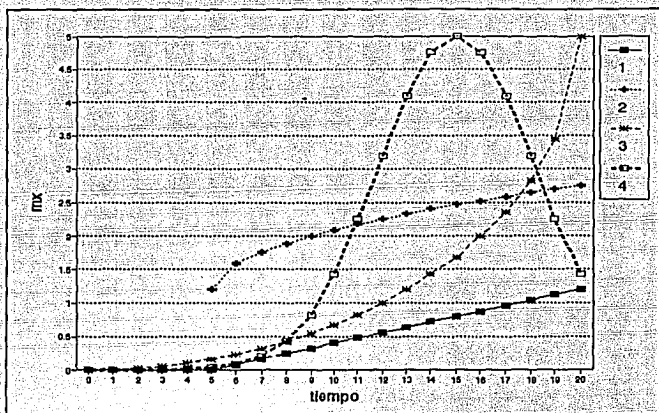


Figura 3 Curvas de fecundidad.

En las poblaciones con sobrevivencia 4, 5 y 6 se tomó una muestra de 5, 10, 20, 40 y 80 individuos (en las poblaciones 4*1, 5*1 y 6*1 se tomó una muestra adicional de 200 individuos), y de cada una de ellas se submuestreó de acuerdo a los métodos de bootstrap y jackknife. Con los datos de los individuos seleccionados se construyeron las matrices de transición para cada muestreo y se calculó λ . Ambos métodos produjeron una λ (λ^B o λ^J según el caso) con una varianza asociada. Por lo que al terminar se obtuvo una λ^B , λ^J , $V(\lambda^B)$ y $V(\lambda^J)$ (ecuaciones 26, 23, 27 y 24 respectivamente) de cada tamaño de muestra de las 12 poblaciones.

Para calcular la λ real de cada tamaño de muestra de todas las

poblaciones se realizaron mil muestreos, calculándose la matriz de transiciones y luego λ para cada muestra; se tomó la media de estas últimas como la λ real y la varianza como la varianza real. Se graficó la distribución de las mil lambdas agrupándolas en 12 categorías, y se construyó la gráfica de frecuencias. La normalidad se comprobó con la prueba de Kolmogorov-Smirnov.

Significancia

La prueba de significancia entre lambdas y varianzas reales con las lambdas y varianzas obtenidas mediante el bootstrap y el jackknife se realizó al 95% de confianza con las fórmulas utilizadas en Álvarez-Buylla y Slatkin (en prep.):

$$\frac{(n-1)(S^{*2})}{\chi^2_{[\alpha/2] \gamma=n-1}} \leq \sigma^2 \leq \frac{(n-1)(S^{*2})}{\chi^2_{[1-\alpha/2] \gamma=n-1}} \quad (25)$$

$$\bar{x}^* - T_{0.05|v} \frac{S^*}{\sqrt{n-1}} \leq \mu \leq \bar{x}^* + T_{0.05|v} \frac{S^*}{\sqrt{n-1}} \quad (26)$$

n =Número de simulaciones

S^{*2} =varianza estimada

\bar{x}^* =media estimada

μ =media real

σ^2 =varianza real

v =grados de libertad

Para estimar las entradas de las matrices se utilizó la fórmula de fenologías continuas (ecuación 6) y para la fecundidad la fórmula de mortalidad neonatal baja (ecuaciones 7 y 8) en las poblaciones con sobrevivencia 1, 2, 3 y 4. Con el fin de probar si

el método de estimación de las entradas de la matriz afecta el cálculo de λ , se tomó la población que presentó un mayor sesgo hacia la izquierda, la que presentó un mayor sesgo hacia la derecha y la más cercana a una distribución normal, y se volvió a calcular la λ real; pero ahora a partir de matrices para fenologías discretas con censo prerreproductivo (ecuación 12 y para la fecundidad la ecuación 13).

La construcción de las matrices de las poblaciones con sobrevivencia 4, 5 y 6 se hizo también a partir de la fórmula para fenologías continuas (ecuación 6); pero para estimar la fecundidad, se usó la fórmula de mortalidad neonatal alta (ecuaciones 7 y 9).

RESULTADOS Y DISCUSIÓN

Las lambdas poblacionales de las 24 poblaciones se presentan en la figura 4.

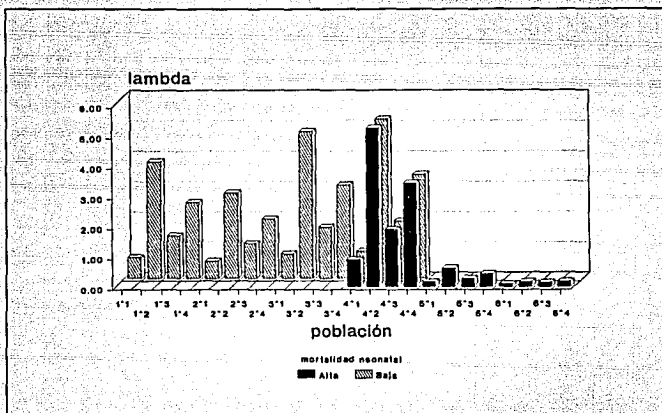


Figura 4 Magnitud de λ poblacional. El nombre de cada población se compone por el número de la función de sobrevivencia y el número de la función de fecundidad separados por un acento circunflejo.

En las tablas III y IV se ve claramente que el promedio de la tasa de crecimiento poblacional tiende al valor de λ poblacional conforme aumenta el tamaño de muestra a partir de las cuales se calcula. Esto es cierto para cualquier combinación de curvas de sobrevivencia y fecundidad. Este resultado nos sirve para verificar el modelo de simulación poblacional y el funcionamiento del

Tabla III Media y desviación típica de mil las obtenidas para cada tamaño de muestra de todas las poblaciones. Se muestra la λ poblacional. Las entradas de la matriz se estimaron con las fórmulas 6, 7 y 8.

Población	Tamaño de muestra												lambda pop.
	6		10		20		40		80				
	Media	DE	Media	DE	Media	DE	Media	DE	Media	DE			
1^1	0.308301	0.20237	0.451053	0.202534	0.595138	0.215513	0.852342	0.096881	0.872622	0.0400771			0.0767
1^2	3.768182	0.023907	3.819432	0.024871	3.859807	0.022177	3.866333	0.020085	3.851691	0.020701			3.9494
1^3	1.170052	0.463724	1.320525	0.285508	1.350485	0.187003	1.385087	0.10788	1.399210	0.072875			1.4038
1^4	2.248899	0.721427	2.423810	0.455903	2.505587	0.278470	2.488164	0.168913	2.500212	0.137768			2.5043
2^1			0.280691	0.272892	0.330501	0.249112	0.477754	0.182491	0.54887	0.083968			0.5695
2^2	2.502404	1.128090	2.778711	0.700541	2.804986	0.473873	2.704353	0.320646	2.815036	0.237386			2.8153
2^3	0.810432	0.543778	0.940210	0.38078	1.046214	0.230078	1.104505	0.145044	1.132094	0.090174			1.1306
2^4	1.540145	0.030782	1.786093	0.822255	1.865004	0.352087	1.940504	0.218022	1.950153	0.153348			1.9581
3^1	0.507401	0.252464	0.672958	0.237249	0.782269	0.116656	0.788978	0.043204	0.760051	0.026091			0.7915
3^2	4.270546	0.802573	4.894233	0.403893	4.830576	0.290080	4.827088	0.211395	4.828775	0.149052			4.8259
3^3	1.582223	0.292718	1.835289	0.18888	1.865207	0.122632	1.875764	0.081759	1.881113	0.055308			1.8819
3^4	3.012802	0.516589	3.080167	0.388913	3.060089	0.207114	3.073862	0.141403	3.074145	0.098855			3.0692
4^1	0.284917	0.215147	0.881337	0.080227	0.882914	0.042129	0.860558	0.028788	0.880056	0.016048			0.89
4^2	6.222234	0.46858	6.235531	0.317305	6.248524	0.219485	6.244143	0.183009	6.2457	0.111534			6.24
4^3	1.84918	0.210321	1.872803	0.125948	1.881805	0.085775	1.876272	0.060443	1.894011	0.04145			1.88
4^4	3.415218	0.345751	3.420546	0.227417	3.418041	0.151055	3.423833	0.114022	3.425939	0.078739			3.43

Tabla IV Mismas abreviaciones que en la tabla III. Las entradas de la matriz se calcularon conforme a las fórmulas de mortalidad neonatal alta: 6, 7 y 9.

Población	Tamaño de muestra												lambda pop.
	6		10		20		40		80				
	Media	DE	Media	DE	Media	DE	Media	DE	Media	DE			
4^1	0.307803	0.227681	0.787134	0.094751	0.865384	0.042555	0.877317	0.020812	0.880109	0.020788			0.88029
4^2	5.210853	0.450537	5.221731	0.310382	5.248041	0.224134	5.242068	0.157768	5.240126	0.106783			5.2429
4^3	1.850462	0.200416	1.874729	0.13034	1.879048	0.083185	1.882611	0.061851	1.883833	0.043782			1.8942
4^4	3.382270	0.248035	3.418473	0.224119	3.420732	0.152102	3.429040	0.110445	3.424803	0.077054			3.4249
5^1	0.008582	0.048248	0.008958	0.049548	0.012536	0.049337	0.009249	0.008025	0.028852	0.008054			0.1623
5^2	0.206713	0.049411	0.330981	0.372554	0.404889	0.315738	0.484793	0.242322	0.512394	0.188685			0.10061
5^3	0.017851	0.107833	0.026836	0.12334	0.040421	0.140007	0.061943	0.13612	0.095381	0.141676			0.284
5^4	0.047803	0.216487	0.055289	0.202518	0.077015	0.228575	0.149272	0.228417	0.216894	0.218986			0.4269
6^1	0.008548	0.053388	0.010112	0.081338	0.013056	0.081328	0.017811	0.083254	0.028958	0.066907			0.099
6^2	0.018423	0.170312	0.027173	0.177659	0.029401	0.140257	0.048643	0.150153	0.054285	0.137802			0.1626
6^3	0.012904	0.115392	0.018450	0.110581	0.010923	0.103081	0.030665	0.108881	0.047711	0.114551			0.1503
6^4	0.012505	0.149315	0.020827	0.15241	0.033185	0.150958	0.048818	0.158520	0.057165	0.144222			0.1789

programa. Los valores de λ promedio (Media) y las estimaciones de desviación típica (DE) de esta tabla se utilizaron como valores de referencia para calcular los sesgos de los estimados de bootstrap

y jackknife y su significancia estadística (ver fórmulas 27 a 30).

En las figuras 5 a 10 se grafican los sesgos de los estimados de bootstrap y jackknife para las λ con respecto al tamaño de muestra utilizado en la estimación de las mismas. Los sesgos se calcularon con las fórmulas siguientes:

$$\text{Sesgo de } \lambda_T = (\lambda_{\text{poblacional}} - \lambda_T) / \lambda_{\text{poblacional}} \quad (27)$$

$$\text{Sesgo de } \lambda_P = (\lambda_{\text{poblacional}} - \lambda_P) / \lambda_{\text{poblacional}} \quad (28)$$

$$\text{Sesgo de } \lambda_B = (\lambda_{\text{poblacional}} - \lambda_B) / \lambda_{\text{poblacional}} \quad (29)$$

$$\text{Sesgo de } \lambda_J = (\lambda_{\text{poblacional}} - \lambda_J) / \lambda_{\text{poblacional}} \quad (30)$$

Donde $\lambda_P = \lambda$ promedio $\lambda_T = \lambda$ total $\lambda_B = \lambda$ bootstrap $\lambda_J = \lambda$ jackknife

En general el sesgo disminuye conforme aumenta el tamaño de muestra. Sin embargo, en algunos casos los sesgos son bastante grandes y no alcanzan a disminuir con los tamaños máximos de muestra utilizados (figuras 5ch, 6ch, 7ch y 8ch). Este es el caso de poblaciones con una curva de fecundidad de tipo 4, en que la fecundidad máxima ocurre a edades intermedias. Todas las poblaciones con este tipo de fecundidad presentaron sesgos mayores y éstos no se acercaron a cero para los tamaños de muestra máximos utilizados ($n = 80$).

Los sesgos también fueron cada vez mayores conforme la sobrevivencia en edades tempranas disminuyó (de la curva tipo I a la tipo III). En el caso de poblaciones con curvas de sobrevivencia

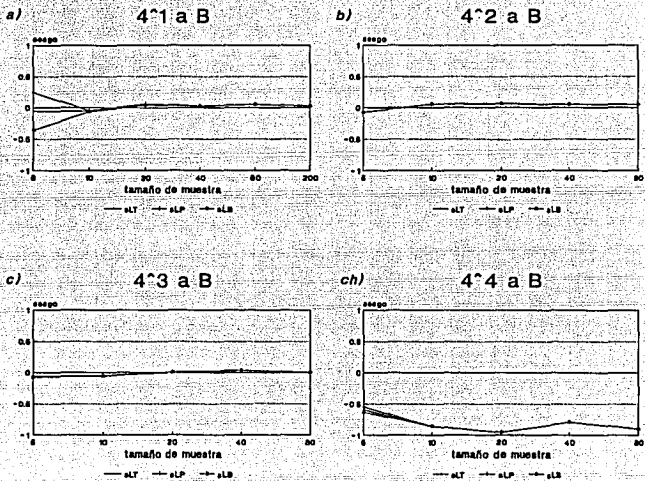


Figura 6. Sesgos de la lambda total, la lambda promedio y de la lambda bootstrap con respecto de la lambda poblacional. Los sesgos para cada tamaño de muestra se calcularon con las siguientes fórmulas:
 $aLT = (\text{lambda poblacional} - \text{lambda total}) / \text{lambda poblacional}$
 $aLP = (\text{lambda poblacional} - \text{lambda promedio}) / \text{lambda poblacional}$
 $aLB = (\text{lambda poblacional} - \text{lambda bootstrap}) / \text{lambda poblacional}$
 a - Alta mortalidad neonatal
 B - Bootstrap
 4¹ - Población con sobrevivencia 4 y fecundidad 1
 4² - Población con sobrevivencia 4 y fecundidad 2
 4³ - Población con sobrevivencia 4 y fecundidad 3
 4⁴ - Población con sobrevivencia 4 y fecundidad 4
 aLT - Sesgo de la lambda total
 aLP - Sesgo de la lambda promedio
 aLB - Sesgo de la lambda bootstrap

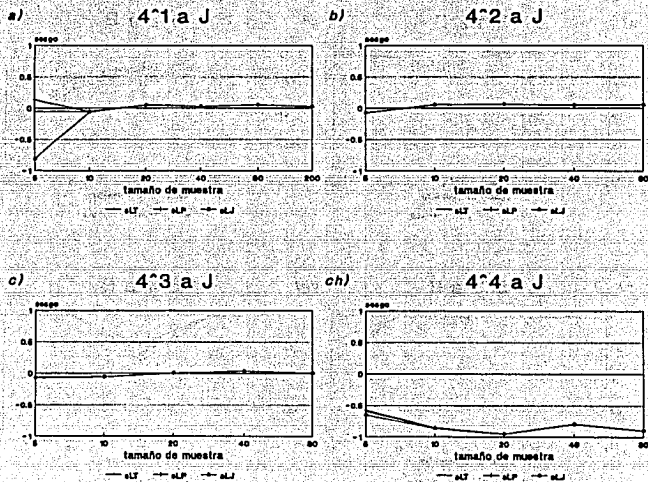


Figura 6. Sesgos de la lambda total, la lambda promedio y de la lambda jackknife con respecto de la lambda poblacional. Los sesgos para cada tamaño de muestra se calcularon con las siguientes fórmulas:
 sLT=(lambda poblacional-lambda total)/lambda poblacional
 sLP=(lambda poblacional-lambda promedio)/lambda poblacional
 sLJ=(lambda poblacional-lambda jackknife)/lambda poblacional
 a -Alta mortalidad neonatal
 J-Jackknife
 4*1-Población con sobrevivencia 4 y fecundidad 1
 4*2-Población con sobrevivencia 4 y fecundidad 2
 4*3-Población con sobrevivencia 4 y fecundidad 3
 4*4-Población con sobrevivencia 4 y fecundidad 4
 sLT-Sesgo de la lambda total
 sLP-Sesgo de la lambda promedio
 sLJ-Sesgo de la lambda jackknife

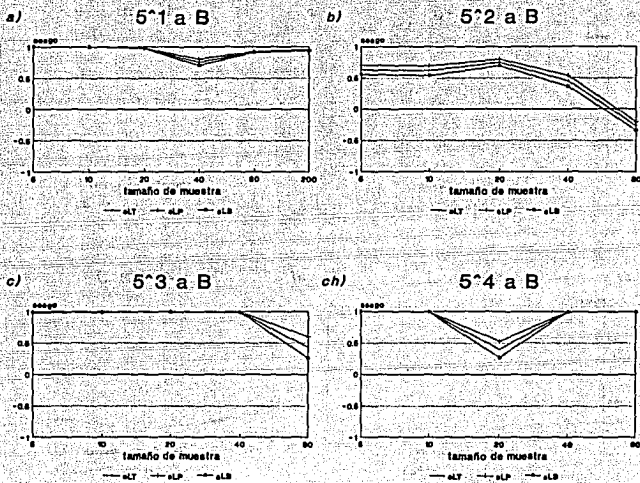


Figura 7. Sesgos de la lambda total, la lambda promedio y de la lambda bootstrap con respecto de la lambda poblacional. Los sesgos para cada tamaño de muestra se calcularon con las siguientes fórmulas:

