A Simulation Study of Some Parameter Estimators for Animal Population Using Capture-Recapture Data

A Thesis

Presented to

The Faculty of Graduate Studies and Research

The University of Manitoba

In Partial Fulfilment

of the Requirements for the Degree

Master of Science

in the Department of Computer Science

by

Peter Wai-Lon Chan

October 1971

ABSTRACT

This thesis is an examination of the properties of the recapture frequency distribution obtained through the capture-recapture method. Two models are used for the simulation of biological population. The models are the Replacement model and the Immigration and Death model. The recapture frequency is compared with the truncated geometric distribution and the duration frequency is compared with Holgate's distribution.

Three estimators of population size, namely, the geometric, the negative binomial and a moment estimator based on Holgate's distribution are evaluated.

ACKNOWLEDGEMENTS

I wish to express my deepest thanks to Professor A. Neil Arnason, my supervisor, for his advice, guidance and assistance in the preparation of this thesis.

I am also grateful to Professor J. F. Lawless for his suggestions and comments.

TABLE OF CONTENTS

CHAPTER I

INTRODUCTION

In order to study certain behaviour of an animal population on a defined area, such as its birth rate, death rate, immigration, emigration rates into and out of the area, biologists frequently use sampling methods combined with marking of animals. The simplest of these methods (called capture-recapture methods) involves taking samples on two occasions. Assuming a closed system, (that is, not subject to addition or depletions) animals are captured, marked and then returned to the population. At some later stage, another trapping experiment is conducted. The second sample contains both marked and unmarked animals. The proportion of marked animals in the sample may reflect the proportion of total marked animals in the whole population truthfully. From this proportion, it is possible to get an estimate of the population size.

This simple recapture method is of limited use. It assumes a closed system, which in reality, may be impossible. It cannot satisfactorily take account of birth or death, or of any factor which may cause a change in the original population size. A more sophisticated and informative sampling experiment for estimating the properties of an animal population is the multiple capture-recapture experiment.

The multiple capture-recapture method is an extension of the simple recapture method. Several trappings are performed over a period of time. An animal captured is marked and then released into the population. In subsequent trappings, it has a probability of being captured again. The number of times each animal is captured is recorded. This can be done either by putting different marks on the animals for different times of captures or by giving a different mark such as a number to each animal captured. At the end of the experiment, a record of recapture frequency is available in the first case, and the complete capture history of each captured individual is available in the second case. Information about the population can be extracted by accurate interpretation of the recapture frequency table or capture history table.

Jolly (1965) derived a general probability distribution designed to fit capture-recapture problems for a homogeneous population. He also derived estimators for a single population that is subject to immigration and death. When a population is made up of a number of discrete homogeneous strata, it may be necessary to have seperate population estimators for each stratum. Jolly's estimates can be adapted to this situation so long as individuals do not change from one stratum to another in the course of the experiment.

With this model, Jolly is able to estimate the number of marked animals in the population, the total number of animals in the population, the probability of survival for an animal released after sampling and the number of animals joining the population between each sample is still available for the next sampling period. The asymptotic variance and covariance for the estimates of the population size, survival rate and the number of immigrants can also be derived, but it is reasonable to use these estimates of the true variance and covariance only when the number of animals captured is sufficiently large.

Jolly makes several assumptions for his general capture-recapture model. These are:

i) The equal survival assumption.

It is assumed that all animals, alive at sampling period t have the same probability of survival to time $t+1$, regardless of age, capture history or location.

ii) The equal catchability assumption.

It is assumed that all animals alive at time t have the same probability of being captured at time t regardless of age, previous capture history or location.

iii) The permanent emigration assumption.

It is assumed that if animals leave the population, they do so permanently. They do not leave the population, remaining absent and so uncatchable for one or more sampling periods, and return at some subsequent point in the experiment.

The validity of assumption (ii) is debatable. Many experimental data show that this assumption is not fulfilled. Cormack (1966) suggests that the probability that a particular individual is captured in any sample may be a property of the individual, or the probability that an individual is caught in any sample may depend on its previous capture history. On the other hand, Eberhardt (1968) postulates that a population might have equal catchability given the same exposure to trapping, but animals appear to have different probabilities of capture owing to variations in contact with traps.

An advantage of Jolly's method is that it gives estimates of the variance of the estimators. There is no restriction on sampling time, that is, there can be unequal time intervals between samplings. It allows for a stratified population with possibly different parameters applying to different strata. This can be the difference in sex or different age groups. Also, Jolly's method is the most complete and efficient estimation procedure available.

But Jolly's method has its disadvantages. For example, the calculations can be tedious, particularly in the calculation of the variances. It is difficult to combine estimates over sampling periods to get the total population, average life time and so on. It is also very sensitive to some failures of the assumptions. A more convenient procedure is to use the Geometric estimators or other simple estimators such as the one developed by Holgate (1964).

Tanton (1965) conducted an extensive capture-recapture experiment on wood mouse (Apodemus sylvaticus) and bank vole (Clethrionomys glareolus). He discovered that the probability of survival for the animals is quite uniform but that the data obtained from the capture-recapture method for estimating population size is unsatisfactory. There were very few animals caught in the summer months and the number of animals captured increased during the winter months. He attributes the difference to the fact that food is more abundant in the summer than in the winter, and hence fewer animals would fall into the traps. The probability of capture was not equal throughout the population, but the probability of capture for an individual is fairly constant during the experiment. The recapture frequency showed that a large number of animals were captured only once; Tanton believed that this was probably because of the presence of a large number of wide-ranging vagrants, and this inflated the estimates considerably.

Tanton (1969) used the stochastic model formulated by Jolly to analyse the data he obtained through another capture-recapture experiment. He confirms his findings published in 1965. The estimates produced by Jolly's capture-recapture method were indeed misleading for the summer months, but those for the winter months were quite realistic. The assumption of equal catchability broke down because many of the juveniles known to be present in the population were not caught. Males were captured more often in the summer months and females in the winter months. The assumption that animals released after capture would mingle randomly throughout the whole population also failed. This is due to the fact that these animals have a relatively small range within the area. The time interval between two capture periods was discovered to be important. When irregular intervals were used, unreasonable estimates were obtained. Tanton further suggested that the truncated Negative Binomial distribution should be used to fit to the frequency of recapture data and modified Geometric distribution to fit to the duration of residence for the animals in the area.

Eberhardt (1968) studied the fitting of the Geometric distribution to the observed distribution of frequency of recapture. Since the number of animals not captured is not known, the fitting was necessarily truncated at the zero class. Assuming equal probability of capture and a closed population with neither losses nor immigration taking place, he developed the geometric estimator, but he did not show its variance. He tested the goodness of fit of both the poisson distribution and the geometric distribution to forty sets of data on ten species of animals. His study showed that for most cases, the geometric distribution represents frequency of recapture data more closely than does the poisson distribution. He further showed that under certain assumptions, the geometric distribution for frequency of recapture could arise in populations not closed to birth or death. This result is discussed in more detail in the next chapter.

Bunham and Overton (1969) investigated the bias of the Geometric estimators when the assumption of equal catchability was violated. They considered a closed population and generated the capture probability P from five distributions: P equal to a constant, P taken from a uniform distribution on (0, 1) and from three Beta distributions. They tested the capture statistics on six estimators, namely the Petersen, Schnabel, Geometric, a modified Geometric estimator and two unnamed estimators. Their result indicted that for low capture probability, all estimators were poor. The Geometric estimator was best only for the Beta distribution ($B(1, b)$, $b > 1$). But on the whole, it performed poorly.

However, no study has been made of the adequacy of the geometric estimator when Jolly's assumptions hold. The present study is designed to investigate the bias of the geometric estimator for a population with birth and death, Jolly's assumptions plus the restriction of equal sampling periods. This study also investigates the behaviour of the zero class, and of a number of estimators based on the duration frequency distribution (Holgate's distribution; these estimators are defined in the next chapter). This behaviour is examined using two models for defining birth and death, and for a number of different values for original population size, mean life, number of sampling periods, capture probabilities, etc.

The adequacy of fit of the recapture data to the geometric distribution and of the observed duration data to Holgate's distribution were tested.

Three estimators of population size were evaluated; these were the Geometric, Negative Binomial, and a moment estimator based on Holgate's distribution. The bias and precision of these estimators were examined. Also, a variance estimate for the geometric estimator was looked at.

Eberhardt shows that, given the assumptions and restrictions mentioned above, a geometric distribution of recapture frequency must result in the limit as the number of sampling periods becomes large. Another purpose of this study will be to study the properties of the geometric estimator in small samples and to determine how many samples an experimenter must take in order to get an adequate fit to the geometric distribution.

CHAPTER II

MODELS, DEFINITIONS AND ESTIMATORS

We wish to examine the properties of the distribution of recaptures and of observed duration on an area where a population is not closed; that is it is subject to addition by birth or immigration and to depletion by death or emigration. We wish to impose the three assumptions of Jolly (1965) mentioned in the previous chapter. In particular, the second assumption (equal survival) prescribes that the life time of every individual comes from an exponential distribution with the same mean. In addition, we also assume that the population is stable (or in equilibrium); that is, the distribution of the number alive on the area at any particular time is independent of time. Moreover, this implies that the expected number alive on the area at any time does not change. This still allows us a certain amount of choice in simulating the mechanism of population turn-over, and two models, which are reasonably realistic, were used. These are the replacement model and the immigration and death model.

REPLACEMENT MODEL

The Replacement model assumes a stable population. Whenever an animal dies, he is immediately replaced by a new individual introduced from outside. This assumption is quite reasonable for many animal populations. For an area that can support a certain number of small animals of the same species such as rodents, animals would come into this area and occupy some space and set up their territorial boundary. This would go on until the area is filled to its capacity. When a new animal now arrives, he would sense these boundaries and be aware that there is no space for him and he would leave the area. When a resident animal dies, a vacuum is created and the space is taken up immediately by a migrant. This mechanism has been suggested by Calhoun (1963).

Due to the independence properties of the death process, each individual alive is subject to the same risk of death, and the length of time that an animal lives is not dependent on the existence of other animals, or on the density of the population. The life span of the animals thus has an exponential distribution with the parameter θ, where θ is the mean life expectancy on the area of all animals who enter the population.

A further consequence of the model is that NT, the number of animals alive at some time during the experiment of length T given that there are N_0 animals alive at the beginning of the experiment is distributed as

$$NT = N_0 + RT$$

where RT has the poisson distribution with mean $N_0 T/\theta$. Thus

$$E(NT) = N_0(1+T/\theta)$$
$$\mathrm{var}(NT) = N_0 T/\theta$$

IMMIGRATION AND DEATH MODEL

For this model, it is assumed that both birth and death can occur. But the whole population size is in equilibrium. The number of animals that die is balanced by the number of animals entering into the population. This assumption is reasonable since in reality, animal populations in an area do tend to be stable, at least over short periods of time.

Death is the same as in the Replacement model. Birth is by immigration. This model differs from the replacement model in that a replacement is not introduced after each resident animal dies. Immigration is independant of the death process and is steady. Animals enter the population randomly, according to the poisson process. The time between the arrival of two animals is a random variable from an exponential distribution. This model is well known in stochastic processes and the properties of the model are discussed in most texts on stochastic processes. (See, for example, Cox and Miller (1968)).

If the mean inter-arrival time is λ, mean life is θ, initial size of the population is N_0, and population at time t is NT, then for equilibrium to occur, (i.e. $E(NT) = N_0 \forall T$) we must have $N_0 = \theta/\lambda$. The number of births in (O, T), B_T, has the poisson distribution with mean T/λ and thus $N_T = N_0 + B_T$ has

$$E(N_T) = N_0 + T/\lambda$$

$$= N_0\ (1+T/\theta) \quad \text{since} \quad \lambda = \theta/N_0$$

$$\mathrm{var}\ (N_T) = T/\lambda$$

$$= N_0 T/\theta$$

NOTATION

Live trapping is conducted every week for T consecutive weeks. Capture probability is assumed to be constant for all animals in the population. Let

N_0 = original population size.

$F(x)i$ = number of animals caught x times at ith sampling period, where $i = 1, 2, 3, \ldots, T$ and $x = 1, 2, 3, \ldots, i$.

Nti = total number of animals alive in the population for some time between 0 and time i.

$$Si = \sum_x F(x)i$$

= number of different animals captured up to time i.

$$Ci = \sum_x xF(x)i$$

= number of captures made up to time i.

$H(0)i = F(1)i$

= number of animals captured only once up to time i.

$H(j)i$ = number of animals captured more than once such that the time between the first and the last capture is j weeks.

Si = $\sum_j H(j)i$.

k = $e^{-1/\theta}$.

= probability that an individual survives at sampling period $t+1$, given that it is alive at period t. (This probability as a consequence of the exponential distribution of life, is independent of t, the individual's age and of the number or behaviour of other individuals in the population.)

θ = the mean number of sampling intervals (weeks) that an individual lives in the area.

q = probability that an individual which is alive at the time of sampling is not taken in the sample.

r = the random variable that has the uniform distribution on $(0, 1)$ obtained through a random number generator.

$\hat{N}g$ = population size estimated by the Geometric estimator.

$\hat{N}h$ = population size estimated by Holgate's estimator.

$\hat{N}n$ = population size estimated by the Negative Binomial estimator.

$C = \sum_i C_i$

$S = \sum_i S_i$

GENERATION OF RANDOM VARIABLES FROM THE EXPONENTIAL DISTRIBUTION

Pseudo-random number generators are used to generate uniformly distributed random numbers between 0 and 1. The exponential distribution has a continuous cumulative function. It is possible to generate life span of animals by mapping a random number onto the function by using the probability integral transformation. The transformation is as follows.

Let r be a continuous random variable uniformly distributed between 0 and 1. It has probability element such that

$$\begin{aligned} f(r)dr &= dr, \quad 0 \leq r \leq 1 \\ &= 0, \quad \text{otherwise.} \end{aligned}$$

Let y be a random variable defined by

$r = H(y)$ where $H(y)$ is monotonically increasing and continuously differentiable.

Hence

$$h(y) = \frac{d}{dy} H(y)$$

$$\frac{dr}{dH(y)} = 1$$

therefore

$$dr = dH(y)$$

substituting $r = H(y)$

$$f(H(y)) = \frac{dH(y)}{dy}\, dy = h(y)dy$$

Thus y has probability density function $h(y)$ and cumulative function $H(y)$ where

$$H(y) = \int_{-\infty}^{y} h(t)dt.$$

Let y be the life span of an animal. Let y have an exponential distribution with the probability density function $1/\theta\ e^{-y/\theta}$.

Then

$$F(y) = \int_{0}^{y} 1/\theta\ e^{-u/\theta}\, du$$

$$= 1 - e^{-y/\theta}$$

If r is randomly distributed between 0 and 1, then $1-r$ is also uniformly distributed between 0 and 1. Therefore, set,

$$1-r = 1 - e^{-y/\theta}$$

$$r = e^{-y/\theta}$$

$$-\ln r = y/\theta$$

$$y = -\theta \ln r.$$

and y is a random observation from the required exponential distribution.

GEOMETRIC ESTIMATOR

The Geometric estimator was proposed and derived by Edwards and Eberhardt (1967).

Prob (an animal lives t weeks, and hence is available for t samples)

$$= (1-k)\,k^t \qquad t = 0, 1, 2, 3, \ldots$$

Then given equally spaced sampling periods and equal capture probability $(1-q)$, Eberhardt shows

P = prob (not caught)

$$= \frac{(1-k)}{1-qk}$$

Q = prob (caught at least once)

$$= \frac{(1-q)k}{1-qk}$$

Prob (caught n times)

$$= Q^n P \qquad n = 0, 1, 2, 3, \ldots$$

Hence n has the Geometric distribution with

$$\text{mean} = Q/P$$

$$\text{var.} = Q/P^2 \qquad n = 0, 1, 2, 3, \ldots$$

This shows that under the assumption that

1) all individuals are subject to the exponential distribution of life and

2) there is independent capture probability and constant capture probability on all sampling occasions and for all individuals.

3) there are equal intervals between sample periods and sampling is carried out over a large number of sampling periods,

then the distribution of observed capture frequency follows the zero truncated Geometric distribution.

The number of animals in the zero class comes from an untruncated Geometric distribution with the same parameter (P); then we have

$$\text{Prob (captured } n \text{ times}) = P\,Q^n$$

$$n = 0, 1, 2, 3, \ldots$$

and

$$E(S) = N \,.\, (\text{prob. an individual is captured at least once})$$

$$= N.Q$$

Chapman and Robson (1960) give an unbiased estimate for Q using observations from the untruncated Geometric distribution. This estimate is

$$\hat{Q} = \frac{C - S}{C - 1}$$

and since it does not make use of the observations in the zero class, $f(0)$, it is also available for observations from the truncated distribution. Thus using this estimate for q, and solving for N, gives the "Geometric estimator for N".