- sLT = (lambda poblacional - lambda total) / lambda poblacional
- sLP = (lambda poblacional - lambda promedio) / lambda poblacional
- sLB = (lambda poblacional - lambda bootstrap) / lambda poblacional
- g = Alta mortalidad neonatal
- B = Bootstrap
- 5^1 - Población con sobrevivencia 5 y fecundidad 1
- 5^2 - Población con sobrevivencia 5 y fecundidad 2
- 5^3 - Población con sobrevivencia 5 y fecundidad 3
- 5^4 - Población con sobrevivencia 5 y fecundidad 4
- sLT - Sesgo de la lambda total
- sLP - Sesgo de la lambda promedio
- sLB - Sesgo de la lambda bootstrap

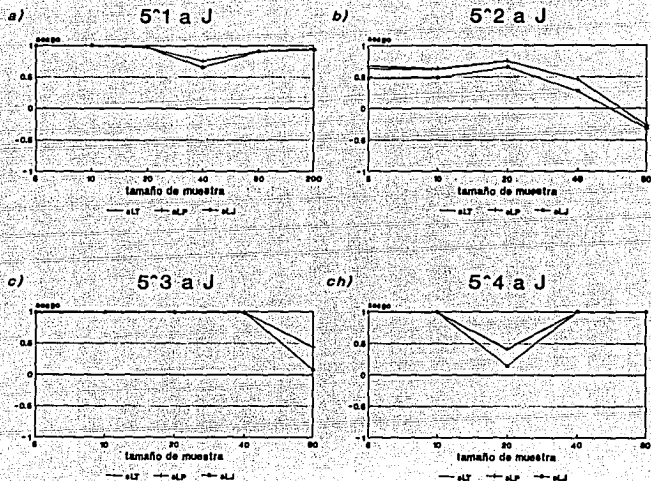


Figura 8. Sesgos de la lambda total, la lambda promedio y de la lambda jackknife con respecto de la lambda poblacional. Los sesgos para cada tamaño de muestra se calcularon con las siguientes fórmulas:
 $sLT = (\text{lambda poblacional} - \text{lambda total}) / \text{lambda poblacional}$
 $sLP = (\text{lambda poblacional} - \text{lambda promedio}) / \text{lambda poblacional}$
 $sLJ = (\text{lambda poblacional} - \text{lambda jackknife}) / \text{lambda poblacional}$
 a- Alta mortalidad neonatal
 J- Jackknife
 5¹- Población con sobrevivencia 5 y fecundidad 1
 5²- Población con sobrevivencia 5 y fecundidad 2
 5³- Población con sobrevivencia 5 y fecundidad 3
 5⁴- Población con sobrevivencia 5 y fecundidad 4
 sLT- Sesgo de la lambda total
 sLP- Sesgo de la lambda promedio
 sLJ- Sesgo de la lambda jackknife

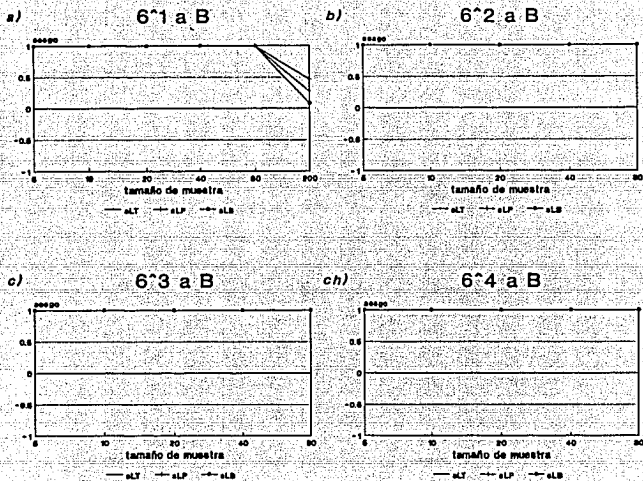


Figura 9. Sesgos de la lambda total, la lambda promedio y de la lambda bootstrap con respecto de la lambda poblacional. Los sesgos para cada tamaño de muestra se calcularon con las siguientes fórmulas:
 sLT=(lambda poblacional-lambda total)/lambda poblacional
 sLP=(lambda poblacional-lambda promedio)/lambda poblacional
 sLB=(lambda poblacional-lambda bootstrap)/lambda poblacional
 a-Alta mortalidad neonatal
 B-Bootstrap
 6¹-Población con sobrevivencia 6 y fecundidad 1
 6²-Población con sobrevivencia 6 y fecundidad 2
 6³-Población con sobrevivencia 6 y fecundidad 3
 6⁴-Población con sobrevivencia 6 y fecundidad 4
 sLT-Sesgo de la lambda total
 sLP-Sesgo de la lambda promedio
 sLB-Sesgo de la lambda bootstrap

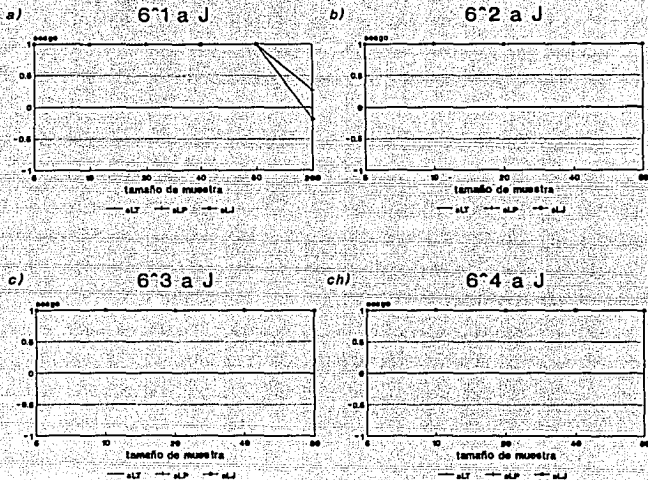


Figura 10. Sesgos de la lambda total, la lambda promedio y de la lambda jackknife con respecto de la lambda poblacional. Los sesgos para cada tamaño de muestra se calcularon con las siguientes fórmulas:

sLT=(lambda poblacional-lambda total)/lambda poblacional

sLP=(lambda poblacional-lambda promedio)/lambda poblacional

sLJ=(lambda poblacional-lambda jackknife)/lambda poblacional

a- Alta mortalidad neonatal

J-Jackknife

6^1-Población con sobrevivencia 6 y fecundidad 1

6^2-Población con sobrevivencia 6 y fecundidad 2

6^3-Población con sobrevivencia 6 y fecundidad 3

6^4-Población con sobrevivencia 6 y fecundidad 4

sLT-Sesgo de la lambda total

sLP-Sesgo de la lambda promedio

sLJ-Sesgo de la lambda jackknife

tipo III los sesgos fueron cercanos a uno, ya que en gran parte de las muestras los estimados de la tasa de crecimiento poblacional fueron iguales a cero. Solamente al utilizar una muestra de 200 individuos para generar la matriz poblacional se obtuvieron sesgos menores y la λ de jackknife o bootstrap se acercó a la λ poblacional (ver figura 9a y 10a).

Los sesgos de las varianzas del bootstrap y del jackknife se presentan en las figuras 11 y 12. En la mayoría de las poblaciones el sesgo disminuye conforme aumenta el tamaño de muestra. Para las poblaciones con sobrevivencia tipo II y III los sesgos no siempre se acercan a cero, sobre todo cuando tienen fecundidad tipo 1 ó 4 (figuras 11b, 11c, 12b y 12c).

El efecto de la sobrevivencia temprana en los sesgos en la estimación de λ ha sido reportado por otros autores. Meyer et al. (1986) encontraron sesgos mayores para poblaciones con altas mortalidades en etapas tempranas al utilizar los métodos de bootstrap y jackknife. Utilizando el método de Monte Carlo, Rago y Dorazio (1984) también encontraron altas probabilidades de obtener tasas de crecimiento de cero al utilizar muestras pequeñas en poblaciones con altas mortalidades en edades tempranas. Ambos autores estimaron las tasas de crecimiento poblacional por medio de la ecuación de Euler.

A pesar de que en varios casos los sesgos disminuyeron considerablemente conforme aumentó el tamaño de muestra, casi siempre fueron estadísticamente distintos de cero y significativos (tablas VI-XVII). En esta mismas tablas se ve que el sesgo de la

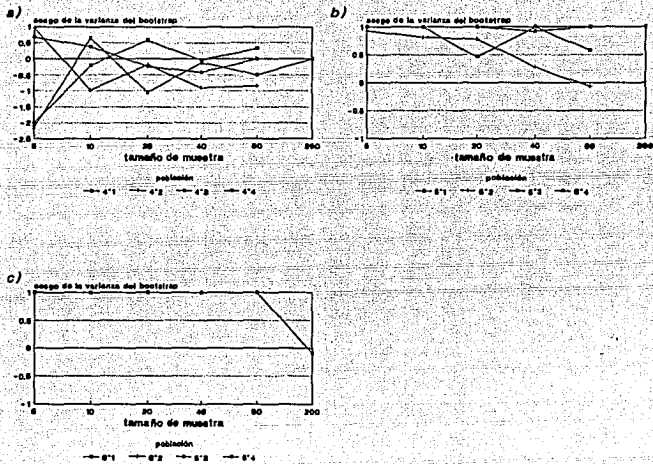


Figura 11. Sesgos de las varianzas del bootstrap con respecto de la varianza obtenida a partir de m lances de cada tamaño de muestra (ver tabla IV). Se calcularon con la fórmula siguiente: $(\text{varianza de ml} - \text{varianza del bootstrap}) / \text{varianza de ml}$

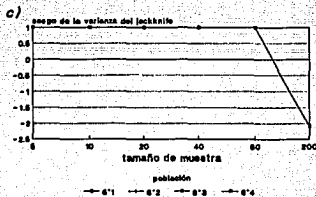
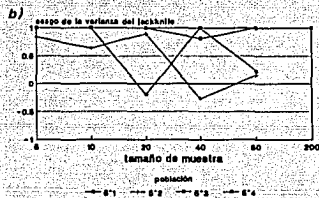
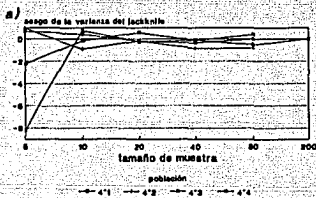


Figura 12. Sesgo de las varianzas del jackknife con respecto de la varianza obtenida a partir de mi lmbdas de cada tamaño de muestra (ver tabla IV). Se calcularon con la fórmula siguiente:

$$\frac{\text{varianza de mi-varianza del jackknife}}{\text{varianza de mi}}$$

Tabla V Lista de las abreviaciones usadas en las tablas VI-XVII

Tam	tamaño de muestra
Mu	media real de λ
Sig ²	varianza real de λ
ME	media de λ estimada con el bootstrap o el jackknife
VE	varianza de λ estimada con el bootstrap o el jackknife
Sm	prueba de significancia para la media de λ
Sv	prueba de significancia para la varianza de λ
LiM	límite inferior de la media de λ
LsM	límite superior de la media de λ
LiV	límite inferior de la varianza de λ
LsV	límite superior de la varianza de λ

Los límites forman el intervalo en el que debe caer el valor de la media y de la varianza estimadas para no ser significativamente diferentes de los valores reales.

Tabla VI

Tam	Mu	Sig2	ME	VE	Sm	Sv	LIM	LeM	LIV	LeV
4^1 bootstrap										
5	0.8078	0.0518	1.2065	0.1634	SI	SI	1.1814	1.2316	0.15	0.1788
10	0.8781	0.0072	0.9436	0.0025	SI	SI	0.9405	0.9467	0.0023	0.0027
20	0.8864	0.0019	0.863	0.0039	SI	SI	0.8491	0.8559	0.0036	0.0043
40	0.8873	0.0009	0.8656	0.001	SI	SI	0.8636	0.8676	0.0009	0.0011
80	0.8891	0.0004	0.8436	0.0005	SI	SI	0.8421	0.8451	0.0006	0.0007
200	0.889	0.0002	0.8648	0.0002	SI	SI	0.8639	0.8657	0.0002	0.0002
4^1 jackknife										
5	0.8078	0.0518	1.6161	0.4769	SI	SI	1.6723	1.6679	0.437	0.6208
10	0.8781	0.0072	0.9409	0.0019	SI	SI	0.9382	0.9436	0.0017	0.0021
20	0.8864	0.0019	0.8475	0.0024	SI	SI	0.8445	0.8505	0.0022	0.0026
40	0.8873	0.0009	0.8664	0.001	SI	SI	0.8644	0.8684	0.0009	0.0011
80	0.8891	0.0004	0.8448	0.0005	SI	SI	0.8433	0.8463	0.0006	0.0007
200	0.889	0.0002	0.8662	0.0002	SI	SI	0.8643	0.8661	0.0002	0.0002

Tabla VII

Tam	Mu	Sig2	ME	VE	Sm	Sv	LIM	LeM	LIV	LeV
4^2 bootstrap										
5	5.2109	0.2084	5.6277	0.007	SI	SI	5.6225	5.6329	0.0064	0.0077
10	5.2217	0.109	4.961	0.2153	SI	SI	4.9322	4.9898	0.1977	0.2356
20	5.2489	0.049	4.9437	0.058	SI	SI	4.9288	4.9686	0.0533	0.0636
40	5.2427	0.0249	4.9873	0.0475	SI	SI	4.9738	5.0008	0.0436	0.052
80	5.2401	0.012	4.9872	0.0222	SI	SI	4.978	4.9964	0.0204	0.0243
4^2 jackknife										
5	5.2109	0.2084	5.6267	0.0079	SI	SI	5.6202	5.6312	0.0073	0.0086
10	5.2217	0.109	4.9632	0.215	SI	SI	4.9344	4.992	0.1974	0.2353
20	5.2489	0.049	4.9293	0.0577	SI	SI	4.9144	4.9442	0.053	0.0631
40	5.2427	0.0249	4.9893	0.0462	SI	SI	4.976	5.0026	0.0424	0.0506
80	5.2401	0.012	4.9863	0.0217	SI	SI	4.9772	4.9954	0.0199	0.0237

Tabla VIII

Tem	Mu	Sig2	ME	VE	Sm	Sv	LIM	LSM	LV	LeV
4^3 bootstrap										
5	1.8505	0.0426	2.0207	0.0136	SI	SI	2.0135	2.0279	0.0125	0.0149
10	1.8747	0.017	1.9863	0.0105	SI	SI	1.9799	1.9927	0.0096	0.0115
20	1.879	0.0069	1.8765	0.0085	NS	SI	1.8697	1.8813	0.0079	0.0094
40	1.8826	0.0038	1.8148	0.0054	SI	SI	1.8102	1.8194	0.005	0.0059
80	1.8838	0.0019	1.6691	0.0019	SI	NS	1.8664	1.8718	0.0017	0.0021
4^3 jackknife										
5	1.8505	0.0426	2.0155	0.013	SI	SI	2.0084	2.0226	0.0119	0.0142
10	1.8747	0.017	1.9854	0.0099	SI	SI	1.9792	1.9916	0.0091	0.0108
20	1.879	0.0069	1.8733	0.0083	SI	SI	1.8677	1.8789	0.0076	0.0091
40	1.8826	0.0038	1.8167	0.0051	SI	SI	1.8113	1.8201	0.0047	0.0056
80	1.8838	0.0019	1.8712	0.0019	SI	NS	1.8685	1.8739	0.0017	0.0021

Tabla IX

Tem	Mu	Sig2	ME	VE	Sm	Sv	LIM	LSM	LV	LeV
4^4 bootstrap										
5	3.3823	0.1213	3.076	0.3646	SI	SI	3.0386	3.1134	0.3348	0.399
10	3.4155	0.0502	3.4947	0.0606	SI	SI	3.4794	3.51	0.0556	0.0663
20	3.4267	0.0231	3.678	0.0096	SI	SI	3.6719	3.6841	0.0088	0.0105
40	3.429	0.0121	3.3721	0.0125	SI	NS	3.3652	3.379	0.0115	0.0137
80	3.4248	0.0069	3.6808	0.004	SI	SI	3.6769	3.6847	0.0037	0.0044
4^4 jackknife										
5	3.3823	0.1213	3.1102	0.3926	SI	SI	3.0713	3.1491	0.3606	0.4297
10	3.4155	0.0502	3.6078	0.0592	SI	SI	3.4927	3.5229	0.0544	0.0648
20	3.4267	0.0231	3.6811	0.0104	SI	SI	3.6748	3.6874	0.0096	0.0114
40	3.429	0.0121	3.375	0.0138	SI	SI	3.3687	3.3833	0.0127	0.0151
80	3.4248	0.0069	3.679	0.0036	SI	SI	3.6753	3.6827	0.0033	0.0039

Tabla X

Tem	Mu	Sig2	ME	VE	Sm	Sv	LM	LeM	LIV	LeV
6^1 bootstrap										
5	0.0066	0.0021	0.0002	0	SI	SI	0.0002	0.0002	0	0
10	0.009	0.0021	0.0002	0	SI	SI	0.0002	0.0002	0	0
20	0.0126	0.0024	0.0039	0	SI	SI	0.0039	0.0039	0	0
40	0.0202	0.0037	0.0477	0.0003	SI	SI	0.0466	0.0488	0.0003	0.0003
80	0.0289	0.0048	0.0148	0	SI	SI	0.0148	0.0148	0	0
200	0.053	0.0068	0.0092	0	SI	SI	0.0092	0.0092	0	0
6^1 jackknife										
5	0.0066	0.0021	0.0002	0	SI	SI	0.0002	0.0002	0	0
10	0.009	0.0021	0.0002	0	SI	SI	0.0002	0.0002	0	0
20	0.0126	0.0024	0.0039	0	SI	SI	0.0039	0.0039	0	0
40	0.0202	0.0037	0.0558	0.0007	SI	SI	0.0542	0.0574	0.0006	0.0008
80	0.0289	0.0048	0.0146	0	SI	SI	0.0146	0.0146	0	0
200	0.053	0.0068	0.0105	0	SI	SI	0.0105	0.0105	0	0

Tabla XI

Tem	Mu	Sig2	ME	VE	Sm	Sv	LM	LeM	LIV	LeV
6^2 bootstrap										
5	0.2967	0.2446	0.2685	0.0171	SI	SI	0.2604	0.2766	0.0157	0.0187
10	0.331	0.1388	0.2805	0.0266	SI	SI	0.2706	0.2904	0.0236	0.028
20	0.4049	0.0997	0.1847	0.0104	SI	SI	0.1784	0.191	0.0095	0.0114
40	0.4648	0.0592	0.3871	0.0427	SI	SI	0.3743	0.3999	0.0392	0.0467
80	0.5138	0.0356	0.8168	0.0378	SI	NS	0.8037	0.8279	0.0347	0.0414
6^2 jackknife										
5	0.2967	0.2446	0.318	0.04	SI	SI	0.3056	0.3304	0.0367	0.0438
10	0.331	0.1388	0.3142	0.0505	SI	SI	0.3003	0.3281	0.0464	0.0553
20	0.4049	0.0997	0.2088	0.0222	SI	SI	0.1996	0.218	0.0204	0.0243
40	0.4648	0.0592	0.4461	0.0752	SI	SI	0.4281	0.4621	0.069	0.0823
80	0.5138	0.0356	0.8013	0.0301	SI	SI	0.7905	0.8121	0.0276	0.0329

Tabla XII

Term	Mu	Sig2	ME	VE	Sm	Sv	LM	LoM	LV	LoV
5^3 bootstrap										
5	0.0178	0.0116	0.006	0	SI	SI	0.005	0.005	0	0
10	0.0298	0.0152	0.0047	0	SI	SI	0.0047	0.0047	0	0
20	0.0484	0.0196	0.0027	0	SI	SI	0.0027	0.0027	0	0
40	0.0619	0.0185	0.0035	0	SI	SI	0.0035	0.0035	0	0
80	0.0954	0.0201	0.2087	0.0084	SI	SI	0.203	0.2144	0.0077	0.0092
6^3 jackknife										
5	0.0178	0.0116	0.0061	0	SI	SI	0.0061	0.0061	0	0
10	0.0298	0.0152	0.0045	0	SI	SI	0.0045	0.0045	0	0
20	0.0484	0.0196	0.0033	0	SI	SI	0.0033	0.0033	0	0
40	0.0619	0.0185	0.0035	0	SI	SI	0.0035	0.0035	0	0
80	0.0954	0.0201	0.2506	0.0246	SI	SI	0.2508	0.2702	0.0225	0.0269

Tabla XIII

Term	Mu	Sig2	ME	VE	Sm	Sv	LM	LoM	LV	LoV
5^4 bootstrap										
5	0.0479	0.0469	0.0002	0	SI	SI	0.0002	0.0002	0	0
10	0.0653	0.041	0.0038	0	SI	SI	0.0038	0.0038	0	0
20	0.0979	0.0523	0.3112	0.0283	SI	SI	0.3008	0.3216	0.026	0.031
40	0.1493	0.0522	0.0025	0	SI	SI	0.0025	0.0025	0	0
80	0.2167	0.0484	0.0028	0	SI	SI	0.0028	0.0028	0	0
6^4 jackknife										
5	0.0479	0.0469	0.0002	0	SI	SI	0.0002	0.0002	0	0
10	0.0653	0.041	0.0046	0	SI	SI	0.0046	0.0046	0	0
20	0.0979	0.0523	0.3543	0.0626	SI	SI	0.3488	0.3798	0.0575	0.0685
40	0.1493	0.0522	0.0025	0	SI	SI	0.0025	0.0025	0	0
80	0.2167	0.0484	0.0028	0	SI	SI	0.0028	0.0028	0	0

Tabla XIV

Tam	Mu	Sig2	ME	VE	Sm	Sv	LIM	LSM	LV	LSV
6~1 bootstrap										
5	0.0055	0.0029	0.0002	0	SI	SI	0.0002	0.0002	0	0
10	0.0101	0.004	0.0002	0	SI	SI	0.0002	0.0002	0	0
20	0.0131	0.004	0.0002	0	SI	SI	0.0002	0.0002	0	0
40	0.0178	0.004	0.0002	0	SI	SI	0.0002	0.0002	0	0
80	0.0304	0.0049	0.0002	0	SI	SI	0.0002	0.0002	0	0
200	0.0458	0.0045	0.1494	0.0049	SI	SI	0.1451	0.1637	0.0045	0.0054
6~1 jackknife										
5	0.0055	0.0029	0.0002	0	SI	SI	0.0002	0.0002	0	0
10	0.0101	0.004	0.0002	0	SI	SI	0.0002	0.0002	0	0
20	0.0131	0.004	0.0002	0	SI	SI	0.0002	0.0002	0	0
40	0.0178	0.004	0.0002	0	SI	SI	0.0002	0.0002	0	0
80	0.0304	0.0049	0.0002	0	SI	SI	0.0002	0.0002	0	0
200	0.0458	0.0045	0.191	0.0139	SI	SI	0.1837	0.1983	0.0128	0.0162

Tabla XV

Tam	Mu	Sig2	ME	VE	Sm	Sv	LIM	LSM	LV	LSV
6~2 bootstrap										
5	0.0164	0.029	0.0002	0	SI	SI	0.0002	0.0002	0	0
10	0.0272	0.0314	0.0002	0	SI	SI	0.0002	0.0002	0	0
20	0.0264	0.0197	0.0002	0	SI	SI	0.0002	0.0002	0	0
40	0.0468	0.0226	0.0002	0	SI	SI	0.0002	0.0002	0	0
80	0.0583	0.019	0.0002	0	SI	SI	0.0002	0.0002	0	0
6~2 jackknife										
5	0.0164	0.029	0.0002	0	SI	SI	0.0002	0.0002	0	0
10	0.0272	0.0314	0.0002	0	SI	SI	0.0002	0.0002	0	0
20	0.0264	0.0197	0.0002	0	SI	SI	0.0002	0.0002	0	0
40	0.0468	0.0226	0.0002	0	SI	SI	0.0002	0.0002	0	0
80	0.0583	0.019	0.0002	0	SI	SI	0.0002	0.0002	0	0

Tabla XVI

Term	Mu	Sig2	ME	VE	Sm	Sv	LM	LoM	LV	LoV
6^3 bootstrap										
5	0.0129	0.0133	0.0002	0	SI	SI	0.0002	0.0002	0	0
10	0.0185	0.0136	0.0002	0	SI	SI	0.0002	0.0002	0	0
20	0.0199	0.0107	0.0002	0	SI	SI	0.0002	0.0002	0	0
40	0.0307	0.0118	0.0002	0	SI	SI	0.0002	0.0002	0	0
80	0.0477	0.0131	0.0002	0	SI	SI	0.0002	0.0002	0	0
6^3 jackknife										
5	0.0129	0.0133	0.0002	0	SI	SI	0.0002	0.0002	0	0
10	0.0185	0.0136	0.0002	0	SI	SI	0.0002	0.0002	0	0
20	0.0199	0.0107	0.0002	0	SI	SI	0.0002	0.0002	0	0
40	0.0307	0.0118	0.0002	0	SI	SI	0.0002	0.0002	0	0
80	0.0477	0.0131	0.0002	0	SI	SI	0.0002	0.0002	0	0

Tabla XVII

Term	Mu	Sig2	ME	VE	Sm	Sv	LIM	LoM	LV	LoV
6^4 bootstrap										
5	0.0126	0.0223	0.0002	0	SI	SI	0.0002	0.0002	0	0
10	0.0208	0.0232	0.0002	0	SI	SI	0.0002	0.0002	0	0
20	0.0332	0.0256	0.0002	0	SI	SI	0.0002	0.0002	0	0
40	0.0468	0.0245	0.0002	0	SI	SI	0.0002	0.0002	0	0
80	0.0672	0.0208	0.0002	0	SI	SI	0.0002	0.0002	0	0
6^4 jackknife										
5	0.0126	0.0223	0.0002	0	SI	SI	0.0002	0.0002	0	0
10	0.0208	0.0232	0.0002	0	SI	SI	0.0002	0.0002	0	0
20	0.0332	0.0256	0.0002	0	SI	SI	0.0002	0.0002	0	0
40	0.0468	0.0245	0.0002	0	SI	SI	0.0002	0.0002	0	0
80	0.0672	0.0208	0.0002	0	SI	SI	0.0002	0.0002	0	0

varianza también fue significativamente distinto a cero para la mayoría de los tamaños de muestra y para la mayoría de las poblaciones. Sin duda estos resultados contrastan con los que obtuvieron, con la ecuación de Euler, Meyer et al. (1986). Es posible que las aproximaciones hechas al construir la matriz puedan estar produciendo sesgos considerables cuando los tamaños de muestra son chicos. Esta idea se apoya también con los resultados obtenidos al usar diferentes métodos de aproximación.

En las figuras 13 a 20 se presentan las distribuciones muestrales de λ para las diferentes poblaciones y tamaños de muestra. En todos los casos las distribuciones tienden a una distribución normal conforme aumenta el tamaño de muestra. No obstante, las distribuciones siempre tienen colas largas hacia valores pequeños para poblaciones con curvas de sobrevivencia I y hacia valores altos para poblaciones con curvas de sobrevivencia II y III. En estos últimos casos hubo una alta probabilidad de obtener tasas de crecimiento iguales a cero. En algunos casos para tamaños de muestra reducidos (alta mortalidad o baja fecundidad), las distribuciones son bimodales (ver figuras 13a, 14a, 15a, 18a, 19a, 19c). Este tipo de resultados ha sido obtenido antes por otros autores (Rago y Dorazio 1984; Meyer et al. 1986). Todas las distribuciones fueron significativamente distintas de una normal. Rago y Dorazio (1984) reportan, para simulaciones de Monte Carlo, que la fecundidad sólo afecta la media de la distribución; sin embargo, todas las distribuciones bimodales se originan de la combinación de cualquier sobrevivencia con la fecundidad tipo 1.

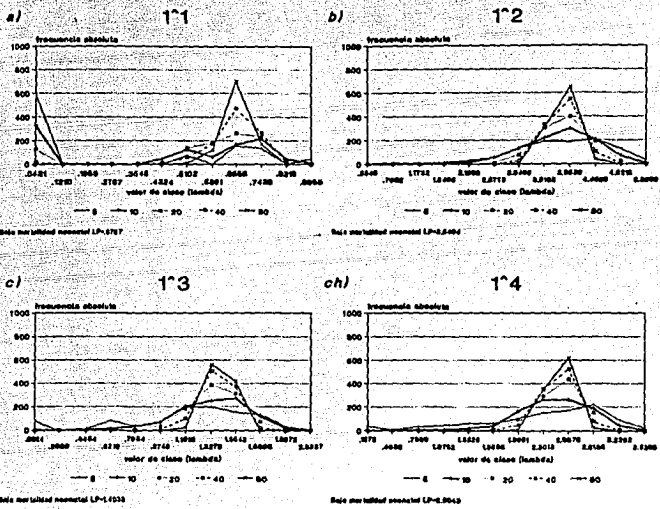


Figura 13. Distribución de las mil lámparas de cada tamaño de muestra.
 Los tamaños de muestra son: 5, 10, 20, 40, y 80.
 LP—Lambdas poblacional
 1*1—Población con supervivencia 1 y fecundidad 1
 1*2—Población con supervivencia 1 y fecundidad 2
 1*3—Población con supervivencia 1 y fecundidad 3
 1*4—Población con supervivencia 1 y fecundidad 4

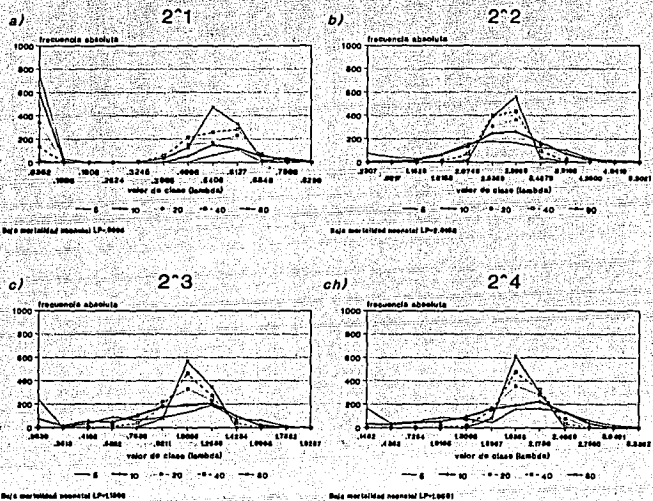


Figura 14. Distribución de las λ medidas de cada tamaño de muestra.
 Los tamaños de muestra son: 5, 10, 20, 40 y 80.
 LP=Lambda poblacional
 2¹-Población con supervivencia 2 y fecundidad 1
 2²-Población con supervivencia 2 y fecundidad 2
 2³-Población con supervivencia 2 y fecundidad 3
 2⁴-Población con supervivencia 2 y fecundidad 4

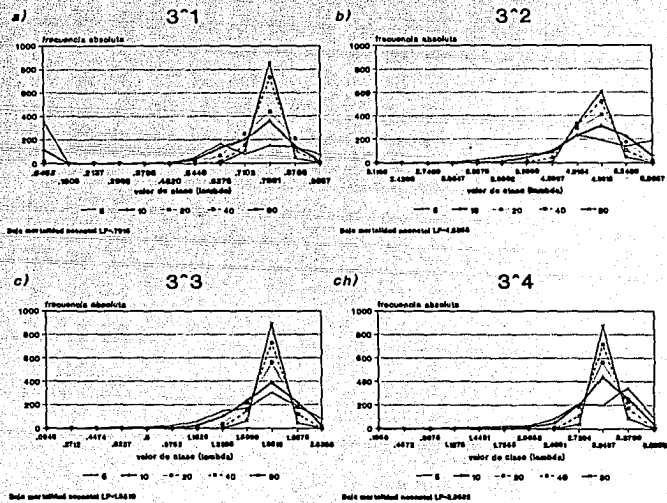
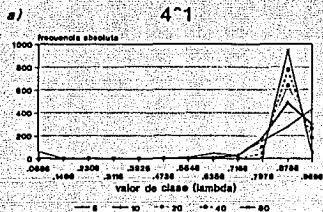
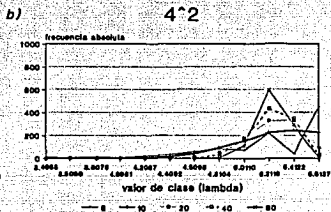


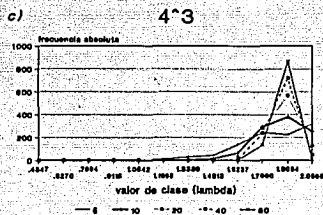
Figura 15. Distribución de las mil lambdas de cada tamaño de muestra.
 Los tamaños de muestra son: 5, 10, 20, 40 y 60.
 LP=Lambda poblacional
 3^1 -Población con supervivencia 3 y fecundidad 1
 3^2 -Población con supervivencia 3 y fecundidad 2
 3^3 -Población con supervivencia 3 y fecundidad 3
 3^4 -Población con supervivencia 3 y fecundidad 4



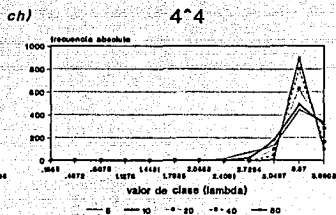
Bajo mortalidad neonatal LP=0.02021



Bajo mortalidad neonatal LP=0.2424



Bajo mortalidad neonatal LP=0.02018



Bajo mortalidad neonatal LP=0.02196

Figura 16. Distribución de los λ de cada tamaño de muestra.

Los tamaños de muestra son 5, 10, 20, 40 y 80.