Hence

$$\hat{N}_g = \frac{S(C - 1)}{C - S}$$

An estimate for the asymptotic variance of this estimate can be obtained by the standard linearisation or delta technique, making use of the variance and covariance of S and C. These expressions are developed in Arnason and Chan (1971) and are:

$$\begin{aligned}(C-S)^4 \text{ var } \hat{N}_g = {} & (S(1-S))^2 \text{ var } C \\ & + (C(1-C))^2 \text{ var } S \\ & -2(S.C(1-S).(1-C)). \\ & \text{cov } (CS)\end{aligned}$$

$$\begin{aligned}\text{var } C &= NQ/P^2 \\ \text{var } S &= NPQ \\ \text{cov}(CS) &= NQ\end{aligned}$$

Estimating N by N_g, and Q by $(C-S)/(C-1)$ as above, and substituting in var N_g gives an asymptotic estimate for var N_g (valid for large C, S, $C-S$)

$$\hat{\text{var}}\ \hat{N}_g = \hat{N}_g.(C.S-C+S)/(C-S)^2$$

HOLGATE'S DISTRIBUTION

Holgate derives the distribution of observed duration on the area (as defined by $H(j)$ in the section of notations) under the same assumptions as used by Eberhardt (as given in the previous section), except that he assumes that an individual that lives T weeks is exposed to capture $T+1$ times. He thus ignores the time between entry of an animal and the first sample to which it is exposed.

He shows that

$$\text{Prob (not caught} \mid T=t) = q^{t+1}$$

$$\text{Prob (caught once} \mid T=t) = (t+1)pq^{t}$$

$$\text{Prob } (T=t) = (1-k)k^{t}$$

Therefore

$$\text{Prob (not caught)} = \frac{(1-k)q}{1-qk}$$

$$\text{Prob (caught once)} = \frac{(1-k)p}{(1-qk)^2}$$

$$\text{Prob (caught at least once)} = 1 - \frac{(1-k)q}{1-qk}$$

$$= \frac{(1-q)}{1-qk}$$

Since $E(S) = N \frac{(1-q)}{1-qk}$

using the moment estimate of N, we have

$$\hat{N}h = \frac{S(1-qk)}{1-q}$$

Letting mean observed duration be d, Holgate shows

$$d = \frac{k(1-q)}{(1-k)(1-qk)}$$

and let

$$\theta = H(0)/S$$

$$= \text{proportion of animals captured once}$$

he obtained the maximum likelihood estimators

$$\hat{k} = 1 - \frac{(1-\theta)}{d}$$

$$\hat{q} = \frac{\hat{k}-(1-\theta)}{\hat{k}\theta}$$

$$\text{var } \hat{k} = \frac{(1-k)^2(1-qk)}{S(1-q)}$$

$$\text{var } \hat{q} = \frac{(1-q)(1-qk)^2(1-qk)}{Sk^2(1-k)}$$

therefore

$$\hat{q}\hat{k} = \frac{\hat{k} - (1-\theta)}{\theta}$$

$$(1-\hat{q})\ \hat{k} = \frac{\hat{k}\theta - \hat{k} + (1-\theta)}{\theta}$$

$$1-\hat{q} = \frac{\hat{k}\theta - \hat{k} + (1-\theta)}{\hat{k}\theta}$$

$$= \frac{(\hat{k}-1)(\theta-1)}{\hat{k}\theta}$$

Hence

$$\hat{N}_h = S\,\frac{(1-\hat{k})}{\theta}\,\frac{\hat{k}\theta}{(\hat{k}-1)(\theta-1)}$$

$$= S\,\frac{\hat{k}}{(1-\theta)}$$

$$= S(\frac{1}{(1-\theta)} - \frac{1}{d})$$

$$= S(\frac{d-1+\theta}{d(1-\theta)})$$

NEGATIVE BINOMIAL DISTRIBUTION

It is often difficult to estimate the parameters for the truncated Negative Binomial distribution. Brass (1958) considered simple methods of fitting the truncated Negative Binomial distribution.

Let m and s^2 be the mean and variance of the observations with the zero class excluded.

Let M be the mean with zero class included.

Let Z be the number of animals in zero class.

And $Y = F(1)/S$

Then Brass gives

$$\hat{M} = m - s^2 Y/m(1-Y)$$

but $M = C/(S+Z)$

$\therefore$ set $\hat{M} = C/(S+\hat{Z})$

$$S+\hat{Z} = C/\hat{M}$$

$$\hat{N}_n = S+\hat{Z}$$

$$= C/\hat{M}$$

CHAPTER III

RESULTS

The mean life span of the animals was set to be equal to ten sampling periods. The experiment was performed over one hundred sampling periods with capture probability ranging from 0.05 to 0.95 in steps of 0.1. Two original population sizes were used to investigate the properties of the estimators. They were one hundred and two hundred animals. Each experiment was repeated fifty times. After every ten sampling periods, the statistics of the true population, the number of animals captured and the total number of captures were collected. The recapture frequencies were then compared with the geometric distribution fitted using maximum likelihood estimates for the parameters. The chi-square test was used to test for the goodness of fit. Also the duration frequencies were obtained and were compared with the fitted Holgate's distribution.

There is a difficulty in using the chi-square test as a measure of goodness of fit. For example, the geometric distribution for recapture frequency does not hold exactly for any finite number of sampling periods, T, but the data will tend to the geometric distribution as T becomes large. However, in a simulation study, using any fixed T, it is always possible to generate a sufficiently large number of observations that the Chi-square statistic will be significantly large. For all experiments, the recapture frequencies and duration frequencies were summed over fifty simulations and the chi-square test was applied to these pooled data. Since the number of simulations was kept constant at fifty for all simulation sets, the chi-square statistics can at least be used as a comparative measure of the goodness of fit over different simulation sets.

The properties of the estimators for sampling periods less than the mean life span of the animals were also investigated The life span of the animals was set to be twenty sampling periods and statistics were collected for periods from seven to twenty over the same range of capture probabilities.

Since animals are coming into the population randomly, the total population for each experiment is a random variable and increases as the length of the experiment increases. It was therefore difficult to compare the bias of the different estimators for different lengths of experiment. A relative error was used. The relative error was obtained by dividing the absolute value of the difference between the true and estimated population size by the true population size. In this way, the error was standardized.

The mean relative error over fifty simulations for the geometric, the Negative Binomial estimators and Holgate's distribution were plotted against the number of sampling periods. (Figures I, II, III, IV, V).

The greatest changes in bias and efficiency of the estimators were at low numbers of sampling periods and capture probabilities. Therefore, the relative absolute error, the bias and variance of the estimates for the geometric and Holgate's distribution over low capture intensity and short sampling periods were collected into table form for the two different models and for the different original populations sizes. (TABLES I, II, III, IV, V, VI).

The number of sampling periods increases across columns and the capture probability increases down the columns. Each entry in the table contains the relative absolute error, the sample bias and the sample variance of the estimator for the geometric and Holgate's estimators. Since for different capture probabilities, the expected population sizes are the same for the same number of sampling periods, the bias can be compared within the same row but the relative absolute error should be compared along the same column.

Below is given a schematic summary of the model and of parameter values used in the simulation study. The statistics collected for each estimator (and the definition of the statistics) is also given.

Replacement model

Immigration and death model

θ = 10 sampling periods	θ = 20 sampling periods
capture probability (P) from 0.05 to 0.95 in steps of 0.1.	
i.e. P = 0.05 (0.1) 0.95	P = 0.05 (0.1) 0.95
number of sampling periods from 10 to 100 in steps of 10	
i.e. t = 10 (10) 100	t = 7 (1) 20

ESTIMATES calculated for each simulation

$$\hat{N}_g,\ \hat{VAR}\ \hat{N}_g;\ \hat{N}_n;\ \hat{N}_h,\ \hat{k},\ \hat{VAR}\ \hat{k}$$

STATISTICS collected for each estimate over fifty simulations

	E = estimate			
mean and variance over fifty simulations of	B = mean bias	=	estimate - parameter value	
		=	E - T	
	k = relative absolute bias	=	$\frac{\lvert E - T \rvert}{T}$	(for N_g, N_h, N_n only)

PROPERTIES OF THE FREQUENCY OF RECAPTURE DISTRIBUTION

The recapture frequency seems to follow a geometric distribution. When the capture probability is large and the sampling period is long, some irregularity can be observed. These irregularities only occur at the tail end of the recapture frequency and are so slight that they can be ignored. When the capture probability is low, few animals are captured in the experiment and the chi-square test does not detect any significant difference between the recapture frequency and the maximum likelihood estimate of the geometric distribution. For capture probability of 0.05, after sixty sampling periods, the chi-square values are between two and four with four or five degrees of freedom. The best value is 2.69 with five degrees of freedom. The chi-square value becomes very large when the capture probability increases. For example, at forty sampling periods, for capture probability 0.35, the chi-square value is 73 with 19 degrees of freedom, and is 93 with 26 degrees of freedom for capture probability 0.55. The maximum likelihood estimator greatly over-estimates the ones class, $f(1)$, when the sampling period is short and hence gives poor chi-square values; also at high capture probability, many animals are captured, and any slight deviation from the geometric distribution can be detected by the chi-square test.

For the same capture probability, the chi-square statistics become smaller as the length of the experiments lengthens. For example, at capture probability 0.25, at ten sampling periods, the chi-square value is 289 with five degrees of freedom; at forty sampling periods, it is 51 with fourteen degrees of freedom; at forty sampling periods, it is 18.38 with 19 degrees of freedom.

Therefore it seems to agree with Eberhardt that a geometric distribution does occur in the limit as the number of sampling periods becomes large.

PROPERTIES OF THE GEOMETRIC ESTIMATOR

The geometric estimator always over-estimates the true population size. When the capture probability is low, very few animals are recaptured and most of the animals captured are in the one-class. This causes the zero class to be seriously over-estimated. When the capture probability is higher than 0.35, the majority of the animals are captured at least once and the zero class does not follow a geometric distribution. It appears that the capture probability does not affect the geometric estimators as much as the number of sampling periods.

For the Replacement model with original population size equal to 100, the greatest improvement in the geometric estimator with increasing capture probability is when the number of sampling period is small. For example (in table I) at ten sampling periods, when the expected value of N is equal to 200, the bias is 147.86 at capture probability 0.05 and is 89.8 at capture probability 0.15, the improvement is about 39%. At twenty sampling periods, the expected value of N is 400, the bias is 98.36 at capture probability 0.05 and is 60.86 at capture probability 0.15, the improvement is 37%. But at 100 sampling periods, the improvement in going from $P = 0.05$ to $P = 0.15$ is only 17%.

When the capture probability is more than 0.35, the improvement with increase in capture probability is slight even for a small number of sampling periods. For example, at ten sampling periods, the improvement in bias between capture probability 0.55 and 0.75 is only 9% and at twenty sampling periods, the improvement in bias between the same capture probabilities is 7.5%.

This is also true for different mean life span. When θ is set to be twenty sampling periods, at fifteen sampling periods, the improvement in bias between capture probability 0.15 and 0.25 is 33% and between capture probabilities 0.35 and 0.45 is 18%. At twenty sampling periods, the improvement in bias between capture probabilities 0.15 and 0.25 is 30% and between capture probabilities 0.35 and 0.45 is 15%. (The values are in table V.)

Also, after thirty sampling periods, the bias for different capture probabilities except 0.05, are approximately the same. For example, (referring to table I), at forty sampling periods, the bias for capture probabilities 0.15, 0.25, 0.55, 0.75, 0.95 are 55.72, 50.76, 45.42, 44.6, 45.24 respectively. At one hundred sampling periods, they are 75.6, 71.8, 72.3, 73.9 and 74.1. Therefore, when the sampling periods are large, no matter how intensively an experimenter does his sampling, the bias would be about the same.

The Geometric estimator improves as the number of sampling period increases. The greatest change is between ten and twenty sampling periods and at low capture probabilities. For example, at ten sampling periods, the relative error is 0.7832 at capture probability 0.05. At twenty sampling periods, it is 0.3383. The change is significant until capture probability is greater than 0.55, where the change is about 0.09. After forty sampling periods, the change for any probability is minimal.

Through observing different original population sizes and life span of the animal, the absolute error seems to be relative to both the capture probability and the length of the sampling periods. For low capture probabilities, the number of samplings has to be large and vice versa. To get an acceptable bias and error for a minimum effort, it is best to perform the experiment between two and three times the mean life span of the animal or at least twenty sampling periods and to capture about one quarter to one third of the animals in the population at each sampling occasion.

The variance of the estimator follows the same general trend except when the sampling period becomes very large. This is due to the fact that the number of animals increases considerably when the time of the experiment increases and hence the bias and the variance also increase.

The greatest improvement in variance is also between twenty and thirty sampling periods and between capture probability 0.23 and 0.35. For example, for the replacement model, with mean life span equal to ten sampling periods, (Table I) at thirty sampling periods, when expected mean population size is 400, the variance is 783.17 at capture probability 0.25 and 440.25 at capture probability 0.55 and 391.45 at capture probability 0.75.

At forty sampling periods, the mean population size is about 500, the variance is 764.80 at capture probability 0.25, 571.77 at capture probability 0.55 and 580.96 at capture probability 0.75.

The variance estimate over-estimates the variance of $\hat{N}_g$ when capture probability was 0.05 and under-estimates it for other capture probabilities. The variance estimate is quite close to the variance for most of the time except for long sampling periods. It then greatly under-estimates the variance. The best results are between capture probabilities 0.05 and 0.55, and sampling periods less than thirty. The mean of $\hat{\text{var}}\ \hat{N}_g$ over fifty simulations is plotted along with the values of the sample variance var $\hat{N}_g$ in figures VI to XI.

PROPERTIES OF THE OBSERVED DURATION DISTRIBUTION AND HOLGATE'S ESTIMATOR

The adequacy of fit of the observed duration data to Holgate's distribution is very much like the fit of recapture data to the geometric distribution. The irregularities are more obvious but are also confined to the tail end of the distribution. The Holgate's distribution has a very long tail and the fit is poor when the number of sampling period is small. The fit improves as the number of sampling periods increases.

Except at capture probability 0.05, the Holgate's estimator always under-estimates the true population size. Increase in capture probability does not improve the accuracy of the estimator. For a fixed number of sampling periods, the bias for each different capture probability is about the same, but the variance of the estimator becomes smaller with the increase in capture probability.

The lengthening of the sampling periods does improve the estimator for all capture probabilities. For example, (table I) with capture probability 0.05, and sampling periods 10, 20, 30, 40, 100, the relative absolute bias are 0.119, 0.089, 0.062, 0.049. For capture probability higher than 0.55, there seems to be an increase in relative absolute error when sampling period is greater than 70. It appears that at around 80 sampling periods, negative errors are produced, and the absolute value increased the relative absolute error.

The maximum likelihood estimate for k is very close to the true value of k as the number of sampling periods or the capture probability increases. When capture probability reaches 0.25, after twenty sampling periods, the two values are exactly the same to four places of decimal. Since the value of k is less than 1, the variance of k is a very small number, therefore the standard deviation, s, is used instead.

The estimate $\hat{s}$ given by Holgate always under-estimates the true s. The two values become quite close as the sampling periods lengthened.

The values for true k, estimate of k, its standard deviation, and $\hat{s}$ are listed in table VII.

PROPERTIES OF NEGATIVE BINOMIAL ESTIMATOR

The Negative Binomial estimator gives fairly good estimates when the number of sampling period is small and sampling intensity is low. When capture probability increases or sampling time lengthens, the estimator becomes bad. The bias does not improve with increase in capture probability. It improves up to capture probability about 0.25 and gets worse from then on. It is also poor when capture probability is 0.05. The relative absolute error is best when sampling period is about twenty, it then gets worse and levels off at about sixty sampling periods. The variance improves as the sampling periods lengthened, and capture probability increases, but as a whole, it is quite poor.

The estimator is only acceptable as regards bias, variance, and relative error when sampling periods are about ten to twenty and capture probability between 0.15 and 0.25.

COMPARISON

From every point of view, Holgate's estimator is better than the geometric estimator. The bias of Holgate's estimator at every sampling intensity is less than that of the geometric estimator and the same is true of the absolute relative error for all sampling periods. Also, the variance of Holgate's estimator is much smaller.

The bias and relative error of the negative binomial estimator lies between the geometric and Holgate's estimator for short sampling periods, but given that the number of sampling periods becomes large, (see figures I to V), the geometric estimator would definitely become better than the negative binomial estimator. The variance of the negative binomial estimator is much larger than that of the geometric estimator for all numbers of sampling periods and for all capture probabilities. The estimator N_n was thus considered to be of little value.