LP=Lambda poblacional

4^1 -Población con supervivencia 4 y fecundidad 1

4^2 -Población con supervivencia 4 y fecundidad 2

4^3 -Población con supervivencia 4 y fecundidad 3

4^4 -Población con supervivencia 4 y fecundidad 4

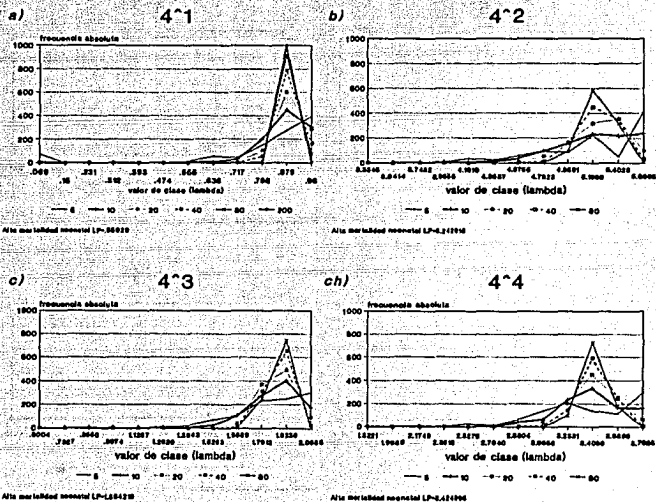


Figura 17. Distribución de las mil lambdas de cada tamaño de muestra.

Los tamaños de muestra son: 5, 10, 20, 40, 80 y 200.
 LP-Lambda población!

4¹-Población con supervivencia 4 y fecundidad 1

4²-Población con supervivencia 4 y fecundidad 2

4³-Población con supervivencia 4 y fecundidad 3

4⁴-Población con supervivencia 4 y fecundidad 4

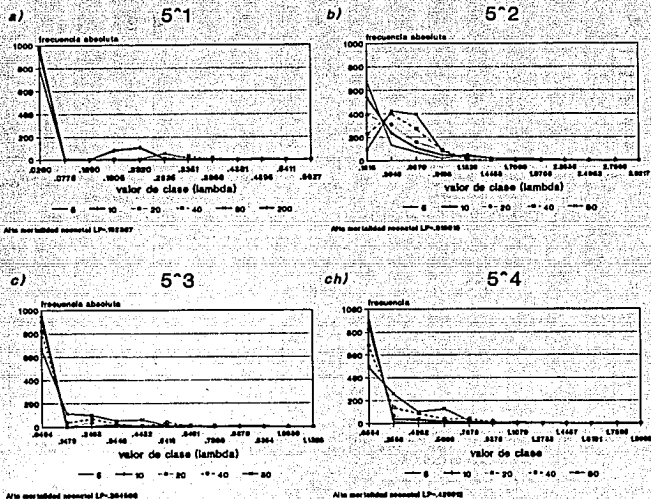


Figura 18. Distribución de las n lambda de cada tamaño de muestra.

Los tamaños de muestra son: 5, 10, 20, 40, 80 y 200.

LP-Lambda poblacional

5¹-Población con supervivencia 5 y fecundidad 1

5²-Población con supervivencia 5 y fecundidad 2

5³-Población con supervivencia 5 y fecundidad 3

5⁴-Población con supervivencia 5 y fecundidad 4

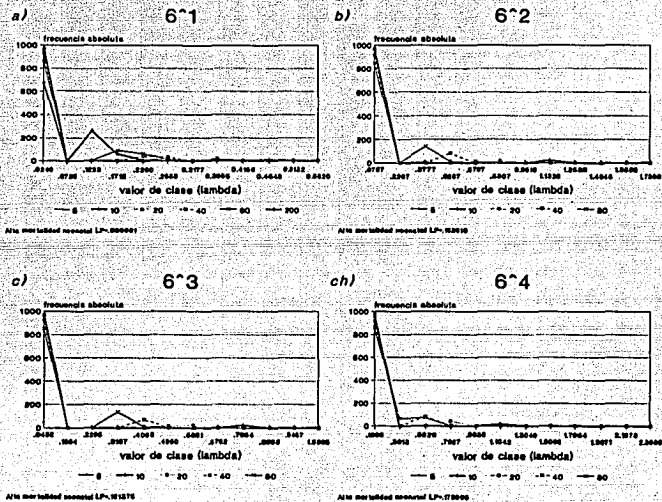
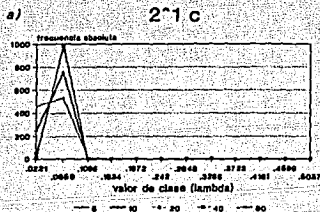
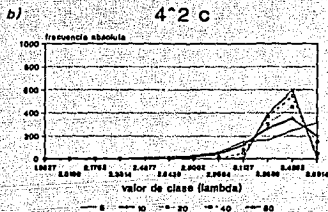


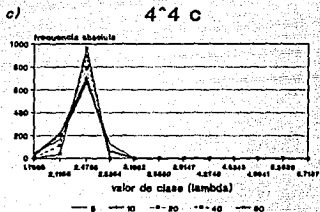
Figura 10. Distribución de las mil lambdas de cada tamaño de muestra.
 Los tamaños de muestra son: 5, 10, 20, 40, 80 y 200.
 LP=Lambda poblacional
 6¹-Población con supervivencia 6 y fecundidad 1
 6²-Población con supervivencia 6 y fecundidad 2
 6³-Población con supervivencia 6 y fecundidad 3
 6⁴-Población con supervivencia 6 y fecundidad 4



Base estadística muestral LP=0,07402



Base estadística muestral LP=0,80072



Base estadística muestral LP=0,4706

Figura 20. Distribución de las λ de cada tamaño de muestra.

Los tamaños de muestra son: 5, 10, 20, 40 y 80.

LP=Lambda poblacional

2^*1 -Población con supervivencia 2 y fecundidad 1

4^*2 -Población con supervivencia 4 y fecundidad 2

4^*4 -Población con supervivencia 4 y fecundidad 4

Para estas tres poblaciones las entradas de la matriz de

transiciones se calcularon a partir de las fórmulas para

poblaciones con reproducción continua y censo

Demográfico (ver fórmulas 10 y 17)

c-Reproducción continua

Analizando los valores de λ poblacional en la tabla IV, percibimos que siempre la λ de menor valor para el mismo tipo de sobrevivencia está asociada con la fecundidad tipo 1. Valores muy pequeños en la fecundidad pueden producir una distribución bimodal.

En poblaciones con sobrevivencia tipo I la bimodalidad se orienta hacia la izquierda, lo cual concuerda con Rago y Dorazio (1984); pero para poblaciones con sobrevivencias tipo II y III se orienta hacia la derecha. Rago y Dorazio (1984) no estudiaron poblaciones con sobrevivencia tipo III, porque usaron la distribución Weibull con el parámetro c mayor o igual a uno, lo que origina curvas tipo I o II (Pinder et al., 1978).

La figura 20 enseña las distribuciones muestrales de λ obtenidas al utilizar un método diferente para estimar las entradas de la matriz. Es importante notar que tanto los valores medios como las formas de las curvas difieren mucho según el método usado para calcular las entradas de la matriz de transiciones.

De los resultados anteriores se puede concluir que el tamaño de muestra que se utiliza para estimar la tasa de crecimiento afecta la precisión con la que los métodos de submuestreo de jackknife y bootstrap estiman los límites de confianza de la misma. Además, el efecto del tamaño de muestra depende de la demografía de la especie bajo estudio. En un estudio futuro se tendrá que cuantificar el efecto que los sesgos en la estimación de la λ promedio y su varianza tienen en la estimación de los límites de confianza de la λ (ver Meyer et al. 1986).

Ya que el tamaño de muestra afecta tanto la precisión de la

estimación de λ , es importante revisar los tamaños de muestra que generalmente se usan en estudios de campo. En la tabla II se enlistan los tamaños de muestra y los tipos de curvas de supervivencia y fecundidad para 32 estudios demográficos. En todos se calculó λ usando la ecuación de Euler. La mayoría de los casos reportan un tamaño de muestra menor de 80 individuos, que es el número máximo utilizado en la mayor parte de las simulaciones. Solamente 10 estudios sobrepasaron esta cifra. Con base en los resultados de este trabajo, se podría asegurar que estimados de bootstrap o jackknife de los límites de confianza de las λ para estos estudios estarían significativamente sesgados de haberse calculado con modelos matriciales. Los mayores sesgos se esperan para especies como *Muricea fruticosa*, *M. californica*, *Littorina rudis* y *Conus pennaceus* que tienen curvas de fecundidad de tipo 1 ó 4 y curvas de supervivencia de tipo III. Para tamaños de muestra menores de 30 se esperan también sesgos muy grandes.

Meyer et al. (1986) concluyen que un tamaño de muestra de 10 es suficientemente bueno para estimar λ utilizando el jackknife y el bootstrap, siempre que la mortalidad juvenil no sea alta y se calcule λ a partir de la ecuación de Euler. Los resultados mostrados aquí indican que un tamaño de 80 no es suficientemente bueno para los modelos matriciales. Habría que explorar si al construir la matriz se pierde precisión al estimar λ , qué tamaño de muestra se necesita como mínimo para tener un estimador confiable con el modelo matricial y de qué manera influye el número de entradas de aquélla sobre el estimador. Este tipo de estudios se

tendría que hacer también para especies estructuradas por otro tipo de características además de la edad, como el tamaño y el estadio, debido a que describen mejor la demografía de organismos como insectos y sobre todo de plantas.

CONCLUSIONES

- * El tamaño de muestra afecta la distribución muestral de λ y el sesgo del estimador de bootstrap y de jackknife.
- * Las aproximaciones hechas al estimar las entradas de la matriz originan sesgos significativos en los estimadores de bootstrap y jackknife de λ aun con tamaños de muestra considerables (más de 80 individuos).
- * La sobrevivencia afecta la distribución muestral de λ .
- * Fecundidades muy pequeñas originan distribuciones muestrales de λ bimodales.
- * La distribución muestral de λ generada por el submuestreo no es de tipo normal.
- * La dirección de las colas y la bimodalidad de la distribución muestral de λ dependen del tipo de sobrevivencia.
- * La manera de calcular las entradas de la matriz afecta la media y la forma de la distribución muestral de λ .
- * La mortalidad neonatal sólo afecta ligeramente la media de la distribución muestral de λ .

APÉNDICE A

El programa

El programa está escrito en Turbo Pascal 5.5. Consta de un programa principal y de varias unidades.

La noción de unidad acentúa el carácter estructurado y modular de Turbo Pascal para el diseño de grandes programas. Las unidades presentan la propiedad de ser compiladas aisladamente. Esto facilita la creación de bibliotecas. Según los cánones de programación modular, un programa grande se divide en unidades que agrupan procedimientos por su función. Además de facilitar la comprensión y la depuración del programa, la compilación independiente de las unidades permite también superar el tamaño del código fuente de un programa de 64 kp. que viene limitado por el tamaño de un segmento del microprocesador; esto significa que ningún programa principal y cualquier segmento dado no puede exceder un tamaño de 64 kp. (Joyanes, 1990).

Turbo Pascal manipula las unidades de forma que el límite superior de almacenamiento ocupado por un programa puede llegar al límite máximo de memoria que puede soportar la máquina y el sistema operativo: 640 kp. en la mayoría de las computadoras personales. Sin las unidades, el código de los programas está limitado a 64 kp.

El programa principal se encarga de mostrar un menú con diferentes procesos, como el de crear una población y muestrear de ella. Después de que se ha escogido el proceso, llama a la unidad correspondiente. Una vez que termina algún proceso, pregunta si se

quiere realizar otro; si se contesta que no, el programa termina. Se llama ARUSHKA.PAS.

Las unidades son POBYMUE.TPU, ABSOLUTA.TPU, BUU.TPU, NAVA-JA.TPU, LXXM.TPU, LAMDARCH.TPU, DESVARJ.TPU, DESVARB.TPU, INFORMA.TPU, KSSI.TPU, ADORNO.TPU, GRAFI.TPU y VARGLOB.TPU. El listado del programa y de las unidades se encuentra en el Apéndice B. El diagrama de flujo se representa en la figura 21 en el mismo apéndice.

ARUSHKA.PAS

Proporciona las instrucciones para manejar el programa, inicializa algunas variables, enseña el menú de opciones y da la condición de salida.

POBYMUE.TPU

Esta unidad se compone de dos procedimientos: GPoblación y Muestras. El primero solicita una función de supervivencia y una de fecundidad y, a partir de ellas, genera a la población total. La población total está formada por mil individuos que a lo largo del tiempo se reproducen y mueren según la función de fecundidad y la función de sobrevivencia seleccionadas.

La longevidad de ningún individuo excede a $t=20$, es decir, la l_x en $t=20$ es cero. Se agrupa en categorías y se calcula las l_x y m_x de cada una, para posteriormente calcular la λ real. La vida de los individuos se representa con números; por ejemplo, si un individuo está vivo será un 0, si está vivo y además tiene

descendencia de dos hijos en el lapso de tiempo de la categoría se representará como un 2. Al morir un individuo se convertirá en un 99. Todo el transcurso de la vida de los mil individuos a lo largo de los 20 días está guardado en un archivo de palabras (bytes) con el nombre de PT y la extensión formada con el número que corresponde a la función de sobrevivencia escogida (1 a 6), un separador (^) y el número que corresponde a la función de fecundidad escogida (1 a 4). Si se escogió la $l(x)$ 1 y la $m(x)$ 3, la población se llamará PT.1^3. Este archivo tiene un tamaño de 20 kilopalabras.

El procedimiento Muestras solicita el tamaño de muestra, el número de muestras y el nombre del archivo que contiene a la población a muestrear. Pregunta el tipo de reproducción de la población, el tipo de censo y si la mortalidad neonatal es alta o no. Todos estos datos se usarán en la unidad LXXM para construir la matriz.

Ejemplo numérico

En este ejemplo se describe cómo se genera la población total, se muestrea de ella, se calculan las l_x y m_x de la muestra y agrupan en categorías (unidad POBYMUE).

Para fines prácticos se creará una población de 10 individuos a lo largo de 20 años y se guardará en un arreglo de 10 x 20. Al terminar se obtendrá el siguiente arreglo:

individuos										tiempo
1	2	3	4	5	6	7	8	9	10	
0	0	0	99	0	0	0	0	0	0	1
99	0	0	99	0	0	0	0	0	0	2
99	0	99	99	0	0	0	0	0	0	3
99	1	99	99	1	1	1	99	1	1	4
99	2	99	99	2	2	2	99	2	2	5
:	:	:	:	:	:	:	:	:	:	:
99	99	99	99	99	99	99	99	99	99	20

- 0 _____ # individuo vivo con cero hijos
 1 _____ # individuo vivo con un hijo
 99 _____ # individuo muerto

En donde se almacena el transcurso de la vida de cada uno de los 10 individuos en 20 años.

CREACIÓN

Para generar la población total se requiere dar la función de sobrevivencia y la de fecundidad e ir calculando la l_x y la m_x desde $t=1$ hasta $t=20$. Supongamos que los valores que arrojaron estas funciones para los primeros 5 años son los que se tabulan a continuación:

t	l_x	m_x
1	.9	0
2	.8	0
3	.7	0
4	.6	0
5	.5	1

Para $t=1$ la l_x es .9, lo que quiere decir que si nuestra población tiene 10 individuos vivos en $t=0$, en $t=1$ tendrá el 90% del total vivos, muere el 10%. Para este caso morirá 1 individuo =

10-(10*9); así que se debe matar 1. Para esto se llama al procedimiento Mata que escoge al azar un individuo y lo mata marcándolo con un 99. Si el individuo que muere es el número 4, la población se ve en este momento así:

individuos										tiempo
1	2	3	4	5	6	7	8	9	10	
			99							1
										2
										3
										4
										5
										:
										20

Se prosigue a calcular la fecundidad de los individuos que continúan vivos. Para $t=1$ la función de fecundidad es cero. El arreglo queda de la siguiente manera:

individuos										tiempo
1	2	3	4	5	6	7	8	9	10	
0	0	0	99	0	0	0	0	0	0	1
										2
										3
										4
										5
										:
										20

De esta manera se genera la población hasta $t=20$.

OBSERVACIONES

- ⊕ Todas las funciones de m_x definidas en el programa producen 0 hijos desde $t=1$ hasta $t=4$.
- ⊕ Los individuos producen descendencia a partir de $t=5$ hasta su

muerte.

⊕ Si la función de fecundidad arroja un número real, digamos 3.3, el individuo se representa como 3: se trunca la fracción.

⊕ Si la función de lx da un resultado real, como por ejemplo, que deban morir 5.3 individuos, la fracción se guarda en un acumulador y se matan por lo pronto 5 individuos; para la siguiente t deben morir 6.8 individuos, ahora se matan 6 y se suma la fracción al acumulador $.8+3$ lo que da 1.1 y se mata 1 individuo más como resultado de la suma de las partes fraccionarias, restamos 1 del acumulador volviéndose a guardar la fracción.

⊕ Los individuos quedarán guardados en un arreglo y en un archivo; están representados por números. Los renglones del arreglo representan el tiempo y las columnas los individuos.

MUESTREO

Una vez creada la población se muestrea de ella. El procedimiento Muestras pregunta cuántas muestras (Meyer et al. (1986) usan 1000) y el tamaño de muestra deseado (Meyer et al. (1986) usan 10). Se escoge un individuo de la población total y se copia en el archivo o en el arreglo de las muestras. Este proceso se repite hasta llegar al tamaño de muestra y al número de muestras.

Las unidades ABSOLUTA, BUU y NAVAJA leen estas muestras y a partir de ellas calcularán la lx y mx para todas las t . Leyendo por renglón se lleva el número de cuántos individuos están vivos, al dividirse entre el número de individuos de la muestra se obtiene la

lx. Para calcular la mx se suman los hijos de todos los individuos y se dividen entre el número de vivos.

Si la muestra es de 10 individuos se tendrán los siguientes resultados:

individuos										t	#vivos	lx	#hijos	mx	
1	2	3	4	5	6	7	8	9	10			#vivos		#hijos	
											#total			#vivos	
0	0	0	99	0	0	0	0	0	0	1	9	.9	0	0	
99	0	0	99	0	0	0	0	0	0	2	8	.8	0	0	
99	0	99	99	0	0	0	0	0	0	3	7	.7	0	0	
99	1	99	99	1	1	1	1	1	1	4	6	.6	6	1	
99	2	99	99	2	99	2	99	2	2	5	5	.5	10	2	
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	
99	99	99	99	99	99	99	99	99	99	20	0	0	0	0	

De esta forma se obtienen las lx y las mx para los veinte intervalos de tiempo. Luego, para calcular la lx y la mx de las cuatro categorías, se toma la lx del límite superior de la categoría y para la mx se suman las mx comprendidas entre los límites de la categoría. Aquí no se necesita dividir la mx entre los individuos sobrevivientes, puesto que la mx ya está promediada entre ellos. Si existe algún individuo vivo dentro de la categoría tendrá la fecundidad que indica la mx para la t dada. Si sobrevive y pasa a la siguiente categoría su mx es la suma de hijos que tuvo en cada t (mx para cada t) comprendida dentro de la categoría. Los límites de las categorías son en t=5, t=10, t=15 y t=20.

t	lx	mx	lx	mx
1	.9	0		
2	.8	0		
3	.7	0		
4	.6	0		
5	.5	1	.5	1
6	.4	1		
7	.3	1		
8	.2	1		
9	.15	2		
10	.1	2	.1	7
11	.05	2		
12	0	0		
13	0	0		
14	0	0		
15	0	0	0	0
16	0	0		
17	0	0		
18	0	0		
19	0	0		
20	0	02	0	0

Desde aquí la manera en que trabaja cada unidad difiere ligeramente. ABSOLUTA manda directamente los datos de las 4 lx y 4 mx hacia la unidad lmx para que construya la matriz y calcule λ . BUU aplica el bootstrap y NAVAJA el jackknife. La forma en que calculan las lx y las mx de sus muestras las unidades ABSOLUTA, BUU y NAVAJA es igual a la descrita antes; sólo varía en el número de muestras que mandan para calcular λ .

ABSOLUTA.TPU

ABSOLUTA toma las muestras creadas por la unidad POBYMUE. Agrupa en categorías y calcula las l_x y m_x para mandarlas a la unidad LXXM. Saca el promedio de las lambdas obtenidas para varias muestras de la misma población y del mismo tamaño de muestra.

BUU.TPU

BUU también toma las muestras creadas por POBYMUE. De la muestra original agrupa en categorías y calcula la l_x y m_x para mandarla a LXXM, pero además aplicará el bootstrap agrupando en categorías y calculando las l_x y m_x tantas veces como se fije el tamaño de muestra del bootstrap. En este caso es 1000. Por lo que cada muestra leída generará 1001 matrices y subsecuentemente 1001 lambdas.

Una vez que los datos de l_x y m_x pasen a la unidad LXXM y de aquí a la unidad LAMDARCH, la unidad BUU llamará a la unidad DESVARB para acumular resultados parciales.

NAVAJA.TPU

NAVAJA hace lo mismo que BUU, con la diferencia de que aplica el jackknife. También manda los datos de l_x y m_x a la unidad LXXM y de aquí pasan a la unidad LAMDARCH. Y después llama a la unidad DESVARJ para acumular resultados parciales. La diferencia importante es que esta unidad generará tantas matrices y lambdas como el tamaño de muestra más uno.

LXXM.TPU

Aquí se toman los datos de las l_x y m_x calculados en las unidades previas y se construye la matriz de transiciones. Existe la posibilidad de calcular la matriz para poblaciones continuas y discretas, para censos prerreproductivos o postreproductivos y de tomar en cuenta si la mortalidad neonatal es alta o no. Ya construida la matriz se le pasa como parámetro a la unidad LAMDARCH.

LAMDARCH.TPU

Toma la matriz de transiciones y calcula λ , la matriz de elasticidades y la matriz de sensibilidades. Guarda las lambdas en un arreglo global para luego usarse en las unidades DESVARB y DESVARJ.

La λ se calcula con el método de la potencia. (Álvarez-Buylla y Slatkin, en prep.).

DESVARB.TPU

Lee el arreglo que contiene todas las lambdas obtenidas de aplicar el bootstrap a una muestra y con ellas calcula λ^p , λ^r , λ^b , σ , σ^2 y el sesgo. También puede acumular estos valores para después promediarlos con los de otras muestras de la misma población y del mismo tamaño de muestra (si se llama desde el procedimiento Muestras de POBYMUE).

DESVARJ.TPU

Lee el arreglo que contiene todas las lambdas obtenidas de aplicar el jackknife a una muestra y con ellas calcula λ^p , λ^T , λ' , σ , σ' y el sesgo. También puede acumular estos valores para después promediarlos con los de otras muestras de la misma población y del mismo tamaño de muestra (si se llama desde el procedimiento Muestras de POBYMUE).

INFORMA.TPU

Genera una ventana de información permanente en la que se señala con cuál población se está trabajando, qué método se usa, el tamaño de muestra de la muestra creada o leída, el número de muestras que existen de esa misma población, y el número de bootstrap o de jackknife que se lleva a cabo.

VARGLOB.TPU

Esta unidad contiene casi todas las declaraciones de las variables y procedimientos que se utilizan en varias unidades. No contiene todas las variables utilizadas en más de una unidad debido a que en ciertos casos se ahorra tamaño en los datos.

KSSI.TPU

Realiza las pruebas de significancia entre medias y varianzas y aplica la prueba de Kolmogorov-Smirnov para probar la normalidad de una distribución aleatoria.

ADORNO.TPU

Proporciona el manejo de pantalla para el menú.

GRAFI.TPU

Permite graficar las funciones de sobrevivencia y de fecundidad.

DATOS TÉCNICOS

Memoria necesaria	90kp
Tamaño del código	70kp
Tamaño de los datos	38kp

APÉNDICE B

DIAGRAMA DE FLUJO

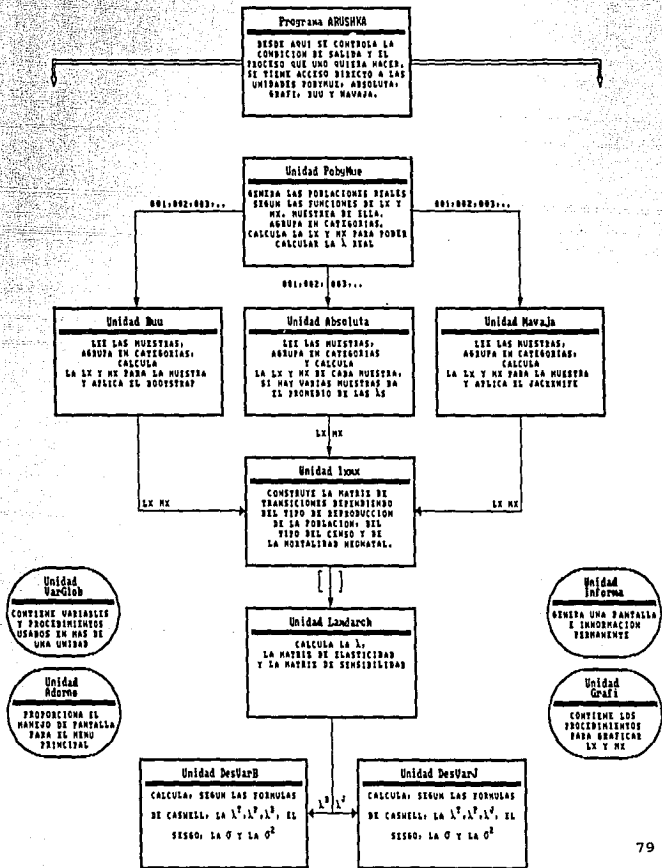


Figura 21

ESTA TESIS NO DEBE SALIR DE LA BIBLIOTECA

PROGRAM ARUSHKA;

{SM 60000, 0, 655360}

{SN+}

{

Este programa genera una población ideal de mil individuos de acuerdo a diferentes combinaciones de seis funciones de sobrevivencia y cuatro funciones de fecundidad. Calcula la l_x y la m_x de la población ideal.

Puede muestrear de 1 a 1000 veces y de 1 a 200 individuos de aquélla, calculando la l_x y la m_x para cada muestra.

Agrupar en cuatro categorías a los individuos según su edad. Construye la matriz de transiciones para poblaciones que se reproducen a continuamente o a pulsos (censo prerreproductivo o postreproductivo) tomando en cuenta su mortalidad neonatal.

Aplica los métodos de jackknife y bootstrap a las muestras y calcula la l_x y la m_x .

A partir de la matriz de transiciones calcula la λ , la matriz de sensibilidades y la de elasticidades de la población ideal (total) o de cualquier muestra.

Las características de las poblaciones son:

- La longevidad de ningún individuo sobrepasa $t=20$
- Se empiezan a reproducir a partir de $t=5$ para cualquier función de fecundidad.
- Una vez que un individuo empieza a reproducirse lo hará hasta que muera, según su función de fecundidad.
- El período de gestación es menor a la unidad de tiempo, por lo que un individuo se reproducirá, mientras viva, en cada unidad de tiempo.

}

USES

Crt,
 Informa, PobyMue, Grafi,
 Dos, Adorno,
 VarGlob,
 KSSi;

VAR

Respuesta :Char;
 Orden :String [100];

{

Procedure Instrucciones;

BEGIN

```

Clrscr;
Window (5,1,80,25);
TextColor (Black+Blink);
Writeln ('                INFORMACIÓN                ');
Writeln ('                ');
TextColor (Yellow);
Writeln ('- Genera una población ideal a partir de una
                función de                ');

```

```

Writeln ('      sobrevivencia y una función de fecundidad.  ');
Writeln ('- Grafica ambas funciones.  ');
Writeln ('- Aplica la prueba de Kolmogorov-Smirnof  ');
Writeln ('- Permite muestrear entre 1 y 200 individuos de aquélla  ');
Writeln ('      y crear hasta 1000 muestras.  ');
Writeln ('- Calcula la lx y la mx para cualquier muestra y  ');
Writeln ('      también para la población total.  ');
Writeln ('- Aplica el bootstrap y el jackknife.  ');
Writeln ('- Construye la matriz de transiciones para poblaciones con  ');
Writeln ('      reproducción continua o a pulsos: (censo  ');
Writeln ('      prerreproductivo o postreproductivo), tomando  ');
Writeln ('      en cuenta su mortalidad neonatal.  ');
Writeln ('- Calcula lambda, la matriz de sensibilidades y  ');
Writeln ('      la de elasticidades a partir de la matriz de transiciones.  ');
Writeln ('- Realiza prueba de significancia para medias y varianzas (95%);');

```

```

Readln;
End;

```

```

-----
Procedure LeeLx0Mx;

```

```

VAR
  I      :byte;
  rl     :char;
  nn     :string[12];
  arch   :text;
  numero:MorFec;
Begin
  I:=0;
  write (' ¿lx o mx? M/F');
  repeat
    rl:=readkey;
    rl:=upcase(rl);
  until (rl='M') or (rl='F');
  write (' ¿Nombre del archivo? ');
  readln (nn);
  Assign (arch,nn);
  Reset (arch);
  while not eof(arch) do begin
    inc (i);
    readln (arch,numero[i]);
  end;
  Pintas (numero,rl);
End;

```

```

-----
Procedure MS_DOS;

```

```

Begin
  Repeat
    writeln ('Orden de DOS: ');
  
```

```

readln (orden);
If orden <> '' then begin
  SwapVectors;
  Exec ('c:\DOS\GetEnv('ComSpec'), '/C'+ Orden);
  SwapVectors;
  If DosError <> 0 then writeln ('No puedo correr el
                                COMMAND.COM');
  sonido (4,2,50,20);
end;
Until Orden='';
End;

```

```

{-----}
Procedure Eleccion;

```

```

VAR
  Escogi:Char;
Begin
  clrscr;
  Presentacion;
  clrscr;
  Window (1,9,80,25);
  Case y of
    1:Instrucciones;
    2:GPoblacion;
    3:Muestras;
    4:MS DOS;
    5:Leelxomx;
    6:Kolmogorov_Smirnov95;
  End;
  ClrScr;
  GotoXY (27,1);
  TextBackground (Green);
  TextColor (Black);
  Write (' ¿Quieres hacer algo más? ');
  Repeat
    Sonido (4,5,50,10);
    Respuesta:=readkey;
    Respuesta:=Uppcase (Respuesta);
  Until (Respuesta='S') or (Respuesta='N');
End;

```

```

{*****}

```

```

BEGIN
  NombreP:='No hay';
  TamMuestraReal:=0;
  NumeroDeMuestras:=0;
  TextBackground (Red);
  TextColor (Yellow);
  Repeat
    Eleccion;
  Until Respuesta='N';

```

END.

UNIT PobyMue;

```
{ $M 60000, 0, 655360 }
{ $N+ }
{ La unidad se compone de dos procedimientos: GPoblación y
Muestras. El primero genera la población y el otro muestrea de
ella subpoblaciones.
}
```

Interface

```
USES Crt, Informa, VarGlob, Grafi, lxm,
Absoluta, Buu, NavaJa, DesVarJ, DesVarB;
Procedure GPoblacion;
Procedure Muestras;
```

Implementation

CONST

```
NumInd=1000; { # total de individuos en una }
{ población. }
}
```

VAR

```
c1 :integer;
Cohorte { Aquí está la vida, a través }
{ del tiempo, de todos los }
{ individuos. }
: Array [1..Tiempo, 1..NumInd] of byte;
```

```
-----
Procedure TipoDeReproduccion;
```

```
{ Se establece el tipo de reproducción de la población y su
mortalidad neonatal.
}
```

Begin

```
ClrScr;
Writeln ('El tipo de reproducción de la población es: ');
Writeln ('(C)ontinua (D)iscreta Discret(a) ');
Write ('censo postreproductivo censo prerreproductivo ');
Repeat
Textcolor (Green+Blink);
Poblacion:=readkey;
Poblacion:=upcase(Poblacion);
TextColor (Magenta)
Until (Poblacion='C') or (Poblacion='D') or (Poblacion='A');
clrscr;
Write ('¿Tu población tiene alta la mortalidad neonatal?
(S/N) ');
Repeat
```



```

Textcolor (Green+Blink);
Res:=readkey;
Res:=Ucase (Res);
Textcolor (Magenta)
Until (Res='N') or (Res='S');
End;

```

```

-----

```

PROCEDURE GPOBLACION;

```

{ Crea a los individuos de la población total (ideal). Los
sigue como una cohorte. La población tiene 1000 individuos, los
que crecen se reproducen y mueren en el transcurso de su vida.
Cada organismo se representa con un número mientras está vivo. El
número indica cuántos hijos tiene en un intervalo de tiempo. Si
el individuo está vivo, pero no tiene descendencia en ese tiempo,
se representa como 0. Al morir se representará como 99.