The result obtained from the Immigration and Death model is very similar to that of the Replacement model. When capture probability is 0.05, the bias, relative error and variance of both the geometric and Holgate's estimators in the Immigration model is slightly higher than that of the Replacement model. Whereas when capture probability is greater than 0.15, the bias, relative error and variance of the geometric estimator is lower in the Immigration and Death model than in the Replacement model. The Holgate's estimator is about the same for both of the two models. These differences may be entirely due to sample variability.

When the original population size is doubled, the bias and variance of both the geometric and Holgate's estimator increase. This is understandable since the true population size is also increased. The relative absolute error for both of the estimators decreased, giving more accurate estimates of the true population size.

The Immigration and Death model is more readily influenced by the increase in original population size than the Replacement model. The bias, absolute relative error and variance for the geometric and Holgate's estimators become greater in the Immigration and Death model than in the Replacement model.

When the mean life span of the animal was set to be twenty sampling periods, the simulations were only performed for twenty sampling periods. The purpose was to examine more closely the behaviour of the estimators when the number of samplings were less than the mean life span of the animal. From the data obtained in these simulations, the change in mean life span of the animals does not have much effect on the models, distributions or estimators.

CHAPTER IV

TABLES AND GRAPHS

Tables I to VI are tables of the relative absolute error, bias, and variance for the geometric and Holgate's estimators.

Table VII indicates the k value estimated by Holgate's distribution, the value of true k, the standard deviation for the estimate of k and the $\hat{s}$ of $\hat{k}$.

Figures I to V are the graphs for the relative absolute error of the geometric, the negative binomial and Holgate's estimators.

Figures VI to XI are comparisons of the var $\hat{N}_g$ and $\hat{\text{Var}}\ \hat{N}_g$. The vertical lines indicate 95% confidence intervals of the values.

TABLE I

REPLACEMENT MODEL
Q = 10 n_0 = 100
MEAN AND VARIANCE OVER 50 SIMULATIONS

T \ P		.05 G	.05 H	.15 G	.15 H	.25 G	.25 H	E(N) (± 2 s.d. (N))
10	R	.7832	.4126	.4713	.1190	.3907	.1021	200
	B	147.86	30.06	89.8	-5.50	75.02	-15.62	(± 20)
	V	22078.93	11763.4	1220.23	932.99	741.13	454.73	
20	R	.3383	.2095	.2990	.0890	.1770	.0639	300
	B	98.36	11.71	60.86	-15.8	51.48	-13.12	(± 28.3)
	V	6853.65	6239.1	806.52	594.15	805.43	554.25	
30	R	.1867	.1449	.1447	.0619	.1246	.0559	400
	B	72.9	-12.65	56.74	-13.72	48.58	-16.38	(± 34.6)
	V	4517.73	4397.17	1274.24	902.79	783.17	639.47	
40	R	.1605	.1243	.1136	.0672	.1039	.0530	500
	B	78.88	-14.59	55.72	-19.14	50.76	-20.52	(± 40)
	V	5130.8	5709.42	2064.47	1554.88	764.80	699.43	
100	R	.0835	.0833	.0688	.0494	.0658	.0469	1100
	B	91.86	-60.68	75.68	-46.23	71.8	-49.73	(± 63.2)
	V	108026.67	11284.52	3524.40	2658.51	2013.8	1735.0	

TABLE I continued

T	P	.55 G	.55 H	.75 G	.75 H	.95 G	.95 H	E(N) (±2 s.d.(N))
10	R	.2489	.1022	.2239	.0955	.2021	.0800	200
	B	47.04	-19.25	42.5	-18.15	38.78	-15.37	(± 20)
	V	221.75	138.30	150.59	82.34	210.88	107.71	
20	R	.1418	.0495	.1320	.0500	.1281	.0442	300
	B	40.96	-13.10	38.16	-14.32	37.46	-12.41	(± 28.3)
	V	262.28	262.26	267.98	218.06	456.94	235.24	
30	R	.1091	.0470	.1060	.0420	.1032	.0396	400
	B	42.26	-16.26	41.32	-16.17	40.68	-15.48	(± 34.6)
	V	440.25	416.27	391.45	302.25	539.62	336.08	
40	R	.0930	.0441	.0910	.0392	.0916	.0378	500
	B	45.42	-20.64	44.6	-19.12	45.24	-18.10	(± 40.0)
	V	571.77	474.93	580.96	517.86	547.48	381.98	
100	R	.0661	.0462	.0678	.0413	.0680	.0435	1100
	B	72.26	-50.44	73.92	-45.03	74.12	-47.41	(± 63.2)
	V	1688.79	1297.56	1087.74	9119.09	1281.11	853.67	

TABLE II

REPLACEMENT MODEL
$Q = 10 \qquad n_0 = 200$
MEANS AND VARIANCE OVER 50 SIMULATIONS

T \ P		.05 G	.05 H	.15 G	.15 H	.25 G	.25 H	E(N) ($\pm$ 2 s.d. (N))
10	R	.5647	.2123	.4793	.0878	.3740	.0889	400
	B	214.1	12.16	182.76	-8.28	141.52	-28.60	($\pm$ 28.3)
	V	21592.06	11253.2	2147.86	1423.53	1155.38	736.25	
20	R	.2777	.1267	.2080	.0631	.1707	.0564	600
	B	161.46	2.921	120.34	-21.02	98.6	-28.95	($\pm$ 40)
	V	10185.47	10848.02	1984.58	2114.04	900.17	794.69	
30	R	.1690	.0967	.1314	.0588	.1236	.0492	800
	B	131.98	-28.79	102.16	-35.64	96.04	-32.85	($\pm$ 48.9)
	V	9890.42	7206.25	2021.85	2637.15	1399.6	1490.77	
40	R	.1409	.0876	.1080	.0521	.0996	.0459	1000
	B	137.58	-31.46	105.2	-42.61	97.28	-43.18	(56.4)
	V	9634.57	9014.30	1833.61	2260.83	1882.11	1788.35	
100	R	.0758	.0649	.0697	.0484	.0663	.0471	2200
	B	166.06	-100.62	151.64	-102.93	144.32	-101.71	($\pm$ 89.5)
	V	17370.60	23280.41	4565.53	4308.72	3434.55	3834.58	

TABLE III

IMMIGRATION AND DEATH MODEL

$\theta = 10 \quad \lambda = .1 \quad N_0' = 100$

MEAN AND VARIANCE OVER 50 SIMULATIONS

T \ P		.05		.15		.25		E(N) (± 2 s.d. (N))
		G	H	G	H	G	H	
10	R	.7959	.4501	.4185	.1279	.3284	.1170	200
	B	138.34	20.07	78.64	-13.82	61.92	-18.21	(± 20)
	V	31959.52	14544.63	1594.20	708.14	411.75	340.48	
20	R	.3037	.2002	.1783	.0914	.1449	.0763	300
	B	84.06	4.23	49.52	-14.51	42.16	-19.71	(± 28.3)
	V	6884.30	5322.22	1493.65	1074.32	557.25	402.85	
30	R	.2110	.1523	.1253	.0760	.0996	.0665	400
	B	65.7	-13.45	45.46	-18.08	38.6	-24.44	(± 34.6)
	V	6192.32	5404.31	1752.36	1382.76	543.72	452.68	
40	R	.1509	.1232	.0880	.0721	.0840	.0647	500
	B	52.48	-25.77	40.32	-28.84	40.5	-28.78	(± 40)
	V	6195.98	6040.15	1831.42	1425.21	720.27	725.82	
100	R	.0834	.0837	.0545	.0616	.0603	.0567	1100
	B	65.7	-54.37	54.82	-64.82	65.58	-61.45	(± 63.2)
	V	9974.64	11278.48	3278.68	2714.91	2144.27	1846.19	

TABLE IV

IMMIGRATION AND DEATH
$\theta = 10 \quad \lambda = .05 \quad n_0 = 200$

T	P	.05 G	.05 H	.15 G	.15 H	.25 G	.25 H	E(N) ± (2 s.d.)
10	R	.6357	.2281	.3813	.1192	.2883	.1364	400
	B	239.28	35.06	144.2	-34.92	109.52	-51.54	(± 28.3)
	V	19161.80	11178.61	2020.9	1378.11	1013.22	683.07	
20	R	.2761	.1449	.1740	.0970	.1395	.0923	600
	B	158.04	3.2123	100.6	-44.04	80.82	-53.05	(± 40)
	V	10171.80	9972.76	1823.67	2251.36	1011.13	803.34	
30	R	.1651	.1048	.1147	.08	.0968	.0778	800
	B	128.32	-22.76	89.44	-53.92	75.7	-60.81	(± 48.9)
	V	11135.92	11571.31	2375.13	2802.88	1280.52	1177.75	
40	R	.1271	.0868	.0913	.0681	.0848	.0665	1000
	B	123.78	-37.26	89.6	-65.898	83.08	-64.68	(± 56.4)
	V	9871.33	10014.43	2723.96	2673.14	1432.14	1350.57	
100	R	.0692	.0675	.0637	.0573	.0599	.0549	2200
	B	150.28	-112.63	138.48	-122.13	130.34	-119.55	(± 89.5)
	V	17265.8	17200.97	5610.44	6505.65	2596.29	3678.23	

TABLE IV continued

T \ P		.45				
		G	H			
10	R	.2231	.1473			
	B	85.22	-56.21			
	V	539.10	317.51			
20	R	.1192	.0816			
	B	68.98	-46.67			
	V	890.57	543.03			
30	R	.0879	.07			
	B	68.10	-54.53			
	V	770.43	602.02			
40	R	.0759	.0636			
	B	74.12	-62.18			
	V	843.64	781.21			
100	R	.0613	.0521			
	B	133.16	-133.40			
	V	3189.43	2591.01			

TABLE V

REPLACEMENT MODEL
$\theta = 20 \quad n_0 = 100$

T \ P		.05 G	.05 H	.15 G	.15 H	.25 G	.25 H	E(N) $\pm$ 2 s.d.
7	R	1.2144	.6202	.7979	.1529	.6037	.1210	135
	B	150.48	43.22	98.7	5.22	74.34	-13.68	($\pm$ 62)
	V	22968.2	11097.42	1089.77	696.97	199.31	106.14	
10	R	.9450	.4676	.5506	.1092	.4073	.0930	150
	B	127.3	45.67	74.74	1.53	54.85	-10.31	(14)
	V	14726.93	6796.95	701.05	406.24	194.6	156.49	
15	R	.6436	.2915	.3663	.0829	.2492	.0799	175
	B	99.64	36.93	57.52	.56	38.38	-11.5	(16.15)
	V	2792.09	1744.07	391.8	322.8	146.29	106.13	
20	R	.4223	.1991	.2603	.0676	.1787	.0553	200
	B	73.3	21.60	46.34	-.8163	31.42	-8.1201	(20)
	V	1495.20	114.5	364.6	274.62	153.66	142.6	

TABLE V continued

T \ P		.35		.45			
		G	H	G	H		
7	R	.5042	.1423	.4003	.1699		
	B	62.68	-17.62	50.14	-21.28		
	V	140.52	95.36	106.61	67.55		
11	R	.3233	.1088	.2583	.1178		
	B	43.92	-14.5	35.5	-16.20		
	V	98.28	68.77	103.22	67.92		
15	R	.2016	.0695	.1627	.0729		
	B	31.32	-10.01	25.48	-11.09		
	V	134.64	114.62	96.76	74.32		
20	R	.1423	.0466	.1176	.0497		
	B	25.1	-7.63	20.86	-8.50		
	V	154.88	110.08	116.10	103.55		

TABLE VI REPLACEMENT AND DEATH MODEL

$\theta = 20 \quad n_0 = 200$

T	P	.05 G	.05 H	.15 G	.15 H	.25 G	.25 H	E(N) ± 2 s.d.
	R	.9488	.3548	.7583	.100	.6406	.0926	
7	B	234.54	52.93	187.66	.7247	159.98	-17.23	270 (16.1)
		17821.26	1364.58	1351.62	919.19	752.69	494.61	
	R	.7853	.2439	.5217	.0660	.4148	.0758	
10	B	212.1	54.62	141.12	.4986	113.62	-18.57	300 (20)
	V	11354.48	4980.7	874.76	584.56	432.83	305.22	
	R	.5743	.2050	.3565	.0594	.2610	.0573	
15	B	178.28	52.61	110.3	-2.99	81.76	-15.94	300 (24.2)
	V	4804.2	3013.34	606.72	543.28	286.29	288.98	
	R	.3953	.1385	.2533	.0503	.1808	.0496	
20	B	138.9	34.25	89.18	-5.90	64.12	-15.21	400 (28.1)
	V	3551.94	2449.78	670.48	513.93	391.80	355.63	

TABLE VII

REPLACEMENT MODEL

6 = 10 n_0 = 100

T	P	.05	.15	.25	
10	KH	.6793	.7207	.7435	
	K	.6810	.7249	.7451	
	S	.0804	.0316	.0206	
	$\hat{S}$	.0122	.0062	.0045	
20	KH	.8136	.8299	.8451	
	K	.8192	.8311	.8463	
	S	.039	.0162	.0110	
	$\hat{S}$	.0088	.0048	.0036	
20	KH	.8542	.8612	.8689	
	K	.8556	.8621	.8691	
	S	.0019	.0127	.0079	
	$\hat{S}$	.0099	.0041	.0031	
40	KH	.8682	.8743	.8961	
	K	.8707	.8749	.8961	
	S	.0196	.0097	.0042	
	$\hat{S}$	.0065	.0031	.0027	
100	KH	.8932	.8944	.8791	
	K	.8939	.8946	.8791	
	S	.0089	.0056	.0061	
	$\hat{S}$	.0042	.0024	.0027	

TABLE VII continued

T	P	.55	.75	.95
10	KH	.7866	.7949	.8026
	K	.7866	.7950	.8026
	S	.0123	.0111	.0107
	$\hat{S}$	.0029	.0025	.0022
20	KH	.8539	.8560	.8570
	K	.8540	.8560	.8570
	S	.0065	.0063	.0092
	$\hat{S}$	.0024	.0020	.0018
30	KH	.8729	.8732	.8730
	K	.8730	.8732	.8730
	S	.0061	.0058	.0015
	$\hat{S}$	.0021	.0017	.0015
40	KH	.8813	.8818	.8813
	K	.8813	.8818	.8813
	S	.0053	.0047	.0050
	$\hat{S}$	.0018	.0016	.0013
100	KH	.8956	.8959	.8957
	K	.8956	.8959	.8957
	S	.0035	.0028	.0031
	$\hat{S}$	.0012	.0011	.0009

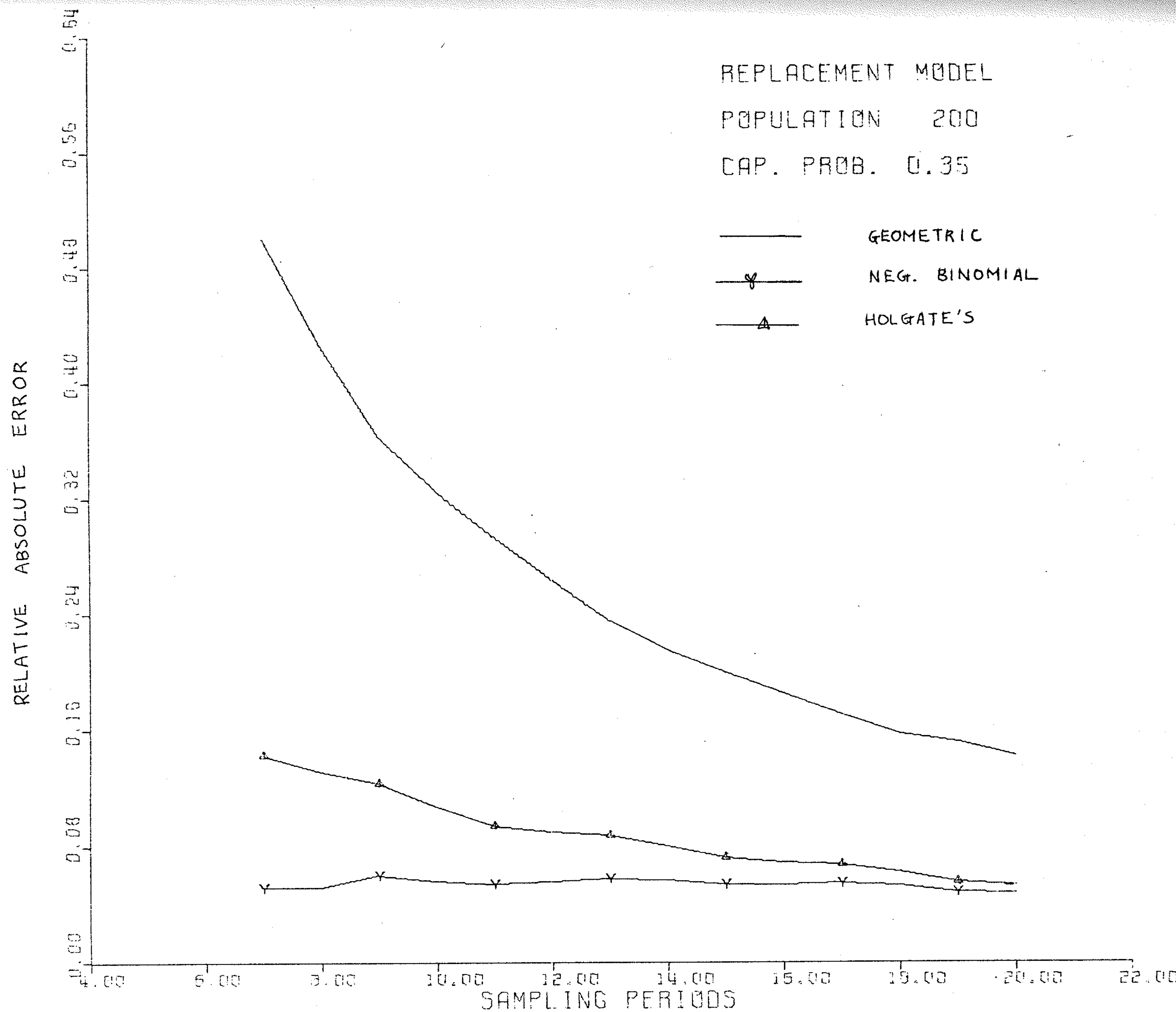

REPLACEMENT MODEL
POPULATION 200
CAP. PROB. 0.35
GEOMETRIC
NEG. BINOMIAL
HOLGATE'S
RELATIVE ABSOLUTE ERROR
SAMPLING PERIODS

FIGURE I

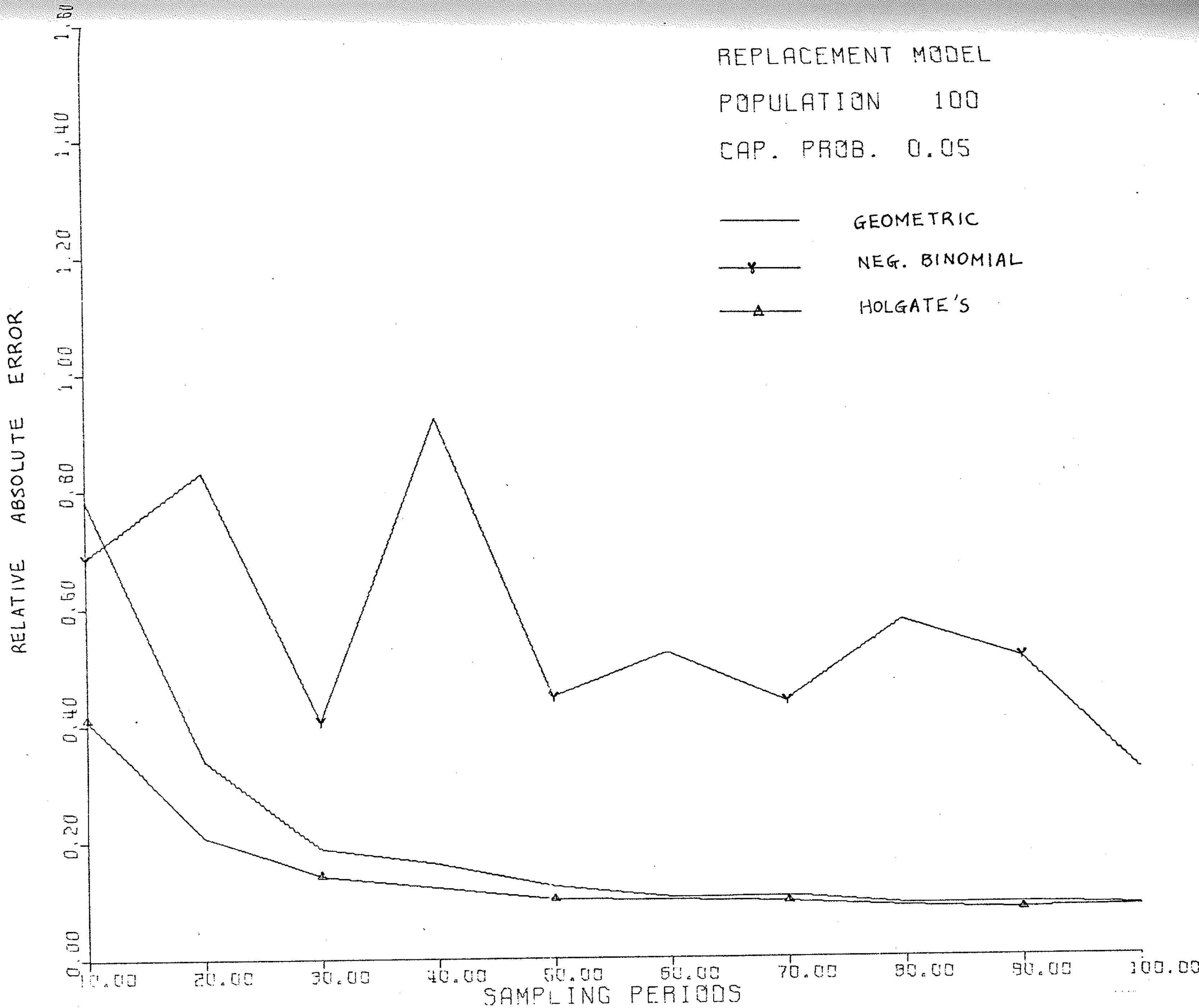

REPLACEMENT MODEL
POPULATION 100
CAP. PROB. 0.05
GEOMETRIC
NEG. BINOMIAL
HOLGATE'S
RELATIVE ABSOLUTE ERROR
0.00
0.20
0.40
0.60
0.80
1.00
1.20
1.40
1.60
10.00
20.00
30.00
40.00
50.00
60.00
70.00
80.00
90.00
100.00
SAMPLING PERIODS

FIGURE II

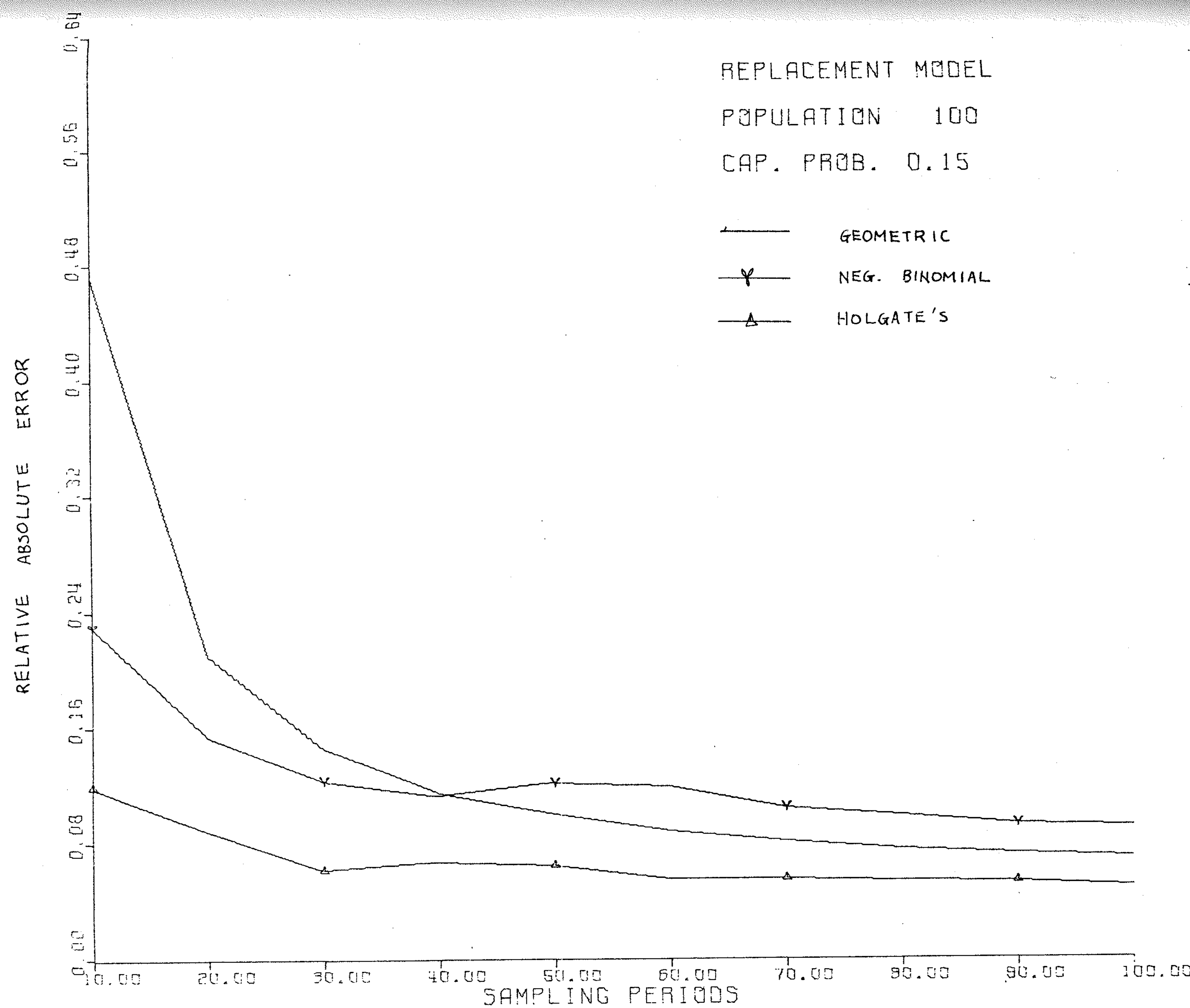

REPLACEMENT MODEL
POPULATION 100
CAP. PROB. 0.15
GEOMETRIC
NEG. BINOMIAL
HOLGATE'S
RELATIVE ABSOLUTE ERROR
0.00
0.08
0.16
0.24
0.32
0.40
0.48
0.56
0.64
10.00
20.00
30.00
40.00
50.00
60.00
70.00
80.00
90.00
100.00
SAMPLING PERIODS

FIGURE III

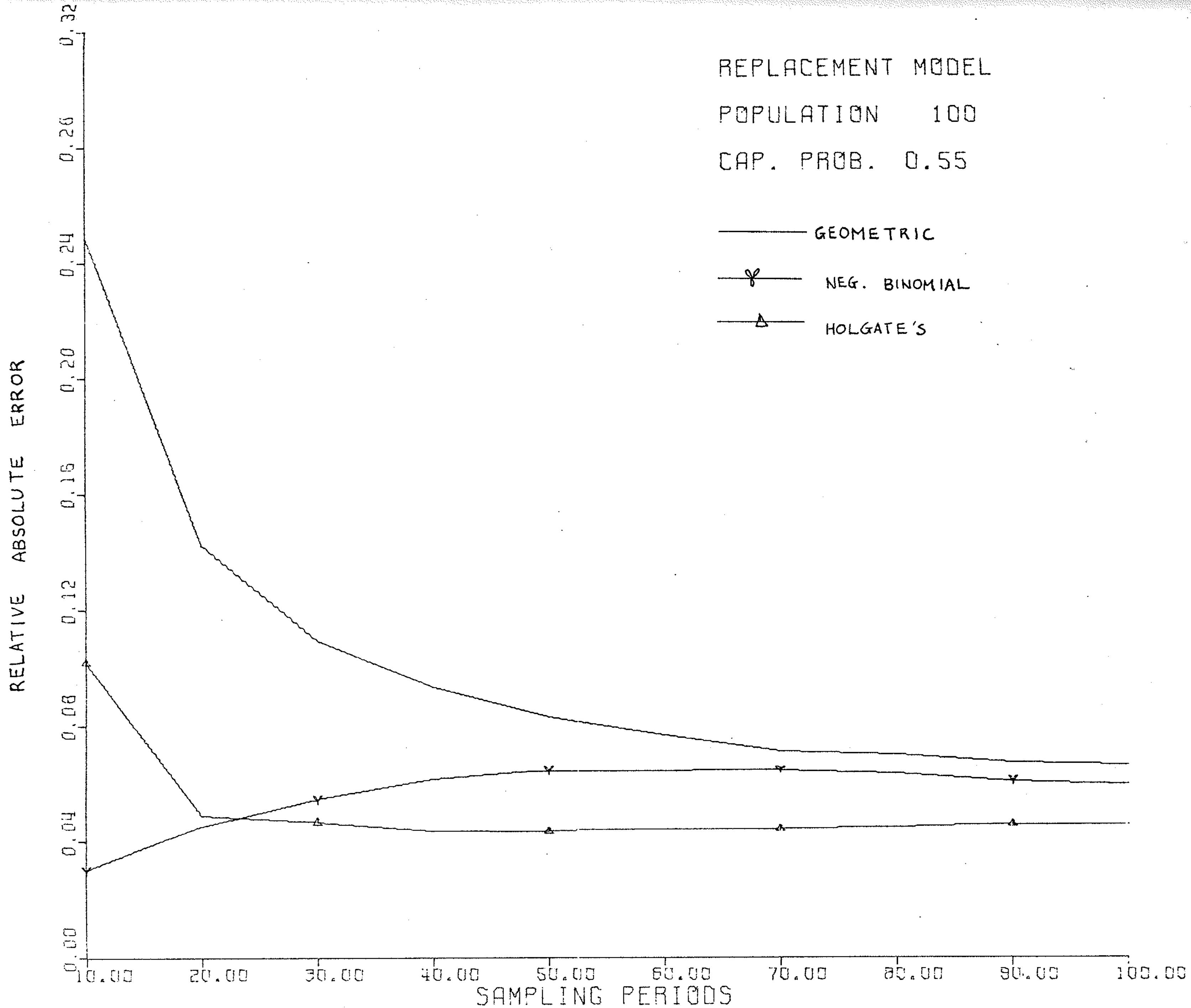

REPLACEMENT MODEL
POPULATION 100
CAP. PROB. 0.55
GEOMETRIC
NEG. BINOMIAL
HOLGATE'S
RELATIVE ABSOLUTE ERROR
0.00
0.04
0.08
0.12
0.16
0.20
0.24
0.28
0.32
10.00
20.00
30.00
40.00
50.00
60.00
70.00
80.00
90.00
100.00
SAMPLING PERIODS