```

Entrada

=====

- * Función de sobrevivencia
- * Función de fecundidad

Salida

=====

- * Archivo de texto con las lx y las mx reales
- * Archivo de bytes con la población total (ideal)

```

}
VAR
  Tipo1,
  Tipo2,
  Hijos      :Byte;
             { Total de hijos de todos los }
             { individuos para un intervalo }
             { de tiempo.                  }
  f,
  mx         :Word;
             { Ver procedimiento Mortalidad. }
  Dlx        :Real;
             { Valor de mx según la función. }
  Archivo,
  Datoslx,
  Datosmx    :Extended;
             { idem.                          }
  Legion     :Text;
             { Guarda las lx y mx de la pt.  }
  Nombre     :File of byte;
             { Guarda la población total,   }
             { o sea, el arreglo Cohorte.   }
  x,
  Nombre     :String [12];
             { lx y mx de la pob total.     }
             { Tiempo transcurrido. Se podría }

```



```

writeln ('(5)                                Tipo II                                ');
writeln ('(6)                                Tipo III                               ');
write ('(7) datos propios                    ----> ');
Repeat
  Readln (Tipo1);
  If (Tipo1>7) or (Tipo1<1) then
    write ('Ese valor no existe. Da un valor válido
           (1-4) ----> ');
Until (Tipo1<=7) and (Tipo1>=1);
writeln;
writeln ('¿Con qué tipo de función de fecundidad quieres
        trabajar?');
writeln;
writeln ('(1)  $mx=(0.8x - 4)/10$                                 ');
writeln ('(2)  $mx=(4 * \sqrt{(x-5) + 3})/10$                         ');
writeln ('(3)  $mx=(50 - \sqrt{(2500*1-(x^2/400))})/10$                 ');
writeln ('(4)  $mx=(50 * e^{-(x-15)^2/20})/10$                     ');
write ('(5) datos propios                    ----> ');
Repeat
  Readln (Tipo2);
  If (Tipo2>5) or (Tipo2<1) then
    write ('Ese valor no existe. Da un valor válido
           (1-4) ----> ');
Until (Tipo2<=5) and (Tipo2>=1);
ClrScr;
Window (1,1,80,25);
GotoXY (26,7);
Write ('Leo las lx y mx...');
Window (1,9,80,25);
If Tipo1<7 then begin
  Assign (DatosLx,'LX.'+chr(Tipo1+48));
  Reset (DatosLx);
  For q:=1 to Tiempo do
    Readln (DatosLx,Flx [q]);
  end
else begin
  writeln ('Introduce los datos de lx:');
  For q:=1 to Tiempo do begin
    writeln (q,' ');
    Readln (Flx [q]);
  end;
end;
Pintalas (Flx,'M');
If Tipo2<5 then begin
  Assign (DatosMx,'MX.'+chr(Tipo2+48));
  Reset (DatosMx);
  For q:=1 to Tiempo do
    Readln (DatosMx,Fmx [q]);
  end
else begin
  writeln ('Introduce los datos de mx:');
  For q:=1 to Tiempo do begin

```



```

      If NumVivos > 0 Then mxes [DiasTrans]:=Hijos/NumVivos
      Else mxes [DiasTrans]:=0;
              { Evita una división entre 0 }
End;

(!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!)
      Procedure Archivos;

      { Crea los archivos de salida que contienen a las poblaciones
      totales. El nombre de los archivos será siempre PT, y la
      extensión estará compuesta por el número de la función de
      sobrevivencia seguido por el número de la función de fecundidad.
      Si se escogió la función de sobrevivencia 3 y la de fecundidad es
      2, el archivo se llamará PT.3^2
      }

Begin
  ClrScr;
  writeln ('Las lx y mx de la población total se guardarán en
                                       Ab000');
  Nombre:='Ab000';
  Assign (Archivo,Nombre);
  writeln ('La población total se guardará en PT.',^,Tip0,Tip0);
  NombreP:='PT.'+chr(Tip0+48)+chr(94)+chr(Tip0+48);
  Pantalla (0);
  Assign (Legion,NombreP);
  Rewrite (Legion);
  Rewrite (Archivo);
End;
(!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!)
      Procedure Letrero;

      { Avisar que ya se creó la población.
      }

VAR
  Y:Byte;
Begin
  ClrScr;
  For y:=1 to 4 do
    Begin
      GotoXY (10,14);
      TextBackground (Cyan);
      TextColor (Yellow+Blink);
      Writeln ('Población ideal creada y guardada con sus lx
                                              y mx');
      Sonido (4,7,30,10);
    End;
  TextColor (Yellow);
End;

```



```

mx4.
)
VAR
  Acumula      :Real;
  Hijos        :Integer;
  I5, J4,
  I4, J4,
  Vivos        :Byte;
  Lx20, Mx20   :Array [1..20] of real;
Begin
  Window (1,1,80,25);
  GotoXY (26,6);
  Window (1,9,80,25);
  For I4:=1 to Tiempo do
  Begin
    Vivos:=0;
    Hijos:=0;
    For J4:=1 to TamMuestraReal do
    Begin
      If MuestraO [J4,I4] <> 99 then
      begin
        inc (vivos);
        hijos:=hijos+MuestraO [J4,I4];
      end;
    End;
    Lx20 [I4]:=Vivos/TamMuestraReal;
    If vivos > 0 then Mx20 [I4]:=Hijos/Vivos
    else Mx20 [I4]:=0;
  End;
  For I5:=1 to 4 do
  Lx4 [I5]:=Lx20 [5*I5];
  { Escribe las lx en otro arreglo }
  { agrupadas en 4 categorias }
  Acumula:=0;
  For I5:=1 to 20 do
  Begin
    Acumula:=Acumula+mx20 [I5];
    { Escribe las lx en otro arreglo }
    If (I5=5) or
    { agrupadas en 4 categorias }
    (I5=10) or
    (I5=15) or
    (I5=20) then
    begin
      mx4 [I5 DIV 5]:=Acumula;
      Acumula:=0;
    end;
  End;

```

```

End;
{-----}
      Procedure PreparaMillLambdas;
(      Prepara el archivo con su encabezado para guardar las 1000
lambdas obtenidas directamente.
)
Begin
  assign (MillLambdasDE,
         'lambdasd.'+NombreP[4]+NombreP[5]+NombreP[6]);
  ($I-) append (MillLambdasDE); {$I+}
  Hay:=(IOresult=0);
  If not Hay then
    begin
      rewrite (MillLambdasDE);
      writeln (MillLambdasDE,'Lambda          tamaño de
              muestra');
      writeln
      (MillLambdasDE,'-----');
    end;
End;

```

```

{-----}
      Procedure Escribe;

```

```

(      De las z muestras que leyó la unida se calculó una lambda para
cada una, aquí se da el promedio de esas lambdas, además de la
lambda real de la población de la que se muestrearon.
)

```

```

Begin
  suma:=suma/NumeroDeMuestras;
  writeln ('Lambda promedio= ',suma:6:4,' Pob= ',NombreP,' #=
          ',NumeroDeMuestras,' Tam Muestra= ',TamMuestraReal);
End;

```

```

{*****}

```

```

BEGIN
  PreparaMillLambdas;
  Calc;
  HazMatriz (lx4,mx4);
  Suma:=Suma+Lambda;
  Close (MillLambdasDE);
END;
END.

```

```

      UNIT Buu;
(      Aquí se leen las muestras tomadas de la población, se
calcula sus lx y mx y se aplica el método del bootstrap.

```

Llama después a la unidad DESVARB para calcular las lambdas, el sesgo, desviación, etc.

Entrada =====	Salida =====
* Tamaño de la muestra que leerá	* 1x y mx obtenidas directamente de la muestra
* Las vidas de 1 a 200 individuos (muestra) del arreglo MuestraO	* 1x y mx obtenidas de usar el bootstrap (1000)

}

Interface

Uses Crt, VarGlob, Informa, LxMx, DesVarB;

Procedure Agujeta;

Implementation

Procedure Agujeta;

VAR

MuestraR { Remuestra del Arreglo MuestraO }
: Array [1..TamMuestraMax,1..Tiempo] of byte;

Procedure But;

{ Del arreglo que contiene la muestra toma individuos al azar y construye otro arreglo de las mismas dimensiones, pero con diferente composición.

La primera vez sólo copia la muestra tal cual en el arreglo MuestraR para más tarde calcular la lambda de la muestra. Produce de una muestra 1001 muestras. La 1ª es la muestra original las otras 1000 son muestras hechas al azar. El número de bootstraps es 1000.

}

VAR

N, { Lleva la cuenta del # del renglón que sigue }
I, { Va de 1 a tiempo, la vida de 1 individuo }
C :Byte;
{ Va de 1 al # de individuos de la muestra }
Suerte :LongInt;
{ Individuo muestreado al azar }

Begin

If I1000=1 then For C:=1 to TamMuestraReal do
For I:=1 to Tiempo do
MuestraR [C,I]:=MuestraO [C,I]
else begin
Randomize;
N:=0;

```

For c:=1 to TamMuestraReal do
  Begin
    Inc (N);
    Repeat
      Suerte:=Random (TamMuestraReal+1);
    Until suerte>0;
    For I:=1 to Tiempo do
      MuestraR [N,I]:=MuestraO
      [suerte,I];
    End;
  End;
End;
-----
Procedure Calc;
(
  Calcula las lx y mx 1001 veces de los diferentes arreglos
  guardados en MuestraR y las guarda en dos arreglos unidimensionales
  lx20 y mx20. Luego calcula las lx y mx para las 4 categorías y las
  guarda en lx4 y mx4. Estos dos vectores se le mandan a la unidad
  lxxm para que construya la matriz de transiciones.
)
VAR
  Acumula      :Real;
  Hijos        :Integer;
  I4,J4,
  Vivos        :Byte;
  lx4,mx4     :Vectores;
  lx20,mx20   :Array [1..20] of real;
Begin
  For J4:=1 to tiempo do
    Begin
      Vivos:=0;
      Hijos:=0;
      For I4:=1 to TamMuestraReal do
        Begin
          If MuestraR [I4,J4] <> 99 then
            Begin
              inc (vivos);
              hijos:=hijos+MuestraR [I4,J4];
            End;
          End;
        Lx20 [J4]:=Vivos/TamMuestraReal;
        If vivos > 0 then Mx20 [J4]:=hijos/vivos
        else Mx20 [J4]:=0;
      End;
    ( Se agrupan las lx y las mx en categorías )
    For I4:=1 to 4 do
      lx4 [I4]:=lx20 [I4*5];

```



```

Acumula:=0;
For I4:=1 to 20 do
  Begin
    Acumula:=Acumula+mx20 [I4];
    If (I4=5) or
       (I4=10) or
       (I4=15) or
       (I4=20) then
      begin
        mx4 [I4:DIV 5]:=Acumula;
        Acumula:=0;
      end;
    End;
  End;
  HazMatriz (1x4,mx4);
  { Llama al procedimiento que hace }
  { matriz en la unidad LXXM }
End;
(-----)
BEGIN
  For I1000:=1 to NumeroDeBuu+1 do begin
    But;
    Calc;
    Pantalla (I1000);
  end;
  DVB (true);
  { Almacena datos en la unidad DESVARB }
END;
END.

```

UNIT Navaja;

```

{$M 60000, 0, 655360}
{
  Lee datos de un archivo de bytes cuyo contenido es una muestra
  de individuos (1-200) a través del tiempo (1-20). Calcula las lx y
  las mx de esa muestra. Aplica el jackknife a la muestra y calcula
  las lx y mx.
  Llama después al procedimiento DVJ de la unidad DESVARJ para
  calcular las lambdas, desviación, sesgo, etc.

```

Entrada =====	Salida =====
* Tamaño de la muestra que leerá.	* lx y mx obtenidas directamente de la muestra.
* Las vidas de 1 a 200 individuos de un archivo de byte. (PobyMue)	* lx y mx calculadas después de usar el

```

jackknife.
)
Interface      USES Crt, Informa, VarGlob, Lxmx, DesVarJ;
               Procedure Jack;
Implementation  Procedure Jack;

VAR
  Vivos,lx,mx      :Real;
  Hijos,Cuenta     :Integer;
  IndRebanado      :Byte;
                  { Excluye a un ind. cada vez que }
                  { calcula el jackknife }

  lxJ,mxJ          :Array
                  [1..TamMuestraMax+1,1..Tiempo] of real;
                  { lx y mx de las muestra y de }
                  { los n jackknife }

  -----
  Procedure CalculaLXMX;
  { Calcula las lx y mx a partir de la muestra de individuos y
  también las lx y mx usando el jackknife.
  }
VAR
  I2,J2 : Byte;

Begin
  For J2:=1 to Tiempo do
  Begin
    Hijos:=0;
    Vivos:=0;
    For I2:=1 to TamMuestraReal do
    Begin
      If IndRebanado<>I2 then
        { Cuando se usa el jackknife }
        { este if elimina de uno a uno }
        { la lectura de los individuos }
        { de la muestra (no entra) }

      If Muestra0 [I2,J2]<> 99 then
        { No entra si el individuo está }
        { muerto: 99 }
        Begin
          Vivos :=Vivos+1;
          Hijos :=Muestra0 [I2,J2]
                + Hijos;

```

```

End;
If IndRebanado=0 then begin
  { Calcula las lx y mx a partir }
  { de todos los individuos y }
  { escribe los resultados en un }
  { archivo de texto }

  lx:=Vivos/TamMuestraReal;
  If Vivos>0 then
    mx:=Hijos/Vivos
  else mx:=0;

  lxJ [1,J2]:=lx;
  mxJ [1,J2]:=mx;
end

else begin
  { Entra cuando va a calcular las }
  { lx y mx usando el jackknife y }
  { guarda los datos en los arre- }
  { glos lxj y mxj. En el procedi- }
  { miento Jacky se usarán para }
  { calcular las lx y mx }

  lx:=Vivos/(TamMuestraReal-1);
  If Vivos>0 then
    mx:=Hijos/Vivos
  else mx:=0;

  lxj [IndRebanado+1,J2]:=lx;
  mxj [IndRebanado+1,J2]:=mx;
end;
End;
End;
}-----}
Procedure LlamaHazMatriz;
{ A partir de las n lx y mx obtenidas después de usar el
jackknife y que están guardadas en dos arreglos, lxJ y mxJ, calcula
las lx y mx para las 4 categorías y guarda sus valores en los
vectores lx4 y mx4. Los vectores se le mandan a la unidad LXXM.TPU
para que haga la matriz de transiciones
}
VAR
  I4           :Byte;
  Acumula     :Real;
  lx4,mx4     :Vectores;

Begin
  For Renqlon:=1 to TamMuestraReal+1 do
    Begin
      For I4:=1 to 4 do

```

```

    lx4 [I4]:=lxJ [renglon,I4*5];
Acumula:=0;
For I4:=1 to 20 do
Begin
    Acumula:=Acumula+mxJ [renglon,I4];
    If (I4=5) or
       (I4=10) or
       (I4=15) or
       (I4=20) then begin
        mx4 [I4 DIV 5]:=Acumula;
        Acumula:=0;
        end;
    End;
HazMatriz (lx4,mx4);           { Llama a la unidad LXXM }
Pantalla (Renglon);          { Actualiza la pantalla de información }
End;
{*****}
BEGIN
    { Este procedimiento está en la }
    { unidad DESVARJ.TPU           }
    IndRebanado:=0;
    CalculaLxMx;                { Calcula las lx y mx para toda }
                                { muestra                          }
    Repeat                       { Repite el bloque hasta que     }
                                { todos los individuos hayan   }
                                { sido excluidos 1 vez de los     }
                                { cálculos                          }
    Inc (IndRebanado);           { Excluye de los cálculos 1 ind.}
    CalculaLxMX;                 { Calcula las lx y mx para n-1 }
                                { individuos                      }
    Until IndRebanado=TamMuestraReal;
    LlamaHazMatriz;              { Obtiene las lx y mx del Jack }
    DVJ (true);                  { Almacena datos en DESVARJ   }
END;
END.

```

UNIT lxmx;

```

{$N+}
{

```

Esta unidad construye la matriz de transiciones a partir del archivo que contiene las lx y las mx, de cualquier población. Las entradas de la matriz se pueden calcular para poblaciones de reproducción continua y a pulsos (censo prerreproductivo y censo postreproductivo) y con alta o baja mortalidad neonatal. Al terminar llama a LAMDARCH.TPU.

Entrada
=====

Salida
=====

* Vectores con lx y mx de las unidades
PobyMue, Absoluta, Buu y Navaja

* matriz de transiciones

$$\begin{bmatrix} F11 & F12 & F13 & F14 \\ P21 & \dots & \dots & \dots \\ \dots & P32 & \dots & \dots \\ \dots & \dots & P43 & \dots \end{bmatrix}$$

```
)  
Interface  
  Uses Crt,Lamdarch,VarGlob;  
  TYPE  
    Vectores=Array [1..4] of real;  
  VAR  
    MatrizTran:Matrizz;  
    Procedure HazMatriz (lx4,mx4:Vectores);
```

```
Implementation  
  Procedure HazMatriz (lx4,mx4:Vectores);
```

```
CONST  
  Max=4;           ( Dimensión máxima de la matriz   )  
  Total=4;        ( producida                         )
```

```
VAR  
  Leer           :Char;  
                { Número de datos de lx y de mx   }  
                { Vectores con las lx y las mx     }  
                { leídas y con las probabilidades  }  
                { de transición y las fecundidades }  
                { para cada tipo de población:    }  
                { C -- Población con rep continua }  
                { D0 -- Pob. con rep. discontinua }  
                { D1 -- Pob. con rep. discontinua }  
                { censo prerreproductivo         }  
                { censo postreproductivo         }  
                { D1 -- Pob. con rep. discontinua }  
                { censo postreproductivo         }
```

```
  :Array [1..Max] of Real;
```

```
{-----}  
  Procedure Limpia;
```

```
{ Se llenan los tres vectores con ceros para evitar errores de  
cálculo.  
}
```

```
VAR  
  I1:Byte;
```

```

Begin
  For I1:=1 to Max do
    Begin
      PiC [I1]:=0;
      PiD0 [I1]:=0;
      PiD1 [I1]:=0;
    End;
  End;

{-----}
                Procedure P1;
{   Se calculan las probabilidades de transición para cualquier
    población: continua o discontinua (censo prerreproductivo o
    postreproductivo)
}

VAR
  I2:Byte;
Begin
  For I2:=1 to Total do
    { Cada if tiene un else para }
    { evitar divisiones entre 0 }
    { y el último es para indicar }
    { que el resultado es = ∞ }
    { 0 = evita que sea dividido entre 0 }
    { r = evita un desbordamiento + 0 - }
    Begin
      { Reproducción continua }
      Case I2 of
        (r-) 1:PiC [I2]:=( 1x4 [I2]+1x4 [I2+1]) / ( 1
                                                    +1x4 [I2]);
          2,3:Begin
            If 1x4 [I2-1]+1x4 [I2]=0 then
              PiC [I2]:=( 1x4 [I2]+1x4 [I2+1]) / 0.0001
            else PiC [I2]:=( 1x4 [I2]+1x4 [I2+1]) / (1x4
                                                    [I2-1]+1x4 [I2]);
            End;
          (r+) 4:Begin
            If 1x4 [I2-1]+1x4 [I2]=0 then
              PiC [I2]:=( 1x4 [I2]+ 0 ) / 0.0001
            else PiC [I2]:=( 1x4 [I2]+ 0 ) / (1x4
                                                    [I2-1]+1x4 [I2]);
            End;
          End;
      End;
    End;
  End;

{ Reproducción discreta. Censo prerreproductivo }
(r-) IF I2=1 then PiD0 [I2]:=1x4 [I2] / 1
(0)  else if 1x4 [I2-1]=0 then PiD0 [I2]:=1x4 [I2] /
                                           0.0001
                                           else PiD0 [I2]:=1x4 [I2] / 1x4