FIGURE IV

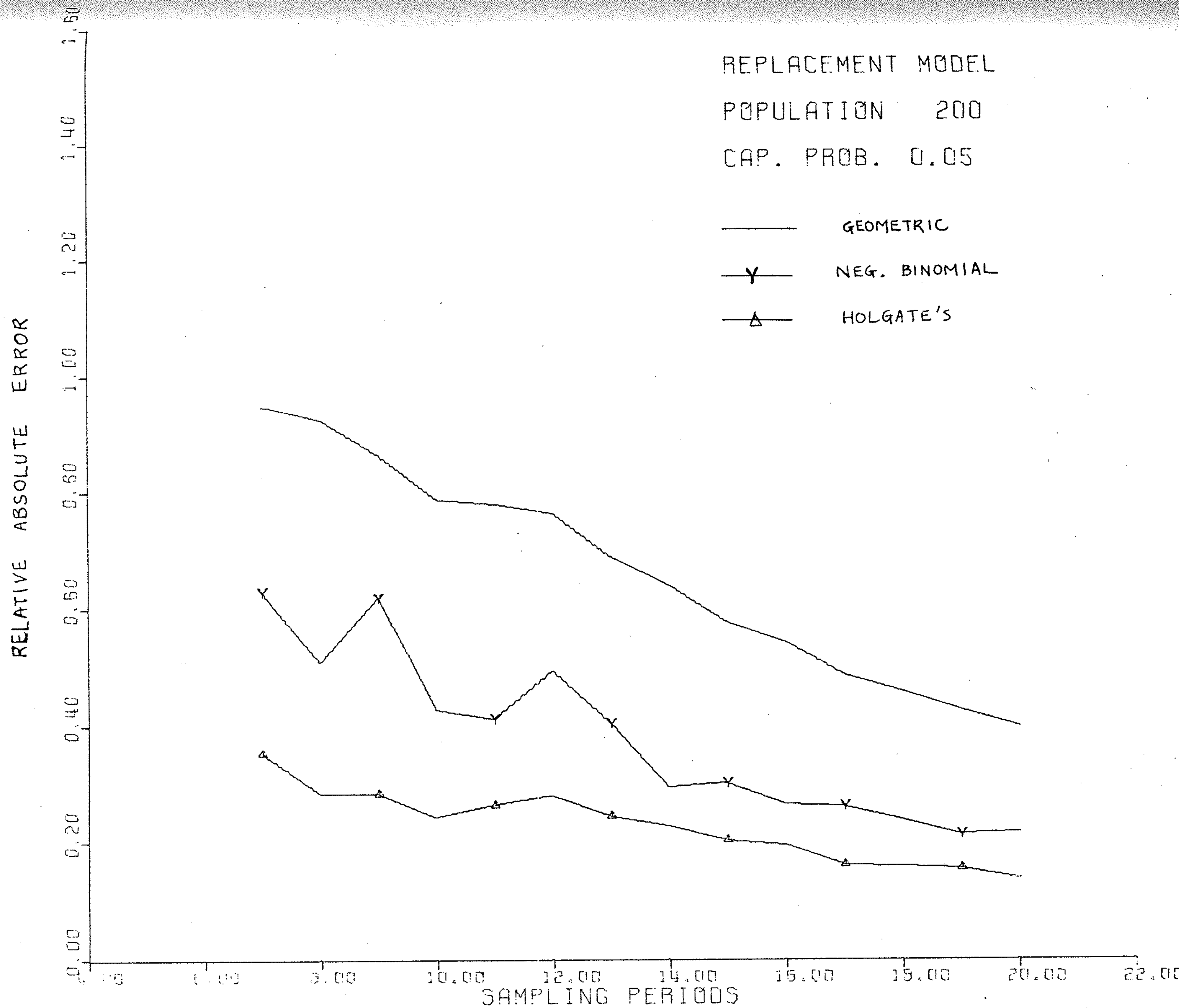
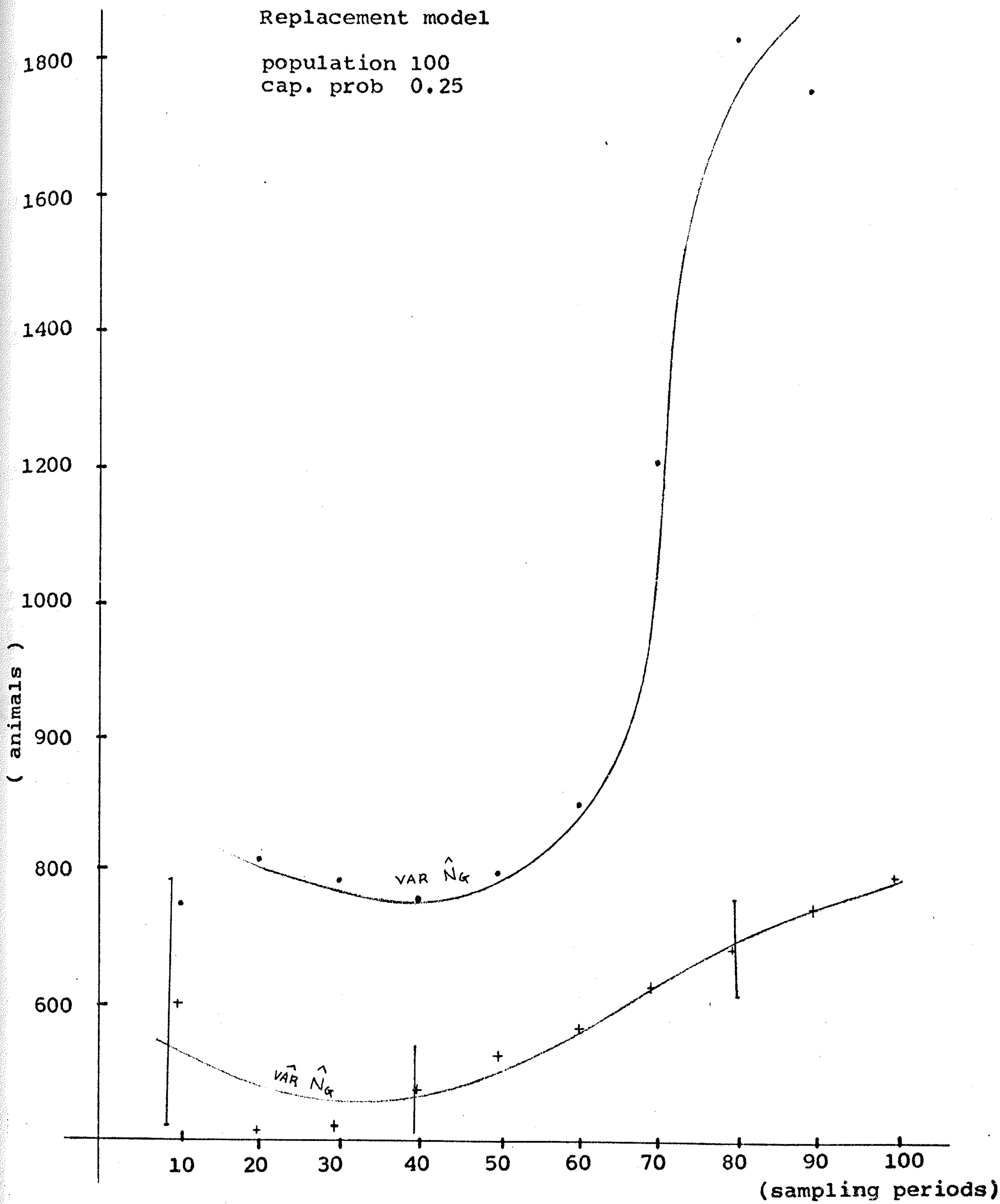

REPLACEMENT MODEL
POPULATION 200
CAP. PROB. 0.05
GEOMETRIC
NEG. BINOMIAL
HOLGATE'S
RELATIVE ABSOLUTE ERROR
SAMPLING PERIODS
1.60
1.40
1.20
1.00
0.80
0.60
0.40
0.20
0.00
8.00
10.00
12.00
14.00
16.00
20.00
22.00

FIGURE V



Replacement model
population 100
cap. prob 0.25
1800
1600
1400
1200
1000
900
800
600
(animals)
VAR $\hat{N}_G$
$\widehat{VAR}$ $\hat{N}_G$
10
20
30
40
50
60
70
80
90
100
(sampling periods)

FIGURE VI

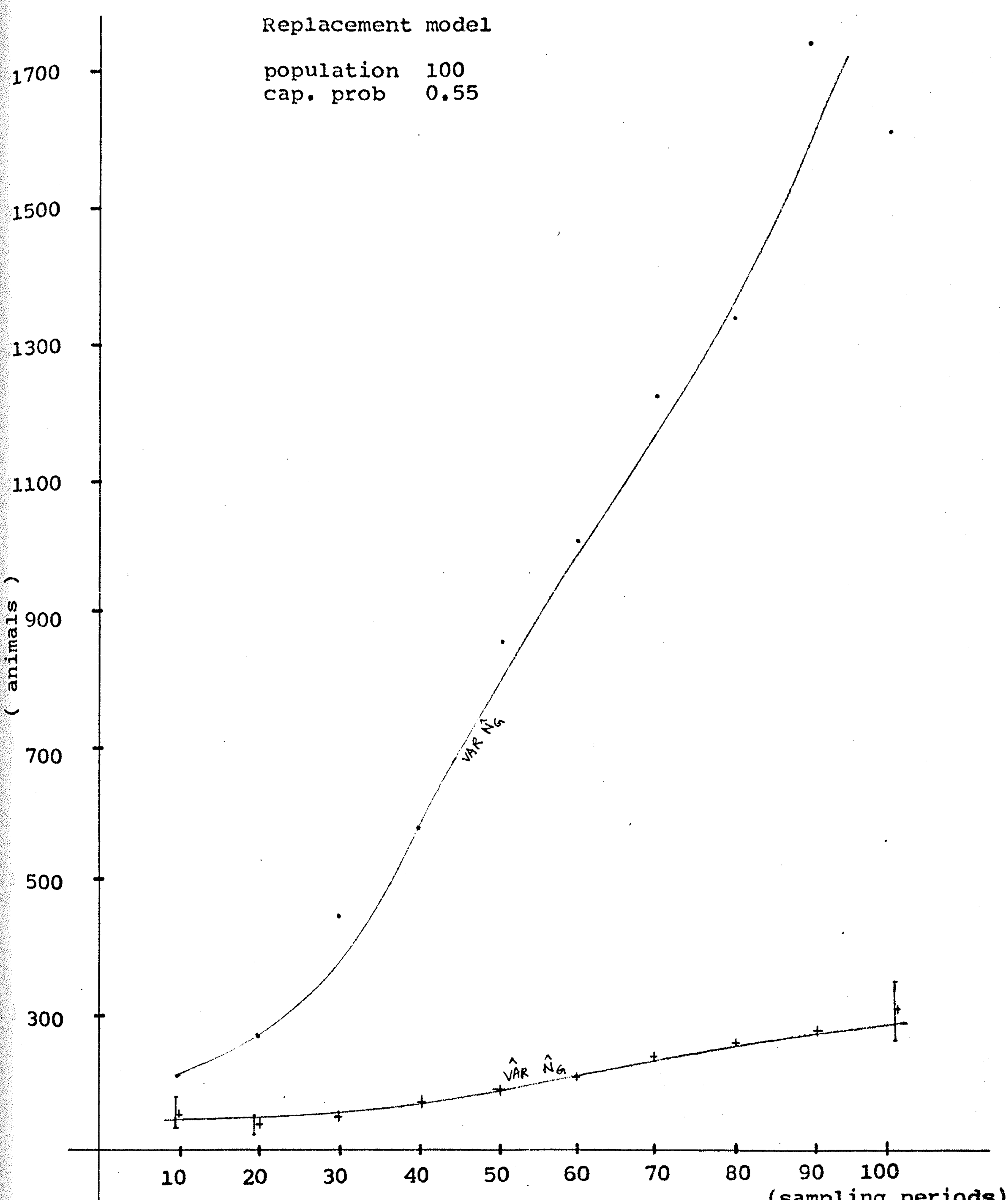

Replacement model
population 100
cap. prob 0.55
1700
1500
1300
1100
900
700
500
300
(animals)
VÂR N̂G
V̂AR N̂G
10
20
30
40
50
60
70
80
90
100
(sampling periods)

FIGURE VII

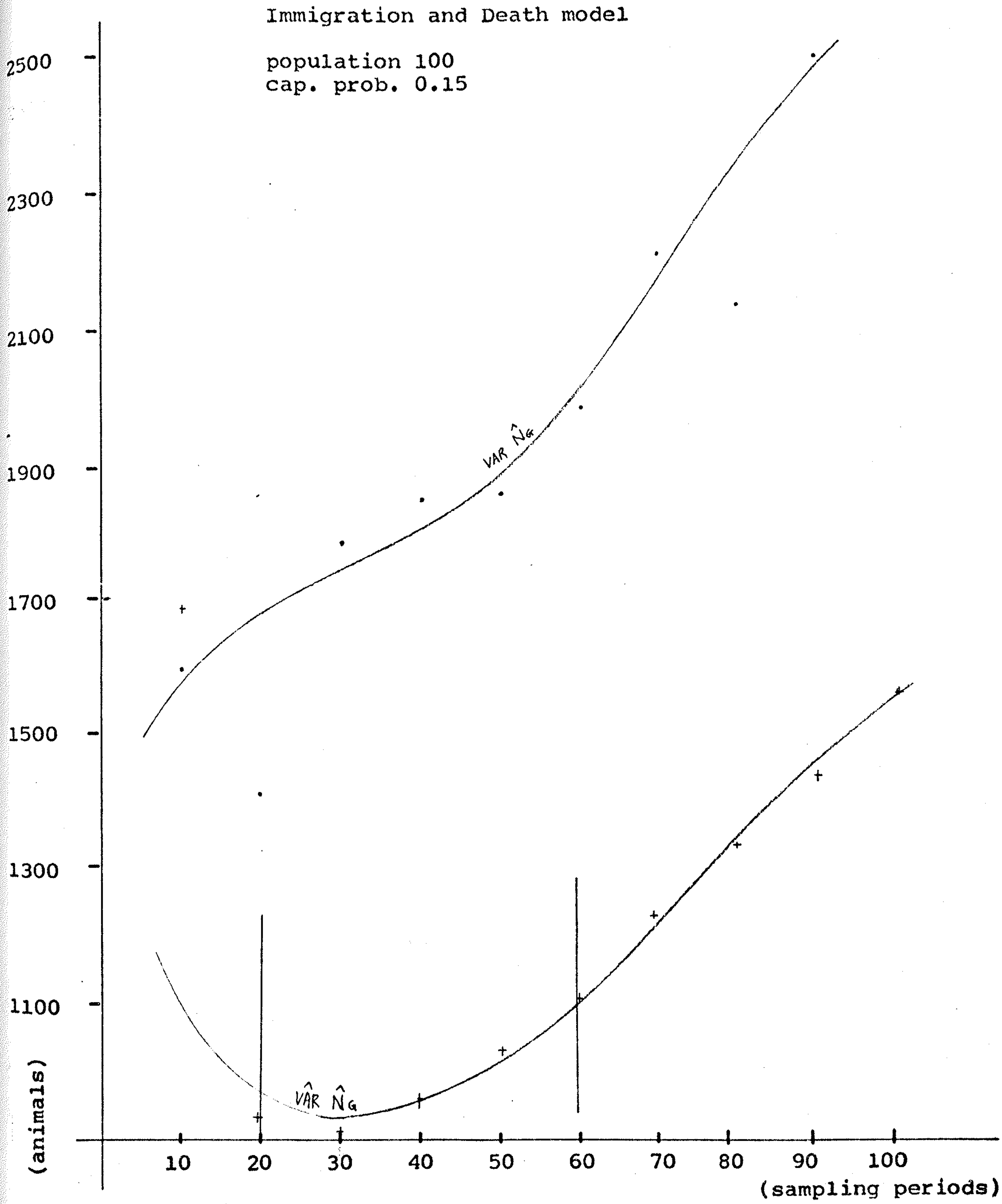

Immigration and Death model
population 100
cap. prob. 0.15
2500
2300
2100
1900
1700
1500
1300
1100
(animals)
10
20
30
40
50
60
70
80
90
100
(sampling periods)
VAR $\hat{N}_G$
$\hat{VAR}$ $\hat{N}_G$

FIGURE VIII

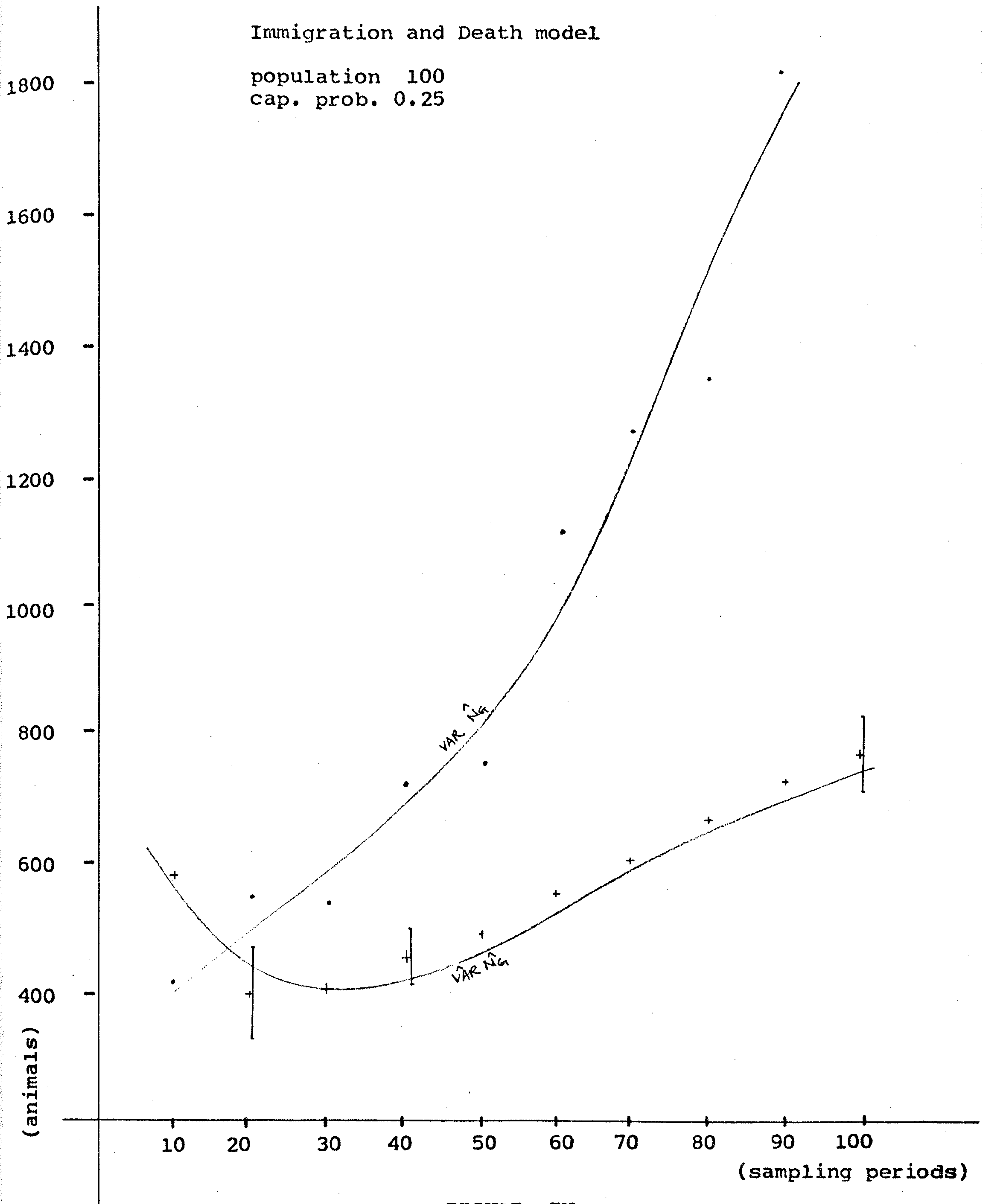
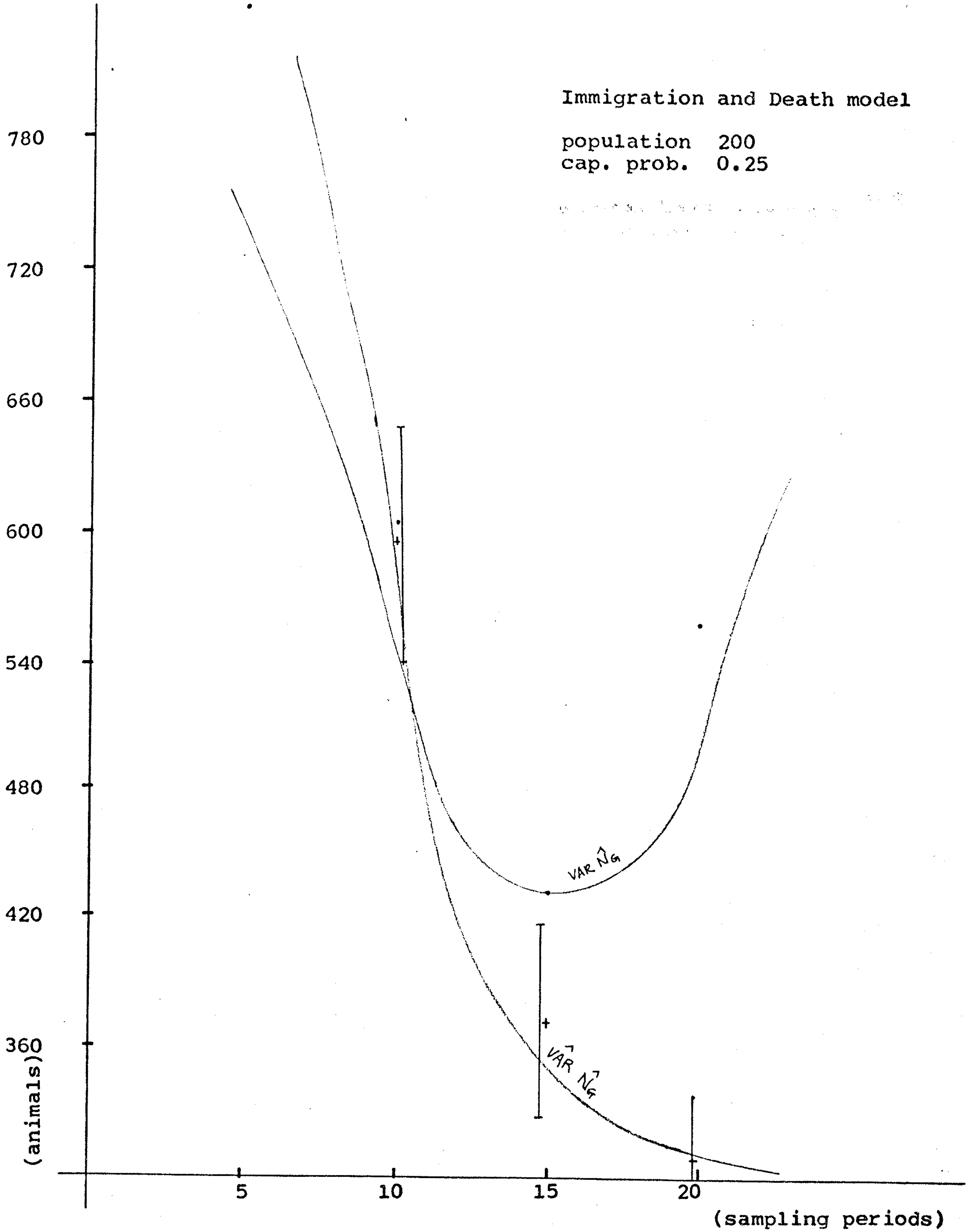

Immigration and Death model
population 100
cap. prob. 0.25
1800
1600
1400
1200
1000
800
600
400
(animals)
10
20
30
40
50
60
70
80
90
100
(sampling periods)
VAR N̂G
VÂR N̂G

FIGURE IX



Immigration and Death model
population 200
cap. prob. 0.25
780
720
660
600
540
480
420
360
(animals)
5
10
15
20
(sampling periods)
VAR $\hat{N}_G$
$\hat{VAR}$ $\hat{N}_G$

FIGURE X

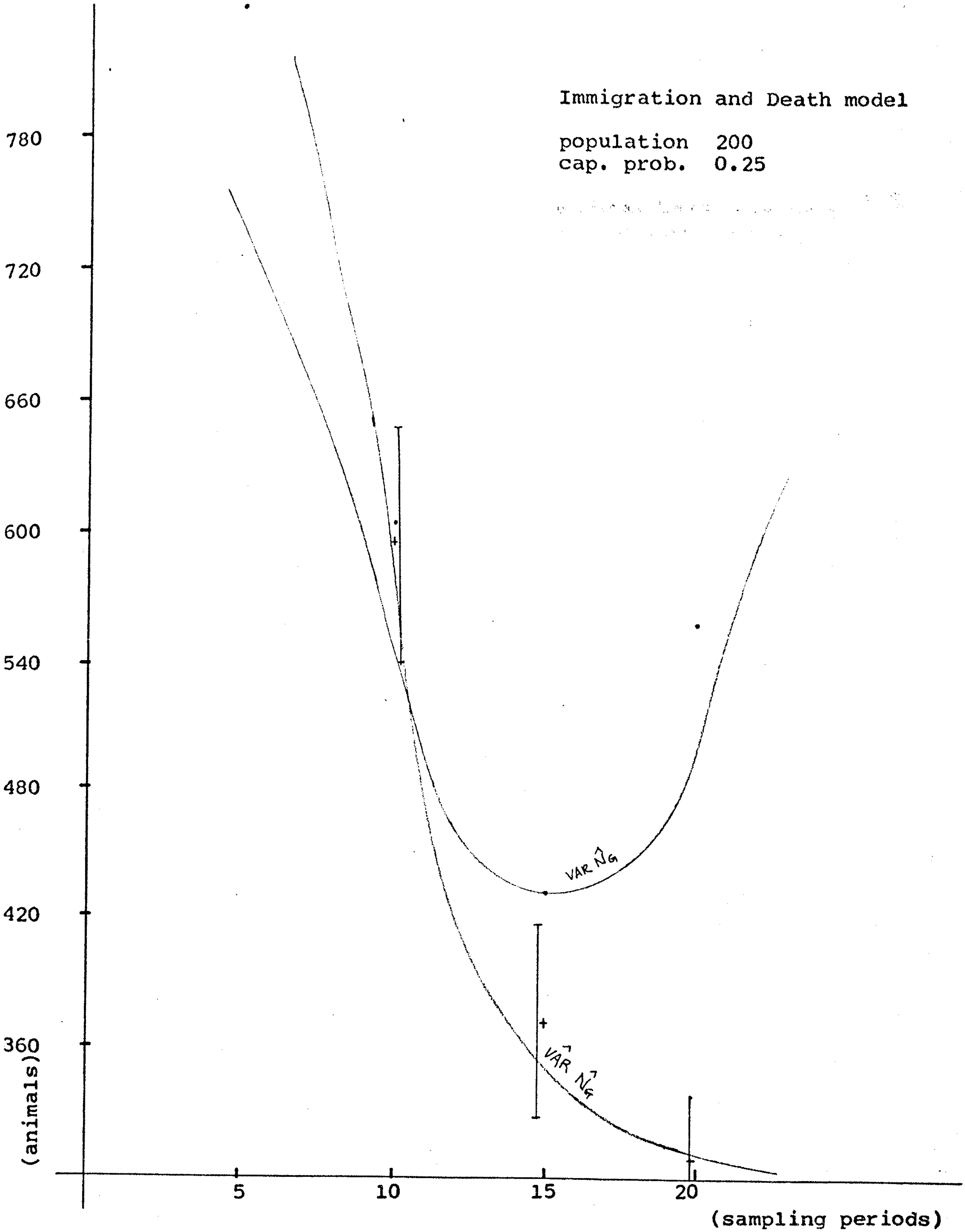

Immigration and Death model
population 200
cap. prob. 0.25
780
720
660
600
540
480
420
360
(animals)
5
10
15
20
(sampling periods)
VAR $\hat{N}_G$
VAR $\hat{N}_G$

FIGURE X

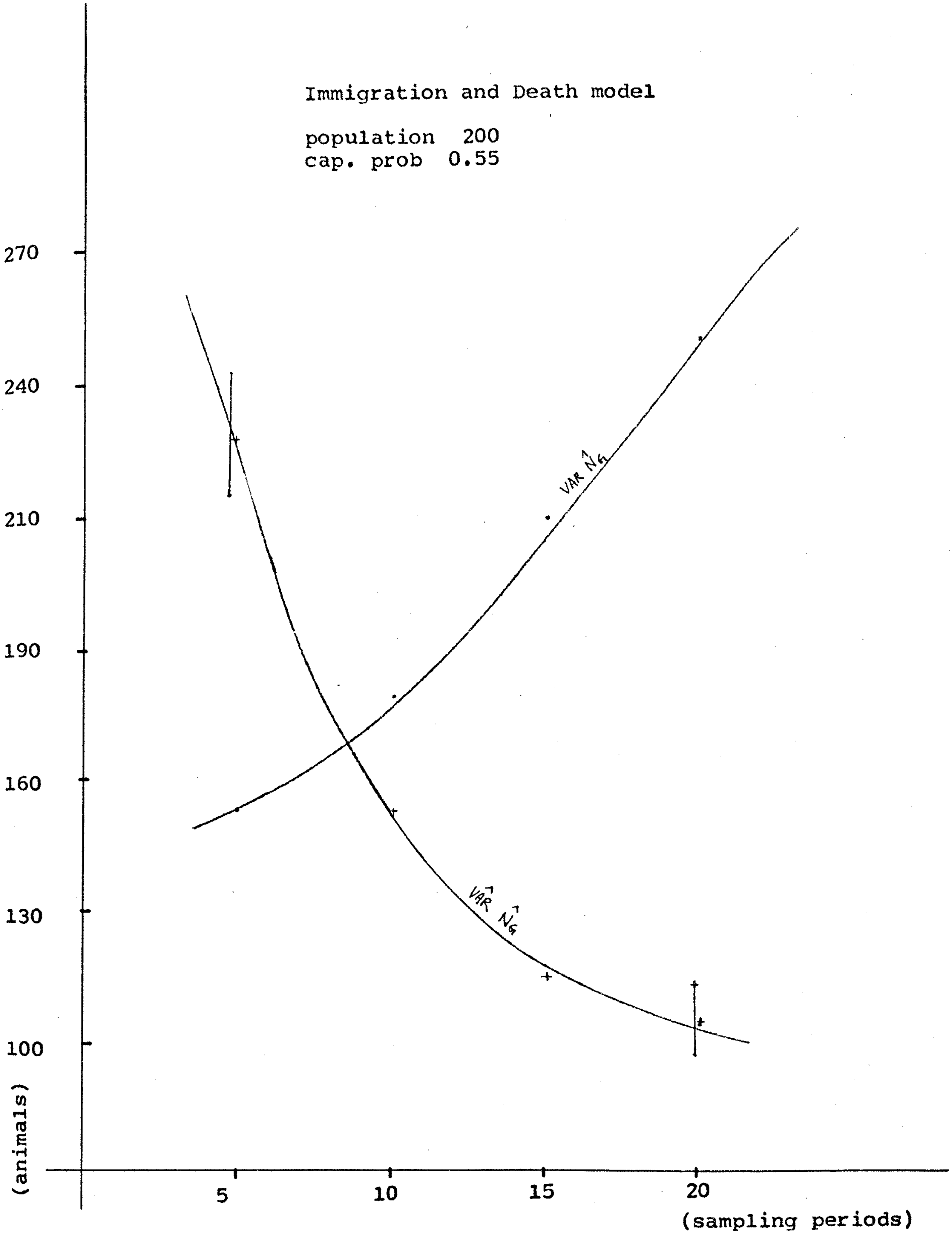

Immigration and Death model
population 200
cap. prob 0.55
270
240
210
190
160
130
100
(animals)
5
10
15
20
(sampling periods)
VAR $\hat{N}_G$
$\hat{VAR}$ $\hat{N_G}$

FIGURE XI

A P P E N D I X 1

The two programs were written in PL/1 and run on IBM 360/65 computer. The compile time for each of the two programs was about 30 seconds and the execution time was about 20 minutes for fifty simulations with capture probability from 0.05 to 0.95 in steps of 0.1

PROGRAM FOR THE REPLACEMENT MODEL

```
  LOC   OBJECT CODE      ADDR1 ADDR2  STMT    SOURCE STATEMENT

000000                                   1           CSECT
                                         2           ENTRY  RANDOM
000000                                   3           USING  *,15
000000 9025 D01C            0001C        4 RANDCM    STM    2,5,28(13)
000004 9823 1000            00000        5           LM     2,3,0(1)
000008 5850 F030            00030        6           L      5,A
00000C 5C42 0000            00000        7           M      4,0(2)
000010 5D40 F034            00034        8           D      4,P
000014 5042 0000            00000        9           ST     4,0(2)
000018 8840 C007            00007       10           SRL    4,7
00001C 5A40 F02C            0002C       11           A      4,CHAR
000020 5043 0000            00000       12           ST     4,0(3)
000024 9825 D01C            0001C       13           LM     2,5,28(13)
000028 07FE                             14           BR     14
00002A 0000
00002C 4C000000                         15 CHAR      DC     F'1073741824'
000030 000041A7                         16 A         DC     F'16807'
000034 7FFFFFFF                         17 P         DC     F'2147483647'
                                        18           END
```