```

```

                                                                    [I2-1];
{ Reproducción discreta. Censo postreproductivo }
{0}      IF lx4 [I2]=0 then P1D1 [I2]:=236
{r+}     else if I2=4 then P1D1 [I2]:=0/lx4 [I2]
                else P1D1 [I2]:=lx4 [I2+1]/lx4 [I2];
      End;
End;
{-----}
                          Procedure Fi;
{ Se calculan las fecundidades para cualquier tipo de población.
}
VAR
  L5 :Real;
  I3 :Byte;
Begin
{ Dependiendo de la mortalidad }
{ neonatal de la población se }
{ calcula de diferente forma }
{ la lx en t=.5. }
  IF (Res='S') then L5:=Sqrt ((1 * lx4 [1]))
                else L5:=(1 + lx4 [1])/2;
  For I3:=1 to Total do
    Begin
{ Reproducción continua. }
      If I3=Total then
        Fic [I3]:=L5 * ((mx4 [I3]+Pic [I3] * 0 )/2)
      else Fic [I3]:=L5 * ((mx4 [I3]+Pic [I3] * mx4
                                [I3+1])/2);
{ Reproducción discreta. Censo prerreproductivo }
      FID0 [I3]:=PID0 [I3] * mx4 [I3];
{ Reproducción discreta. Censo postreproductivo }
      FID1 [I3]:=lx4 [1] * mx4 [I3];
    End;
  End;
{-----}
                          Procedure Dibujamatriz;
{ Se construye la matriz de transiciones a partir de los datos
de un vector de probabilidades de transición y de un vector de
fecundidad.
}
VAR
  J1,I5:Byte;

```

```

Begin
  Writeln;
  Writeln ('La matriz es:');
  Writeln;
  For I5:=1 to Total do
    For J1:=1 to Total do
      MatrizTran [I5,J1]:=0;
  IF Poblacion='C' then
    { Si la reproducción es }
    { continua }
    Begin
      For I5:=1 to Total do
        MatrizTran [1,I5]:=F1C [I5];
      For I5:=1 to Total-1 do
        MatrizTran [I5+1,I5]:=P1C [I5];
      End;
  IF Poblacion='D' then
    { Si la reproducción es }
    { discreta (censo prerrep.) }
    Begin
      For I5:=1 to Total do
        MatrizTran [1,I5]:=F1D0 [I5];
      For I5:=1 to Total-1 do
        MatrizTran [I5+1,I5]:=P1D0 [I5];
      End;
  IF Poblacion='A' then
    { Si la reproducción es }
    { discreta (censo postrep.) }
    Begin
      For I5:=1 to Total do
        MatrizTran [1,I5]:=F1D1 [I5];
      For I5:=1 to Total-1 do
        MatrizTran [I5+1,I5]:=P1D1 [I5];
      End;

  For I5:=1 to Total do
    IF MatrizTran [1,I5]=0 then MatrizTran [1,I5]:=0.0001;
  For I5:=1 to Total-1 do
    IF MatrizTran [I5+1,I5]=0 then MatrizTran [I5+1,I5]:=0.0001;

    { Esta parte examina las fecun- }
    { didades y las probabilidades }
    { de transición de la matriz }
    { para evitar que alguna sea 0. }
    { Si alguna es 0 se cambia por }
    { .0001. }
  End;
  {*****-----*****}
BEGIN
  Limpia;
  Pi;
  Fi;
  DibujaMatriz;
  Lambda (MatrizTran);
END;
END.

```


UNIT Lamdarch;

Por: Elena Álvarez-Buylla

Álvarez-Buylla, E. y Slatkin, M. (en prep)

UNIT DesVarB;

{SN+}

```
{ Calcula, según la fórmula de Caswell, la lambda promedio, la
lambda bootstrap el sesgo, la varianza, desviación típica, de las
lambdas obtenidas al muestrear una población
}
```

Interface

```
Uses Crt,VarGlob;
Procedure InicializaB;
Procedure DVB (No:Boolean);
```

Implementation

CONST

```
NombreDelArchivoConLambdasBuu='Bfin';
{ Donde se guardan los resultados}
```

VAR

```
unaLambda,
LambdaT,          ALambdaT,      { Acumuladores de las variables }
LambdaP,          ALambdaP,      { de la izquierda          }
LambdaB,          ALambdaB,
Sesgo,            ASesgo,
Varianza,         AVarianza,
Desviacion,       ADesviacion,
SumatoriaDeDif
Salida
Indicador
:Real;
:Text;
:Boolean;
```

```
{-----}
Procedure InicializaB;
```

```
{ Inicializa las variables que acumulan la suma de otras,
revisa si existe el archivo donde se guardan, si no existe lo crea
con su encabezado.
```

```
Este procedimiento se llama desde la unidad BUU.TPU
```

```
}
```

Begin

108

```

ALambdaT:=0;
ALambdaP:=0;
ALambdaB:=0;
Asego:=0;
AVarianza:=0;
ADesviacion:=0;
Assign (salida, NombreDelArchivoConLambdasBuu);
{$I-} Append (salida); {$I+}
Indicador:=(IOresult=0);
If not indicador then
begin
rewrite (salida);
writeln (salida, 'Poblac      Tamaño      LambdaT
      LambdaP      Sesgo      LamdaB       $\sigma^2$        $\sigma$       #');
writeln
(salida, '-----');
end;
End;
{-----}
Procedure Calcula;
{ Efectúa todos los cálculos para obtener las lambdas y demás
variables.
}
VAR
I:Integer;
Begin
LambdaP:=0;
SumatoriaDeDif:=0;
For I:=1 to NumeroDeBuu+1 do
begin
If I=1 then LambdaT:=Lambdateca [I]
else LambdaP:=LambdaP + Lambdateca [I];
end;
LambdaP:=LambdaP/NumeroDeBuu;
For I:=2 to NumeroDeBuu+1 do
SumatoriaDeDif:=SumatoriaDeDif + sqr (Lambdateca [I]
- LambdaP);
Sesgo :=LambdaP-LambdaT;
LambdaB :=(2*LambdaT) - LambdaP;
Varianza:=(1/(NumeroDeBuu-1)) * (SumatoriaDeDif);
Desviacion:=sqrt(Varianza);
End;
{-----}
Procedure Acumula;
{ En caso de que se tengan varias muestras y se quiera obtener
los resultados promedio de todas las muestras generadas, cada
variable se acumula en otra hasta que ya no existan más muestras
}

```

```

Begin
  Calcula;
  ALambdaT:=ALambdaT+LambdaT;
  ALambdaP:=ALambdaP+LambdaP;
  ALambdaB:=ALambdaB+LambdaB;
  ASesgo:=ASesgo+Sesgo;
  AVarianza:=AVarianza+Varianza;
  ADesviacion:=ADesviacion+Desviacion;
End;
{-----}
                Procedure Escribe;
{
  Se escribe en la pantalla y en el archivo los resultados
  obtenidos
}
Begin
  ClrScr;
  writeln ('Población           ',NombreP);
  writeln ('Tamaño de muestra      ',TamMuestraReal:10);
  writeln ('Numero de muestras      ',NumeroDeMuestras:10);
  writeln ('Lambda T                ',ALambdaT/NumeroDeMuestras:10:4);
  writeln ('Lambda P                ',ALambdaP/NumeroDeMuestras:10:4);
  writeln ('Sesgo                   ',ASesgo/NumeroDeMuestras:10:4);
  writeln ('Lambda B                ',ALambdaB/NumeroDeMuestras:10:4);
  writeln ('Varianza                ',AVarianza/NumeroDeMuestras:10:4);
  writeln ('Desviación              ',ADesviacion/NumeroDeMuestras:10:4);

  write (salida,NombreP);
  write (salida,TamMuestraReal:10);
  write (salida,ALambdaT/NumeroDeMuestras:10:4);
  write (salida,ALambdaP/NumeroDeMuestras:10:4);
  write (salida,ASesgo/NumeroDeMuestras:10:4);
  write (salida,ALambdaB/NumeroDeMuestras:10:4);
  write (salida,AVarianza/NumeroDeMuestras:10:4);
  write (salida,ADesviacion/NumeroDeMuestras:10:4);
  writeln (salida,NumeroDeMuestras:10);

  Close (salida);
End;
{-----}
                Procedure DVB (No:Boolean);
{
  Mientras haya muestras la variable No será verdadera y sólo se
  realizará el procedimiento acumula, que calcula las lambdas y las
  acumula en otras variables. En el momento en que No sea falsa
  (quiere decir que ya no hay más muestras) se efectuará el procedi-
  miento Escribe, el cual escribe las variables que acumularon
  los resultados de cada muestra
}
Begin
  If No then Acumula

```

```

        else Escribe;
End;
{-----}
BEGIN
END.

```

UNIT DesVarJ;

```

{   Calcula, según la fórmula de Caswell, la lambda, promedio, la
    lambda jackknife, el sesgo, la desviación típica de las lambdas
    obtenidas al aplicar el jackknife a las muestras de una población
}

```

Interface

```

    Uses Crt,VarGlob, Printer;
    Procedure InicializaJ;
    Procedure DVJ (No:Boolean);

```

Implementation

CONST

```

    NombreDelArchivoConLambdasYac='Jfin';
                                     { Archivo con los resultados}

```

VAR

```

    unaLambda,
    LambdaJ,           ALambdaJ,
                       { Acumuladores de las variables }
    LambdaT,          ALambdaT,
                       { de la izquierda }
    LambdaP,          ALambdaP,
    Sesgo,            ASesgo,
    Varianza,         AVarianza,
    Desviacion,       ADesviacion,
    SumatoriaDeDif   :Real;
    Salida           :Text;
    Indicador        :Boolean;

```

```

{-----}

```

Procedure InicializaJ;

```

{   Inicializa las variables que acumulan la suma de otras, revisa
    si ya existe el archivo donde se guardan, si no existe lo crea con
    su encabezado.
    Este procedimiento se llama desde la unidad NAVAJA.TPU
}

```

Begin

```

    ALambdaJ:=0;
    ALambdaT:=0;
    ALambdaP:=0;
    ASesgo:=0;

```

```

AVarianza:=0;
ADEsviacion:=0;
Assign (salida,NombreDelArchivoConLambdasVac);
($I-) Append (salida); {$I+}
Indicador:=(IOresult=0);
If not indicador then
  begin
    rewrite (salida);
    writeln (salida,'Poblac      Tamaño      LambdaT
      LambdaP Sesgo      LambdaJ       $\sigma^2$        $\sigma$       #');
    writeln
      (salida,'-----');
    end;
End;
{-----}
          Procedure Calcula;
{
  Efectúa todos los cálculos para obtener las lambdas y demás
  variables
}
VAR
  I:Byte;
Begin
  LambdaP:=0;
  SumatoriaDeDif:=0;
  For I:=1 to TamMuestraReal+1 do
    Begin
      If I=1 then LambdaT:=Lambdateca [I]
        else LambdaP:=LambdaP + Lambdateca [I];
    End;
  LambdaP:=LambdaP/TamMuestraReal;
  For I:=2 to TamMuestraReal+1 do
    SumatoriaDeDif:=SumatoriaDeDif + sqr (Lambdateca [I]
      - LambdaP);
  Sesgo :=(TamMuestraReal-1) * (LambdaP-LambdaT);
  LambdaJ :=(TamMuestrareal*LambdaT) - (TamMuestraReal-1) *
    LambdaP;
  Varianza:=((TamMuestraReal-1)/TamMuestraReal) *
    (SumatoriaDeDif);
  Desviacion:=sqrt(Varianza);
End;
{-----}
          Procedure Acumula;
{
  En caso de que se tengan varias muestras y se quiera obtener
  los resultados promedio de todas las muestras generadas, cada
  variable se guarda en un acumulador hasta que ya no existan más
  muestras. Luego se dividirá entre el número de muestras para sacar
  el promedio
}

```

```

Begin
  Calcula;
  ALambdaT:=ALambdaT+LambdaT;
  ALambdaP:=ALambdaP+LambdaP;
  ALambdaJ:=ALambdaJ+LambdaJ;
  ASesgo:=ASesgo+Desviacion;
  AVarianza:=AVarianza+Desviacion*Desviacion;
  ADesviacion:=ADesviacion+Desviacion;
End;
{-----}
      Procedure Escribe;
{   Se escribe en la pantalla y en el archivo los resultados
obtenidos
}
Begin
  ClrScr;
  writeln ('Población           ',NombreP);
  writeln ('Tamaño de muestra      ',TamMuestraReal:10);
  writeln ('Numero de muestras     ',NumeroDeMuestras:10);
  writeln ('Lambda T               ',ALambdaT/NumeroDeMuestras:10:4);
  writeln ('Lambda P               ',ALambdaP/NumeroDeMuestras:10:4);
  writeln ('Lambda J               ',ALambdaJ/NumeroDeMuestras:10:4);
  writeln ('Sesgo                  ',ASesgo/NumeroDeMuestras:10:4);
  writeln ('Lambda J               ',ALambdaJ/NumeroDeMuestras:10:4);
  writeln ('Varianza               ',AVarianza/NumeroDeMuestras:10:4);
  writeln ('Desviación            ',ADesviacion/NumeroDeMuestras:10:4);

  write (salida,NombreP);
  write (salida,TamMuestraReal:10);
  write (salida,ALambdaT/NumeroDeMuestras:10:4);
  write (salida,ALambdaP/NumeroDeMuestras:10:4);
  write (salida,ASesgo/NumeroDeMuestras:10:4);
  write (salida,ALambdaJ/NumeroDeMuestras:10:4);
  write (salida,AVarianza/NumeroDeMuestras:10:4);
  write (salida,ADesviacion/NumeroDeMuestras:10:4);
  writeln (salida,NumeroDeMuestras:10);

  Close (salida);
End;
{-----}
      Procedure DVJ (No:Boolean);
{   Mientras haya muestras la variable NO será verdadera y sólo se
realizará el procedimiento Acumula, que calcula las lambdas y las
acumula en otras variables. En el momento en que NO sea falsa
(quiera decir que ya no hay más muestras) se efectuará el procedi-
miento Escribe, cuya función es escribir las variables que acumulan
los resultados de cada muestra.
}
Begin
  If No then Acumula

```

```

else Escribe;
End;
{-----}
BEGIN
END.

```

UNIT VarGlob;

```

{ Contiene todas las variables que se utilizan en más de una
unidad. También los procedimientos que usan varias unidades.
}

```

Interface

```

USES
  Crt;
TYPE
  RegistroMusical=record
    C,CF,D,DF,E,F,FF,G,GF,A,AF,B:Integer;
  end;
CONST
  Notas : RegistroMusical=
    (C:1;CF:2;D:3;DF:4;E:5;F:6;FF:7;G:8;GF:9;
    A:10;AF:11;B:12);
  Tiempo=20;
  NumeroDeBuu=1000;
  TamMuestraMax=200;
VAR
  I1000, { Contador de bootstraps }
  renglon,
  NumeroDeMuestras : Integer;
  Res,
  Metodo,
  Poblacion : Char;
  NombreP : String [6];
  { Nombre de la población total }
  TamMuestraReal : Byte;
  MuestraO : Array
    [1..TamMuestraMax,1..Tiempo] of byte;
  Lambdateca : Array [1..1001] of extended;
  { Almacén de lambdas de cada
  { muestra y de cada bootstrap
  { o jackknife que se realice }
  Hay : boolean;
  MillambdasDE : text;
  { Archivo con 1000 lambdas
  { obtenidas directamente }

  Procedure Sonido (Octava,Nota,Duracion,Repe:Integer);

```

Implementation

```

                Procedure Sonido(Octava,Nota,Duracion,Repe:Integer);
VAR
    I,J:Integer;
    Frecuencia:Real;
Begin
    Frecuencia:=32.625;
    For I:=1 to Octava do Frecuencia:=Frecuencia*2;
    For I:=1 to Nota-1 do Frecuencia:=Frecuencia*1.059463094;
    For J:=1 to Repe do
        Begin
            If duracion <> 0 then
                begin
                    Sound (Round(frecuencia));
                    Delay (Duracion);
                    NoSound;
                end else Sound (Round(frecuencia));
        End;
    End;
    End;
    (*****
BEGIN
END.

                UNIT Informa;

{   Crea una pantalla o ventana de información. Nos dice cuántas
muestras tenemos, de cuántos individuos, de qué población, qué
método estamos utilizando. También qué proceso se lleva acabo
muestra o en qué número de bootstrap o jackknife va.
}
Interface
    USES Crt,VarGlob;
    Procedure Pantalla (Cuenta:Integer);
Implementation
    Procedure Pantalla (Cuenta:Integer);
    { Cuenta lleva la cuenta del }
    { número de bootstraps o de }
    { jackknives }
VAR
    Letrero:String [9];
BEGIN
    Case Metodo of
        'A':Letrero:=' Directo';
        'B':Letrero:='Bootstrap';
        'J':Letrero:='Jackknife';
        'T':Letrero:=' Total';
        else Letrero:='*****';
    End;
    Window (25,1,55,9);
    TextBackground (Black);
    TextColor (Magenta);

```



```

Write (' ');
Write (' Población activa: ', NombreP, '| ');
Write (' Número de muestras: ', NumeroDeMuestras, '| ');
Write (' Tamaño de muestra: ', TamMuestraReal, '| ');
Write (' Método: ', Letrero, '| ');
Write (' Proceso: ', '| ');
Write (' ', Cuenta, '| ');
Write (' ');
Window (1,9,80,25);
END;
END.