SIM: PROCEDURE OPTIONS (MAIN);

```
SIM: PROCEDURE OPTIONS (MAIN);
     DCL (SAMPLE,N,POPSIZE,PER,TIME,TIM1,J,INT,PSPAN,JP)
            BINARY FIXED (31),
         (TSPAN,MEANL,CAPROB,CAPST,CAPEN,CAPIN,K,KK,REAL)
            BINARY FLOAT (31);
     DCL RANDOM ENTRY (BINARY FIXED (31),BINARY FLOAT (31));
     GET EDIT (N,SAMPLE,PER,POPSIZE,JP,CAPST,CAPEN,CAPIN,MEANL,INT)
             (5(F(5)),4(F(8,3)),F(7));
     PUT SKIP(4) EDIT ('POP SIZE =    ',POPSIZE) (X(20),A(13),F(5));
     TIME=(SAMPLE-JP)/PER+1;
     TIM1=TIME+1;
     MEANL=-MEANL;
     K=EXP(1/MEANL);
     KK=1-K;
     BEGIN;
       DCL (ANS,FSCAP,FNCAP,PHO) (N,TIME) BINARY FIXED (31),
           PTR (TIM1) BINARY FIXED (31),
           (TLEN,NG,NS,C,S,ZERO) BINARY FIXED (31),
           (TME,TVE,MB) (N,TIME) BINARY FLOAT (31),
           (ADAR,KAR,AVN,SDAA,AAXP) (N) BINARY FLOAT (31),
           (MU,VAR,MUO,VARO,KM,QM,PM,TEPK) BINARY FLOAT (31);
       PTR(1)=0;
       PTR(2)=JP;
       ZERO=JP;
       DO J=3 TO TIM1;
         ZERO=(J-2)*PER+ZERO+JP;
         PTR(J)=ZERO;
       END;
       TLEN=PTR(TIME)+SAMPLE;
       BEGIN;
         DCL (TGCAP,TSTAY,GCAP,ESTAY) (TLEN) BINARY FIXED (31),
             (SCAP,NCAP)  (SAMPLE) BINARY FIXED (31),
             AEXP (SAMPLE) BINARY FLOAT (31),
             (MCAR,NDAR,SCAR,DAR,VNG) (N) BINARY FIXED (31),
             (M,N1,J1,G,LAST,ESTZ,LLEN,J2) BINARY FIXED (31),
             (SPAN,P,Q,CHIV,STU,T1,TZ,AIN,PP,QQ) BINARY FLOAT (31);
     CA: DO CAPROB=CAPST TO CAPEN BY CAPIN;
             ANS,FSCAP,FNCAP=0;
             TGCAP,TSTAY=0;
         SIAM: DO NSIM=1 TO N;
                 GCAP,ESTAY=0;
                 SCAP,NCAP=0;
             PSZ: DO NAN=1 TO POPSIZE;
                   TSPAN=1;
                   PSPAN=0;
           AGAIN: CALL RANDOM(INT,REAL);
                   SPAN=MEANL*LOG(REAL);
                   IF PSPAN < JP THEN J=1;
                                 ELSE J=(PSPAN-JP+PER)/PER+1;
                   ANS(NSIM,J)=ANS(NSIM,J)+1;
                   TSPAN=TSPAN+SPAN;
                   IF TSPAN-PSPAN < 1 THEN GO TO AGAIN;
                   IF TSPAN > SAMPLE THEN TSPAN = SAMPLE;
                   CALL DONE(PSPAN,TSPAN,CAPROB,SCAP,NCAP,GCAP,ESTAY,
                         INT,PER,TIME,PTR,JP);
                   IF TSPAN ¬= SAMPLE THEN
                                       DO;
```

```
SIM: PROCEDURE OPTIONS (MAIN);

 46                                       PSPAN=FLOOR(TSPAN);
 47                                       GO TO AGAIN;
 48                                     END;
 49                       END PSZ;
 50                       TGCAP=TGCAP+GCAP;
 51                       TSTAY=TSTAY+ESTAY;
 52                       M,N1=0;
 53                       DO J=1 TO TIME;
 54                        IF J =1 THEN C=1; ELSE C=(J-2)*PER+1+JP;
 57                         S=PER*(J-1)+JP;
 58                         DO J1=C TO S;
 59                           M=SCAP(J1)+M;
 60                           N1=NCAP(J1)+N1;
 61                         END;
 62                         FSCAP(NSIM,J)=M;
 63                         FNCAP(NSIM,J)=N1;
 64                         MUO=FLOAT(M)/N1;
 65                         C=PTR(J);
 66                         S=PTR(J+1);
 67                         PHO(NSIM,J)=GCAP(C+1);
 68                         VAR,QQ=0;
 69                         DO G=C+1 TO S;
 70                           MU=(G-C)-MUO;
 71                           VAR=VAR+MU*MU*GCAP(G);
 72                           QQ=QQ+(G-C-1)*ESTAY(G);
 73                         END;
 74                         TME(NSIM,J)=MUO;
 75                         TVE(NSIM,J)=VAR/N1;
 76                         MB(NSIM,J)=QQ/N1;
 77                       END;
 78                     END SIAM;
 79                  MDAR=0;
 80                  DO J=1 TO TIME;
 81                    CALL PRTR(J,CAPROB,PER,N,JP);
 82                    PUT SKIP(5) EDIT ('TOTAL','MEAN','VAR.')
                                (X(71),A(5),2(X(14),A(4)));
 83                    NS,ZERO,S,C=0;
 84                    DO J1 =1 TO N;
 85                      MDAR(J1)=MDAR(J1)+ANS(J1,J);
 86                      NS=NS+MDAR(J1);
 87                      NDAR(J1)=FNCAP(J1,J);
 88                      S=S+NDAR(J1);
 89                      SDAR(J1)=FSCAP(J1,J);
 90                      C=C+SDAR(J1);
 91                      DAR(J1)=MDAR(J1)-NDAR(J1);
 92                      ZERO=ZERO+DAR(J1);
 93                    END;
 94                    CALL MV(MDAR,N,MU,VAR);
 95                    PUT SKIP(2) EDIT ('POPUL. ( TRUE )',NS,MU,VAR)
                                (X(20),A(15),X(29),F(12),2(X(6),F(12,4)));
 96                    CALL MV(DAR,N,MU,VAR);
 97                    PUT SKIP(2) EDIT ('ZERO ( TRUE )',ZERO,MU,VAR)
                                (X(20),A(13),X(31),F(12),2(X(6),F(12,4)));
 98                    CALL MV(NCAR,N,MU,VAR);
 99                    PUT SKIP(2) EDIT ('INDIVIDUALS',S,MU,VAR)
                                (X(20),A(11),X(33),F(12),2(X(6),F(12,4)));
100                    CALL MV(SDAR,N,MU,VAR);
```

```
101          PUT SKIP(2) EDIT ('CAPTURES',C,MU,VAR)
                         (X(20),A(8),X(36),F(12),2(X(6),F(12,4)));
102          PUT PAGE;
103          PUT SKIP(5) EDIT ('OBSERVED RECAPTURE FQ.')
                         (X(5),A(24));
104          M=PTR(J)+1;
105          N1=PTR(J+1);
106          LLEN=N1-M+1;
1C7          DO J1=M TO N1;
108            IF TGCAP(J1) > 0 THEN LAST=J1;
110          END;
111          PUT SKIP(3) EDIT ('(',ZERO,')')
                         (X(4),A(1),F(8),X(1),A(1));
112          PUT SKIP(2) EDIT ((TGCAP(J1) DO J1=M TO LAST))
                         (10(X(4),F(8)));
113          STU=FLOAT(S);
114          NG=STU*(C-1)/(C-STU);
115          P=STU/C;
116          Q=1-P;
117          ESTZ=NG-S;
118          MUO=C/STU;
119          VARO=0;
120          DO J1=1 TO LAST-M+1;
121            AIN=J1-MUO;
122            VARO=VARO+AIN*AIN*TGCAP(M+J1-1);
123          END;
124          VARO=VARO/STU;
125          PUT SKIP(4) EDIT ('MEAN','VAR.') (X(70),2(X(10),A(4)));
126          PUT SKIP(2) EDIT ('ZERO CLASS EXCLUDED',MUO,VARO)
                         (X(30),A(19),X(21),2(X(6),F(8,4)));
127          CALL MVO (TGCAP,M,N1,C,ESTZ,MUO,MU,VAR);
128          PUT SKIP(2) EDIT ('ZERO CLASS INCLUDED',MU,VAR)
                         (X(30),A(19),X(21),2(X(6),F(8,4)));
129          PUT SKIP(4) EDIT ('MAXIMUM LKD. FIT') (X(5),A(17));
130          PUT SKIP(3) EDIT ('(',ESTZ,')')
                         (X(4),A(1),F(8),X(1),A(1));
131          LAST=LLEN;
132          AIN=P*NG;
133          DO J2=1 TO LLEN;
134            AIN=AIN*Q;
135            IF AIN >= 1.0 THEN AEXP(J2)=AIN;
137                         ELSE
137                         DO;
138                           LAST=J2-1;
139                           GO TO MUM;
140                         END;
141          END;
142      MUM: PUT SKIP(2) EDIT ((AEXP(J1) DO J1=1 TO LAST))
                           (10(X(4),F(8,1)));
143          CALL CHITEST (TGCAP,AEXP,LAST,CHIV,N1,M,G);
144          PUT SKIP(2) EDIT ('P  =   ',P) (X(5),A(6),X(8),F(8,2));
145          PUT SKIP(2) EDIT ('Q  =   ',Q) (X(5),A(6),X(8),F(8,2));
146          PUT SKIP(4) EDIT ('CHI VALUE IS  ',CHIV,G)
                    (X(5),A(14),F(8,2),X(10),F(3));
147          PUT SKIP(4) EDIT ('MEAN','VAR.') (X(60),2(X(14),A(4)));
148          DO J1=1 TO N;
149            J2=SDAR(J1);
```

```
150        STU=FLOAT(NDAR(J1));
151        DAR(J1)=STU*(J2-1)/(J2-STU);
152        VNG(J1)=DAR(J1)*(STU*J2-J2+STU)/(J2-STU)**2;
153        SDAR(J1)=DAR(J1)-MDAR(J1);
154        AAXP(J1)=FLOAT(SDAR(J1))/MDAR(J1);
155      END;
156      Q=1-CAPROB;
157      PP=KK/(1-Q*K);
158      QQ=(1-Q)*K/(1-Q*K);
159      CALL MV (DAR,N,MU,VAR);
160      PUT SKIP(2) EDIT ('NG',MU,VAR)
                        (X(38),A(2),X(20),2(X(6),F(12,4)));
161      CALL MV (SDAR,N,MU,VAR);
162      PUT SKIP(2) EDIT ('NG-N',MU,VAR)
                        (X(38),A(4),X(18),2(X(6),F(12,4)));
163      CALL MV (VNG,N,MU,VAR);
164      PUT SKIP(2) EDIT ('VAR. NG',MU,VAR)
                        (X(38),A(7),X(15),2(X(6),F(12,4)));
165      CALL MIV (AAXP,N,MU,VAR);
166      PUT SKIP(2) EDIT ('(NG-N)/N',MU,VAR)
                        (X(38),A(8),X(14),2(X(6),F(12,4)));
167      CALL THEO (AEXP,S,PP,QQ,LLEN,LAST);
168      PUT SKIP(4) EDIT ('THEORETICAL FIT') (X(5),A(16));
169      PUT SKIP(2) EDIT ((AEXP(J1) DO J1=1 TO LAST))
                        (10(X(4),F(8,1)));
170      CALL CHITEST(TGCAP,AEXP,LAST,CHIV,N1,M,G);
171      PUT SKIP(2) EDIT ('CHI VALUE IS  ',CHIV,G) (X(5),A(14),
                                      F(8,2),X(10),F(3));
172      PUT PAGE;
173      LAST=M+1;
174      DO J1=M TO N1;
175        IF TSTAY(J1) > 0 THEN LAST=J1;
177      END;
178      TZ=FLOAT(TGCAP(M))/S;
179      T1=1-TZ;
180      VAR,AIN=0;
181      DO J1=M+1 TO LAST;
182        AIN=AIN+(J1-M)*TSTAY(J1);
183      END;
184      AIN=AIN/S;
185      DO J1=M TO LAST;
186        MU=(J1-M)-AIN;
187        VAR=VAR+MU*MU*TSTAY(J1);
188      END;
189      VAR=VAR/S;
190      PUT SKIP(5) EDIT ('OBSERVED DURATION FREQUENCY ',
       '  (TOTAL) ','MEAN = ',AIN,'VAR. = ',VAR)
             (X(5),A(29),A(10),X(5),2(X(5),A(7),X(2),F(9,6)));
191      PUT SKIP(3) EDIT ((TSTAY(J1) DO J1=M TO LAST))
                        (10(X(4),F(8)));
192       KM=1-(T1/AIN);
193       QM=(KM-T1)/(KM*TZ);
194       PM=(1-KM)/(1-QM*KM);
195       CALL HOL (AEXP,QM,KM,PM,LLEN,LAST,S);
196      PUT SKIP(4) EDIT ('MAXIMUM LKD FIT')
                     (X(5),A(15));
197      PUT SKIP(2) EDIT ('Q =   ',QM) (X(12),A(5),F(8,6));
```

```
PUT SKIP(2) EDIT ('K =   ',KM) (X(12),A(5),F(8,6));
PUT SKIP(3) EDIT ((AEXP(J1) DO J1=1 TO LAST))
                  (10(X(4),F(8,1)));
TEPK=0;
DO J1=1 TO LAST;
  TEPK=TEPK+AEXP(J1);
END;
IF S > TEPK THEN TEPK=S-TEPK;
            ELSE TEPK=0;
PUT SKIP(2) EDIT ('REMAINDER',TEPK)
        (X(10),A(9),X(5),F(8,1));
CALL CHITEST(TSTAY,AEXP,LAST,CHIV,N1,M,G);
PUT SKIP(2) EDIT ('CHI VALUE IS   ',CHIV,G) (X(5),A(14),
                              F(8,2),X(10),F(3));
G=0;
DO J1=1 TO N;
  STU=NDAR(J1);
  TZ=PHO(J1,J)/STU;
  T1=1-TZ;
  AIN=TME(J1,J)-(TVE(J1,J)*TZ)/(TME(J1,J)*T1);
  IF AIN <= 0.0 THEN DO;
                       G=G+1;
                       AVN(J1)=0;
                     END;
                ELSE AVN(J1)=FSCAP(J1,J)/AIN;
  TEPK=1-(T1/MB(J1,J));
  ADAR(J1)=STU/T1*TEPK;
  SDAA(J1)=ADAR(J1)-MDAR(J1);
  AAXP(J1)=SDAA(J1)/MDAR(J1);
  KAR(J1)=TEPK;
END;
PUT SKIP(4) EDIT ('MEAN','VAR.','S.D.')
                                  (X(60),3(X(14),A(4)));
CALL MIV (ADAR,N,MU,VAR);
PUT SKIP(2) EDIT ('NH',MU,VAR)
                 (X(38),A(2),X(20),2(X(6),F(12,4)));
CALL MIV (SDAA,N,MU,VAR);
PUT SKIP(2) EDIT ('NH-N',MU,VAR)
                 (X(38),A(4),X(18),2(X(6),F(12,4)));
CALL MIV (AAXP,N,MU,VAR);
PUT SKIP(2) EDIT ('(NH-N)/N',MU,VAR)
                 (X(38),A(8),X(14),2(X(6),F(12,4)));
DO J1=1 TO N;
 ADAR(J1)=AVN(J1)-MDAR(J1);
 AAXP(J1)=ADAR(J1)/MDAR(J1);
END;
CALL MIV (AVN,N,MU,VAR);
PUT SKIP(2) EDIT ('NB',MU,VAR,'REJECT',G,'/',N)
                 (X(38),A(2),X(20),2(X(6),F(12,4)),
                 X(5),A(6),X(1),F(3),X(2),A(1),X(1),F(3));
CALL MIV (ADAR,N,MU,VAR);
PUT SKIP(2) EDIT ('NB-N',MU,VAR)
                 (X(38),A(4),X(18),2(X(6),F(12,4)));
CALL MIV (AAXP,N,MU,VAR);
PUT SKIP(2) EDIT ('(NB-N)/N',MU,VAR)
                 (X(38),A(8),X(14),2(X(6),F(12,4)));
MU=KK*KK*PP;
```

```
                    DO J1=1 TO N;
                      ADAR(J1)=KAR(J1)-K;
                      AAXP(J1)=ADAR(J1)/K;
                      SDAA(J1)=SQRT(MU/NDAR(J1));
                    END;
                    CALL MIV(KAR,N,MU,VAR);
                    MUO=SQRT(VAR);
                    PUT SKIP(2) EDIT ('KH',MU,VAR,MUO)
                                    (X(38),A(2),X(20),3(X(6),F(12,4)));
                    CALL MIV (ADAR,N,MU,VAR);
                    MUO=SQRT(VAR);
                    PUT SKIP(2) EDIT ('KH-K',MU,VAR,MUO)
                                    (X(38),A(4),X(18),3(X(6),F(12,4)));
                    CALL MIV (AAXP,N,MU,VAR);
                    MUO=SQRT(VAR);
                    PUT SKIP(2) EDIT ('(KH-K/K)',MU,VAR,MUO)
                                    (X(38),A(8),X(14),3(X(6),F(12,4)));
                     CALL MIV(SDAA,N,MU,VAR);
                    PUT SKIP(2) EDIT ('S. D. OF K',MU,VAR)
                                  (X(38),A(10),X(12),3(X(6),F(12,9)));
                     CALL HOL(AEXP,Q,K,PP,LLEN,LAST,S);
                    PUT PAGE;
                    PUT SKIP(10) EDIT ('THEORETICAL FIT')
                                    (X(5),A(15));
                    PUT SKIP(2) EDIT ('K =   ',K) (X(12),A(5),F(8,6));
                    PUT SKIP(3) EDIT ((AEXP(J1) DO J1=1 TO LAST))
                                    (10(X(4),F(8,1)));
                    TEPK=0;
                    DO J1=1 TO LAST;
                      TEPK=TEPK+AEXP(J1);
                    END;
                    IF S > TEPK THEN TEPK=S-TEPK;
                               ELSE TEPK=0;
                    PUT SKIP(2) EDIT ('REMAINDER',TEPK)
                              (X(10),A(9),X(5),F(8,1));
                    CALL CHITEST(TSTAY,AEXP,LAST,CHIV,N1,M,G);
                    PUT SKIP(2) EDIT ('CHI VALUE IS   ',CHIV,G) (X(5),A(14),
                                                  F(8,2),X(10),F(3));
         END;
         END;
         END;
         END;
         DONE: PROCEDURE (INI,SPAN,CAPROB,SCAP,NCAP,GCAP,ESTAY,INT,PER,PERI,
                        PTR,JP);
                DCL (SCAP(*),NCAP(*),GCAP(*),ESTAY(*),MARK,INT,LCAP,FCAP,
                      PER,INI,PERI,M,JJ,JK,IBI,JM,PTR(*),J,JZ,JP)
                        BINARY FIXED (31),
                    (REAL,CAPROB,SPAN) BINARY FLOAT (31);
                DCL RANDOM ENTRY (BINARY FIXED (31),BINARY FLOAT (31));
                MARK=0;
                DO J=INI+1 TO SPAN;
                  CALL RANDOM (INT,REAL);
                  IF REAL < CAPROB THEN
                  DO;
                    IF MARK=0 THEN
                      DO;
                        NCAP(J)=NCAP(J)+1;
```

```
292                          FCAP=J;
293                          IF J <= JP THEN JJ=1;
295                                     ELSE JJ=(J-JP+PER-1)/PER+1;
296                        END;
297                        ELSE
297                        DO;
298                          IF J <= JP THEN JK=1;
300                                     ELSE JK=(J-JP+PER-1)/PER+1;
301                          IF JK > JJ THEN
302                            DO;
303                              M=LCAP-FCAP+1;
304                              DO JZ=JJ TO JK-1;
305                                JM=PTR(JZ);
306                                GCAP(JM+MARK)=GCAP(JM+MARK)+1;
307                                ESTAY(JM+M)=ESTAY(JM+M)+1;
308                              END;
309                              JJ=JK;
310                            END;
311                        END;
312                      MARK=MARK+1;
313                      SCAP(J)=SCAP(J)+1;
314                      LCAP=J;
315                    END;
316                  END;
317                  IF MARK > 0 THEN
318                    DO;
319                      IF LCAP <= JP THEN JK=1;
321                                    ELSE JK=(LCAP-JP+PER-1)/PER+1;
322                      M=LCAP-FCAP+1;
323                      DO JZ=JK TO PERI;
324                        JM=PTR(JZ);
325                        GCAP(JM+MARK)=GCAP(JM+MARK)+1;
326                        ESTAY(JM+M)=ESTAY(JM+M)+1;
327                      END;
328                    END;
329                END DONE;
330          MVO: PROCEDURE (ARA,ST,LEN,S,Z,MU,MUO,VAR);
331                DCL (ARA(*),ST,LEN,Z,S,J) BINARY FIXED (31),
                       (MU,MUO,VAR,VAL,INI,AIN) BINARY FLOAT (31);
332                INI=S+Z;
333                MUO=MU*S/INI;
334                VAR=MUO*MUO*Z;
335                AIN=0;
336                DO J=ST TO LEN;
337                  AIN=AIN+1;
338                  VAL=AIN-MUO;
339                  VAR=VAR+VAL*VAL*ARA(J);
340                END;
341                VAR=VAR/INI;
342              END MVO;
343          CHITEST: PROCEDURE (OBS,EXPECT,LAST,VALUE,SAMI,FST,M);
344                  DCL (OBS(*),LAST,SAMI,FST,J,JJ,M,PT,FT) BINARY FIXED (31),
                         (EXPECT(*),VALUE,IDIF,INT) BINARY FLOAT (31);
345                  FT=FST-1;
346                  M=LAST;
347                  DO J=1 TO LAST;
348                    IF EXPECT(J) < 5 THEN
```

```
                        DO;
                          M=J;  INT=0;
                          DO JJ=M TO LAST;
                             INT=INT+EXPECT(JJ);
                          END;
                        IF INT < 5 THEN
                          DO;
                             M=M-1;
                             EXPECT(M)=EXPECT(M)+INT;
                          END;
                          ELSE EXPECT(M)=INT;
                          GO TO BG;
                        END;
                   END;
              BG:  PT=M+FT;
                   INT=0;
                   DO JJ=PT TO SAMI;
                   INT=INT+OBS(JJ);
                   END;
              AG:  IF INT < 5 THEN
                             DO;
                               PT=PT-1;
                               INT=INT+OBS(PT);
                               M=M-1;
                               EXPECT(M)=EXPECT(M)+EXPECT(M+1);
                               GO TO AG;
                             END;
                   OBS(PT)=INT;
                   VALUE=0;
                   DO J=1 TO M;
                      IDIF=OBS(J+FT)-EXPECT(J);
                      IF IDIF¬= 0 THEN VALUE=VALUE+(IDIF*IDIF)/EXPECT(J);
                   END;
                   END CHITEST;
        THEO: PROCEDURE (ARY,NUM,PP,QQ,LEN,N);
                  DCL (ARY(*),PP,QQ,VAL) BINARY FLOAT (31),
                      (NUM,LEN,N,J) BINARY FIXED (31);
                  ARY(1),VAL=NUM*PP;
                  N=LEN;
                  DO J=2 TO LEN;
                     VAL=VAL*QQ;
                     IF VAL < 1.0 THEN
                        DO;
                          N=J-1;
                          GO TO FINI;
                        END;
                     ARY(J)=VAL;
                  END;
        FINI: END THEO;
        HOL: PROCEDURE (ARY,Q,K,KK,LEN,LAST,N);
                 DCL (ARY(*),Q,K,KK,AIN) BINARY FLOAT (31),
                      (J,LEN,LAST,N) BINARY FIXED (31);
                 LAST=LEN;
                 ARY(1)=KK*N;
                 ARY(2)=ARY(1)*(1-Q)*K;
                 DO J=3 TO LEN;
                    AIN=ARY(J-1)*K;
```

```
406                IF AIN < 1.0 THEN
407                             DC;
408                               LAST=J-1;
409                               GO TO FINE;
410                             END;
411                ARY(J)=AIN;
412              END;
413        FINE: END HOL;
414       MV: PROCEDURE (ARY,LEN,MEAN,VAR);
415            DCL (ARY(*),LEN,J) BINARY FIXED (31),
                    AY2 (LEN) BINARY FIXED (31),
                    (MEAN,VAR,TEM,TM) BINARY FLOAT (31);
416            MEAN,VAR=0;
417            TEM=ARY(1);
418            DO J=1 TO LEN;
419              TM=ARY(J)-TEM;
420              AY2(J)=TM;
421              MEAN=MEAN+TM;
422            END;
423            MEAN=MEAN/LEN;
424            DO J=1 TO LEN;
425              TM=AY2(J)-MEAN;
426              VAR=VAR+TM*TM;
427            END;
428            MEAN=MEAN+TEM;
429            VAR=VAR/LEN;
430          END MV;
431       MIV: PROCEDURE (ARY,LEN,MU,VAR);
432              DCL (ARY(*),AIN,VAR,MU,TM,TEM) BINARY FLOAT (31),
                      YY (LEN) BINARY FLOAT (31),
                      (J,LEN) BINARY FIXED (31);
433              AIN,VAR=0;
434              TEM=ARY(1);
435              DO J=1 TO LEN;
436                TM=ARY(J)-TEM;
437                YY(J)=TM;
438                AIN=AIN+TM;
439              END;
440              MU=AIN/LEN;
441              DO J=1 TO LEN;
442                AIN=YY(J)-MU;
443                VAR=VAR+AIN*AIN;
444              END;
445              MU=MU+TEM;
446              VAR=VAR/LEN;
447            END MIV;
448       PRTR: PROCEDURE (I,CA,SAM,N,JP);
449               DCL (I,NUM,SAM,N,JP) BINARY FIXED (31),
                      CA BINARY FLOAT (31);
450               NUM=(I-1)*SAM+JP;
451               PUT PAGE;
452               PUT LINE (10) EDIT ('MODEL') (X(64),A(5));
453               PUT SKIP(0) EDIT ('_____') (X(64),A(5));
454               PUT SKIP(3) EDIT ('   REPLACEMENT   ') (X(58),A(17));
455               PUT SKIP(3) EDIT ('PARAMETERS') (X(62),A(10));
456               PUT SKIP(0) EDIT ('__________') (X(62),A(10));
457               PUT SKIP(3) EDIT ('NO. OF SAMPLING PERIODS  : ',NUM)
```