```

UNIT Graf1;

```
{ $N+ }
```

```
Interface
```

```
TYPE
```

```
MorFec=Array [1..20] of extended;
```

```
Procedure Pintalas (y:MorFec;FoM:char);
```

```
Implementation
```

```
USES
```

```
Graph,Crt, varglob;
```

```
Procedure Pintalas (y:MorFec;FoM:char);
```

```
CONST
```

```
Margen=50;
```

```
X
```

```
:Array [1..20] of integer=
```

```
(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20);
```

```
VAR
```

```
I
```

```
:Byte;
```

```
xs, titulo
```

```
:string;
```

```
espera
```

```
:char;
```

```
FactorX, FactorY, RangoXd, RangoYd,
```

```
EscalaMinX, EscalaMaxX, EscalaMinY, EscalaMaxY :real;
```

```
Xmax, Ymax, RangoXp, RangoYp,
```

```
MaG, MoG, xp, yp, xi, yi, xf, yf :integer;
```

```
Function InvierteY (ygriega:integer):integer;
```

```
begin
```

```
InvierteY:=Ymax-ygriega;
```

```
end;
```

```
Function Xpantalla (EQUIS:integer):integer;
```

```
begin
```

```
Xpantalla:=round(FactorX*(EQUIS-EscalaMinX))+margen;
```

```
end;
```

```
Function Ypantalla (YE:real):integer;
```

```
begin
```

```
Ypantalla:=InvierteY (round(FactorY*(YE-EscalaMinY)))-margen;
```

```

end;

BEGIN
  EscalaMaxX:=20;
  EscalaMinX:=0;
  EscalaMaxY:=-1000040;
  EscalaMinY:= 1000000;
  For i:=1 to tiempo do begin
    if y[i] > EscalaMaxY then EscalaMaxY:=y[i];
    if y[i] < EscalaMinY then EscalaMinY:=y[i];
  end;
  Detectgraph (MaG,MoG);
  Initgraph (MaG,MoG,'C:\TP');
  Xmax:=getmaxX;
  Ymax:=getmaxY;
  xi:=margen;
  yi:=InvierteY(Margen);
  xf:=Xmax-margen;
  yf:=InvierteY (Ymax-margen);
  setcolor (5);
  If FoM='M' then begin
    EscalaMaxY:=1;
    titulo:='SOBREVIVENCIA';
  end
  else begin
    EscalaMaxY:=5;
    titulo:='FECUNDIDAD';
  end;
  settextstyle (1,0,3);
  outtextxy (300,10,titulo);
  setcolor (14);
  line (xi-10,yi,xf,yi);
  line (xi,yi+10,xi,yf);
  line (xi,yf,xi-10,yf);
  line (xf,yi,xf,yi+10);
  setcolor (2);
  str(EscalaMinX:4:1, xs);
  outtextxy (xi-10,yi+10,xs);
  str(EscalaMaxX:4:1, xs);
  outtextxy (xf-10,yi+10,xs);
  str(EscalaMaxY:4:1, xs);
  outtextxy (xi-50,yf,xs);
  str(EscalaMinY:4:1, xs);
  outtextxy (xi-50,yi,xs);

  RangoXp:=xf-xi;
  RangoYp:=yi-yf;
  RangoXd:=EscalaMaxX-EscalaMinX;
  RangoYd:=EscalaMaxY-EscalaMinY;
  FactorX:=Rangoxp/rangoxd;
  FactorY:=Rangoyp/rangoyd;
  setcolor (4);

```

```

for i:=1 to 20 do begin
  xp:=xpantalla (x[I]);
  yp:=ypantalla (y[i]);
  Putpixel (xp,yp,white);
  circle (xp,yp,5);
end;
espera:=readkey;
restorecrtmode;
END;
BEGIN
END.

```

UNIT Adorno;

```

Interface
  Uses Crt;
  VAR
    y:byte;
  Procedure Presentacion;
Implementation
  Procedure Presentacion;
VAR
  c:char;
  i,j,x:byte;
{-----} Procedure u; {-----}
Begin
  gotoxy (1,1);
  if y>0 then begin
    textcolor (yellow);
    textbackground (red);
    writeln ('<<<<<<< ET IN ARCADIA EGO >>>>>>>');
    end;
  writeln;
  if y=1 then begin
    textcolor (red);
    textbackground (yellow);
    writeln (' Información ');
    end
  else begin
    textcolor (yellow);
    textbackground (red);
    writeln (' Información ');
    end;
  if y=2 then begin
    textcolor (red);
    textbackground (yellow);
    writeln (' Crear una población ideal ');
    end
  else begin
    textcolor (yellow);
    textbackground (red);

```

```

        writeln (' Crear una población ideal ');
    end;
    if y=3 then begin
        textcolor (red);
        textbackground (yellow);
        writeln (' Muestrear una población ');
    end
    else begin
        textcolor (yellow);
        textbackground (red);
        writeln (' Muestrear una población ');
    end;
    if y=4 then begin
        textcolor (red);
        textbackground (yellow);
        writeln (' Salir al MS DOS ');
    end
    else begin
        textcolor (yellow);
        textbackground (red);
        writeln (' Salir al MS DOS ');
    end;
    if y=5 then begin
        textcolor (red);
        textbackground (yellow);
        writeln (' Graficar lx o mx ');
    end
    else begin
        textcolor (yellow);
        textbackground (red);
        writeln (' Graficar lx o mx ');
    end;
    if y=6 then begin
        textcolor (red);
        textbackground (yellow);
        writeln (' Significancia y Kolmogorov-Smirnov
                (95%) ');
    end
    else begin
        textcolor (yellow);
        textbackground (red);
        writeln (' Significancia y Kolmogorov-Smirnov
                (95%) ');
    end;
    if y=7 then begin
        textcolor (red);
        textbackground (yellow);
        writeln (' Salir del programa ');
    end
    else begin
        textcolor (yellow);
        textbackground (red);
    
```

```

        writeln (' Salir del programa ');
    end;
end;
BEGIN
    y:=1;x:=1;
    clrscr;
    textcolor (yellow);
    textbackground (red);
    for i:=1 to 25 do
        for j:=1 to 80 do write (' ');
        u;
        repeat
            c:=readkey;
            if c=#72 then y:=y-1;
            if y=0 then y:=7;
            if c=#80 then y:=y+1;
            if y=8 then y:=1;
            gotoxy (x,y);
            u;
        until c=#13;
    END;
BEGIN
END.

```

REFERENCIAS

- Ainley, D. y DeMaster, D. (1980) Survival and mortality in a population of Adélie penguins. *Ecology*. 61(3), 522-530.
- Álvarez-Buylla, E. y M. Slatkin. (1991) Finding confidence limits on population growth rates. *Trends Ecol. Evol.* 6(2):221-224.
- Finding confidence limits on population growth rates: Monte Carlo test of a simple analytic method. En prep.
- Baker, M., Mewaldt, R. y Stewart, R. (1981) Demography on white-crowned sparrows (*Zonotrichia leucophrys nuttalli*). *Ecology*. 62(3), 636-644.
- Ballinger, R. (1979) Intraspecific variation in demography and life history of the lizard, *Sceloporus jarrovi*, along an altitudinal gradient in southwestern Arizona. *Ecology*. 60(5), 901-909.
- Begon, M., Harper, J. y Townsend, C. (1988) *Ecology: Individuals, Populations and Communities*. Harper & Row, EE.UU.
- Birch, L. (1953a) Experimental background to the study of the distribution and abundance of insects, I: The influence of temperature, moisture and food of the innate capacity for the increase of three grain beetles. *Ecology*, 34:698-711.
- (1953b) Experimental background to the study of the distribution and abundance of insects, II: The relation between innate capacity for increase in numbers and the abundance of three grain beetles in experimental populations. *Ecology*, 34:712-726.
- Bronson, M. (1979) Altitudinal variation in the life history of the golden-mantled ground squirrel (*Spermophilus lateralis*). *Ecology*. 60(2), 272-279.
- Caswell, H. (1989) *Matrix Population Models*. Sinauer Press, EE.UU.
- Crouse, D., Crowder, L. y Caswell, H. (1987) A stage-based population model for loggerhead sea turtles and implications for conservation. *Ecology*. 68(5), 1412-1421.

- Culiney, T. y D. Pimentel. (1986) Effects on a chemically contaminated sewage sludge on an aphid population. *Ecology*. 67(2):16-65-1669.
- Choquenot, D. (1991) Density-dependent growth, body condition and demography in feral donkeys: testing the food hypothesis. *Ecology*. 72(3), 805-813.
- Diaconis, P. y Efron, B. (1983) Computer intensive methods in statistics. *Sci. Amer.* 248(5), 96-108.
- Frank, P. (1960). Prediction of population growth form in *Daphnia pulex* cultures. *The American Naturalist*. 94(878):199-214.
- Frazer, N., Gibbons, J. y J. Greene (1991). Life history and demography of the common mud turtle *Kinosternon subrubrum* in South Carolina, USA. *Ecology*. 72(6), 2218-2231.
- Freund, J. y Smith, R. (1989). *Estadística*. Prentice Hall, México.
- Gilbert, F. (1990). *Insect life cycles*. Springer-Verlag, Inglaterra.
- Goulden, C., Henry, L. y Tessier, A. (1982) Body size, energy reserves and competitive ability in three species of Cladocera. *Ecology*. 63(6):1780-1789.
- Grant, P. y Grant, R. (1992) Demography and the genetically effective sizes of two populations of Darwin's finches. *Ecology*. 73(3), 766-784.
- Grigg, R. (1977) Population dynamics of two gorgonian corals. *Ecology*. 58, 278-290.
- Groenendael, J. van, Kroon, H. de y Caswell, H. (1988) Projection matrices in population biology. *Trends Ecol. Evol.* 3, 264-269.
- Hansson, S., Rudstam, L. y Johansson, S. (1990) Are marine planktonic invertebrates food limited? The case of *Mysis mixta* (Crustacea, Mysidacea) in the Baltic Sea. *Oecologia*. 84,

430-432.

- Hasegawa, M. (1990) Demography of an island population of the lizard, *Eumeces okadae*, on Miyake-jima Izu islands. *Res. Popul. Ecol.* 32, 119-133.
- Houllier, F., Lebreton, J. y Pontier, D. (1989) Sampling properties of the asymptotic behavior of age- or stage-grouped population models. *Mathematical biosciences.* 95:161-177.
- Hughes, R. y Roberts, D. (1981) Comparative demography of *Littorina rudis*, *L. nigrolineata* y *L. neritoides* on three contrasted shores in north Wales. *Journal of animal ecology.* 50, 251-268.
- Joyanes, L. (1990) Programación en Turbo Pascal. McGraw-Hill, México.
- Kent, E. (1981) Life history responses to resource variation in a sessile predator, the ciliate protozoan *Tokophrya lemnae* Stein. *Ecology.* 62(2), 296-302.
- Kinnaird, M. y O'Brien, G. (1991) Viable populations for an endangered forest primate, the Tana river crested Mangabey. *Conservation Biology.* 5(2), 203-213.
- Krebs, Ch. (1985) Ecología. Estudio de la distribución y la abundancia. Harla, México.
- Landahl, J. y Root R. (1969) Differences in the life tables of tropical and temperate milkweed bugs, genus *Oncopeltus* (Hemiptera:Lygaeidae). *Ecology.* 50(4), 734-737.
- Lande, R. (1988). Demographic models of the northern spotted owl (*Strix occidentalis caurina*). *Oecologia* 75:601-607.
- Lasker, H. (1991) Population growth on a gorgonian coral: equilibrium and non-equilibrium sensitivity to changes in life history variables. *Oecologia.* 86, 503-509.
- Law, R. y Edley, M. (1990) *Ecology.* 71(5), 1863-1870.

- Leslie, P. (1945) On the use of matrices in certain population mathematics. *Biometrika*. 33(III):183-212.
- Lefkovitch, L. (1965) The study of population growth in organisms grouped by stages. *Biometrics*. 21, 1-18.
- Levin, L. y Huggett, D. (1990) Implications of alternative reproductive modes for seasonality and demography in an estuarine polychaete. *Ecology*. 71(6), 2191-2208.
- Meyer, J., Ingersoll, C., McDonald, L. y Boyce, M. (1986) Estimating uncertainty in population growth rates: jackknife vs. bootstrap techniques. *Ecology*. 67(5), 1156-1166.
- Messina, F. (1991) Life history variation in a seed beetle: adult egg-laying vs. larval competitive ability. *Oecologia*. 85, 447-455.
- Moloney, K. (1986) A generalized algorithm for determining category size. *Oecologia*. 69, 176-180.
- (1988) Fine-scale spatial and temporal variation in the demography of a perennial bunchgrass. *Ecology*. 69(5), 1588-1598.
- Nakamura, K., Abras, I. y Hasyim, A. (1989) Survivorship and fecundity of two sumatran tortoise beetles, *Aspidomorpha miliaris* y *A. sanctaerucis* (Coleoptera:Chrysomelidae) under laboratory conditions. *Res. Popul. Ecol.* 31, 25-34.
- Neill, W. (1975) Experimental studies of microcrustacean competition, community composition and efficiency of resource utilization. *Ecology*. 56, 809-826.
- Ohgushi, T. (1991) Lifetime fitness and evolution of reproductive pattern in the herbivorous lady beetle. *Ecology*. 72(6), 2110-2122.
- Perron, F. (1983) Growth, fecundity and mortality of *Conus pennaeus* in Hawaii. *Ecology*. 64(1), 53-62.

- Pinder, J., Wiener, J. y Smith, M. (1978) The Weibull distribution: a new method of summarizing survivorship data. *Ecology*. 59(1), 175-179.
- Price, P. (1984) *Insect ecology*. John Wiley, EE.UU.
- Polis, G. y Farley, R. (1980) Population biology of a desert scorpion: survivorship, microhabitat, and the evolution of life history strategy. *Ecology*. 61(3), 620-629.
- Rago, P.J. y Dorazio, R.M. (1984) Statistical inference in life-table experiments: the finite rate of increase. *Can. J. Fish. Aquat. Sci.* 41, 1361-1374.
- Readshaw, J. y Gerwen, C. (1983) Age specific survival, fecundity and fertility of the adult bowfly, *Lucilia cuprina*, in relation to crowding, protein food and population cycles. *Journal of animal ecology*. 52, 879-887.
- Rickart, R. (1988) Population structure of the Piute ground squirrel (*Spermophilus mollis*). *The Southwestern Naturalist*. 33(1), 91-96.
- Robertson, R. y G. Salt (1981) Responses in growth, mortality and reproduction to variable food levels by the rotifer, *Asplanchna girodi*. *Ecology*. 62(6), 1585-1596.
- Sarukhán, J. y Gadgil, M. (1974) Studies on plant demography: *Ranunculus repens* L., *R. bulbosus* L. and *R. acris* L. III. A mathematical model incorporating multiple modes of reproduction. *J. Ecol.* 62, 921-936.
- Skadsheim, A. (1990) A cohort life table for *Gammarus salinus*. *Oikos*. 57, 207-214.
- Slade, N. y Balph, D. (1974) Population ecology of Uinta ground squirrels. *Ecology*. 55, 989-1003.
- Tanaka, Y. (1990) Age specific of inbreeding depression during a life cycle of *Callosobruchus chinensis* (Coleoptera: Bruchidae). *Res. Popul. Ecol.* 32, 329-335.

Tinkle, D. y Ballinger, R. (1972) *Sceloporus undulatus*: a study of the intraspecific comparative demography of a lizard. *Ecology*. 53(4), 570-584.

Tinkle, D., Congdon, J. y Rosen, P. (1981) Nesting frequency and success: implications for the demography of painted turtles. *Ecology*. 62(6), 1426-1432.

Vanni, M. (1986) Fish predation and zooplankton demography: indirect effects. *Ecology*. 67(2), 337-354.

Vinegar, M. (1975) Demography of the striped plateau lizard, *Sceloporus virgatus*. *Ecology*. 56, 172-182.

Wilbur, H. (1975) The evolutionary and mathematical demography of the turtle *Chrysemys picta*. *Ecology*. 56, 64-77.