SIM: PROCEDURE OPTIONS (MAIN);

```
                                        (X(51),A(27),F(4));
458                     PUT SKIP(2) EDIT ('NO. OF SIMULATIONS          : ',N)
                                        (X(51),A(27),F(4));
459                     PUT SKIP(2) EDIT ('CAPTURE PROBABILITY         : ',CA)
                                        (X(51),A(27),F(4,2));
460                 END PRTR;
461          END;
```

PROGRAM FOR THE IMMIGRATION

AND DEATH MODEL

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT

000000                               1           CSECT
                                     2           ENTRY  RANDOM
000000                               3           USING  *,15
000000 9025 D01C            0001C    4 RANDCM    STM    2,5,28(13)
000004 9823 1000            00000    5           LM     2,3,0(1)
000008 5850 F030            00030    6           L      5,A
00000C 5C42 0000            00000    7           M      4,0(2)
000010 5D40 F034            00034    8           D      4,P
000014 5042 0000            00000    9           ST     4,0(2)
000018 8840 0007            00007   10           SRL    4,7
00001C 5A40 F02C            0002C   11           A      4,CHAR
000020 5043 0000            00000   12           ST     4,0(3)
000024 9825 D01C            0001C   13           LM     2,5,28(13)
000028 07FE                         14           BR     14
00002A 0000
00002C 4CC0C000                     15 CHAR      DC     F'1073741824'
000030 000041A7                     16 A         DC     F'16807'
000034 7FFFFFFF                     17 P         DC     F'2147483647'
                                    18           END
```

```
MUS: PROCEDURE OPTIONS (MAIN);

 1            MUS: PROCEDURE OPTIONS (MAIN);
 2                 DCL (N,SAMPLE,POPSIZE,INT,POPUL,PERIOD,INI,TIME,TIM1,NSIM,J,J1,JP)
                     BINARY FIXED (31),
                       (CAPROB,CAPST,CAPEN,CAPIN,MEANL,ARRIV,K,KK,REAL,VALUE,TEPK)
                           BINARY FLOAT (31);
 3                 DCL RANDOM ENTRY (BINARY FIXED (31),BINARY FLOAT (31));
 4                 GET EDIT (N,SAMPLE,POPSIZE,JP,PERIOD,INI,MEANL,ARRIV,CAPST,CAPEN,
                            CAPIN,INT) (6(F(5)),5(F(8,3)),F(7));
 5                 PUT SKIP(4) EDIT ('POP SIZE =    ',POPSIZE) (X(20),A(13),F(5));
 6                 TIME=(SAMPLE-JP)/PERIOD+1;
 7                 TIM1=TIME+1;
 8                 MEANL=-MEANL;
 9                 ARRIV=-1/ARRIV;
10                 K=EXP(1/MEANL);
11                 KK=1-K;
12                 BEGIN;
13                   DCL (ANS,FSCAP,FNCAP,PHO) (N,TIME) BINARY FIXED (31),
                         (TME,TVE,MB) (N,TIME) BINARY FLOAT (31),
                         PTR (TIM1) BINARY FIXED (31),
                         (TLEN,NG,NS,C,S,ZERO) BINARY FIXED (31),
                         (ADAR,KAR,AVN,SDAA,AAXP) (N) BINARY FLOAT (31),
                         (MU,VAR,MUO,VARO,KM,QM,PM) BINARY FLOAT (31);
14                   PTR(1)=0;
15                   PTR(2)=JP;
16                   ZERO=JP;
17                   DO J=3 TO TIM1;
18                     ZERO=(J-2)*PERIOD+ZERO+JP;
19                     PTR(J)=ZERO;
20                   END;
21                   TLEN=PTR(TIME)+SAMPLE;
22                   BEGIN;
23                     DCL (TGCAP,TSTAY,GCAP,ESTAY) (TLEN) BINARY FIXED (31),
                           (SCAP,NCAP)  (SAMPLE) BINARY FIXED (31),
                           AEXP (SAMPLE) BINARY FLOAT (31),
                           (MCAR,NDAR,SCAR,DAR,VNG) (N) BINARY FIXED (31),
                           (M,N1,M1,G,LAST,ESTZ,LLEN,ITO,IITO,J2) BINARY FIXED (31),
                           (SPAN,P,Q,CHIV,STU,T1,TZ,SAM,AIN,PP,QQ) BINARY FLOAT (31);
24                 CA: DO CAPROB=CAPST TO CAPEN BY CAPIN;
25                         ANS,FSCAP,FNCAP=0;
26                         TGCAP,TSTAY=0;
27                     SIAM: DO NSIM=1 TO N;
28                               GCAP,ESTAY=0;
29                               SCAP,NCAP=0;
30                        ORI: DO NAN=1 TO POPSIZE;
31                                 ANS(NSIM,1)=ANS(NSIM,1)+1;
32                                 CALL RANDOM (INT,REAL);
33                                 SPAN = MEANL*LOG(REAL);
34                                 IF SPAN >= 1 THEN
35                                 DO;
36                                   IF SPAN > SAMPLE THEN SPAN = SAMPLE;
38                                   CALL DONE (INI,SPAN,CAPROB,SCAP,NCAP,GCAP,ESTAY,
                                                INT,PERIOD,TIME,PTR,JP);
39                                 END;
40                              END ORI;
41                                 ITO=1;     TZ=1;
43                      AGAIN: CALL RANDOM (INT,REAL);
44                                 T1=ARRIV*LOG (REAL);
```

```
                    TZ=TZ+T1;
                    IF TZ > SAMPLE THEN GO TO FINI;
                    IF (TZ-ITO) > 1 THEN ITO=FLOOR(TZ);
                    IF ITO = SAMPLE THEN J=TIME;
                                   ELSE IF ITO < JP THEN J=1;
                                   ELSE J=(ITO-JP+PERIOD)/PERIOD+1;
                    ANS(NSIM,J)=ANS(NSIM,J)+1;
                    CALL RANDOM (INT,REAL);
                    SPAN =MEANL*LOG (REAL);
                    SAM=SPAN+TZ;
                    IF (SAM-ITO) < 1 THEN GO TO AGAIN;
                    IITO=ITO+1;
                    IF SAM > SAMPLE THEN SAM = SAMPLE;
                    CALL DONE (IITO,SAM,CAPROB,SCAP,NCAP,GCAP,ESTAY,
                           INT,PERIOD,TIME,PTR,JP);
                    GO TO AGAIN;
          FINI: TGCAP=TGCAP+GCAP;
                TSTAY=TSTAY+ESTAY;
                M,N1=0;
                DO J=1 TO TIME;
                 IF J =1 THEN C=1; ELSE C=(J-2)*PERIOD+1+JP;
                  S=PERIOD*(J-1)+JP;
                  DO J1=C TO S;
                    M=SCAP(J1)+M;
                    N1=NCAP(J1)+N1;
                  END;
                  FSCAP(NSIM,J)=M;
                  FNCAP(NSIM,J)=N1;
                  MUO=FLOAT(M)/N1;
                  C=PTR(J);
                  S=PTR(J+1);
                  PHO(NSIM,J)=GCAP(C+1);
                  VAR,QQ=0;
                  DO G=C+1 TO S;
                    MU=(G-C)-MUO;
                    VAR=VAR+MU*MU*GCAP(G);
                    QQ=QQ+(G-C-1)*ESTAY(G);
                  END;
                  TME(NSIM,J)=MUO;
                  TVE(NSIM,J)=VAR/N1;
                  MB(NSIM,J)=QQ/N1;
                END;
              END SIAM;
          M1=1;
          MDAR=0;
          DO J=1 TO TIME;
            CALL PRTR(J,CAPROB,PERIOD,N,JP);
            PUT SKIP(5) EDIT ('TOTAL','MEAN','VAR.')
                        (X(71),A(5),2(X(14),A(4)));
            NS,ZERO,S,C=0;
            DO J1 =1 TO N;
              MDAR(J1)=MDAR(J1)+ANS(J1,J);
              NS=NS+MDAR(J1);
              NDAR(J1)=FNCAP(J1,J);
              S=S+NDAR(J1);
              SDAR(J1)=FSCAP(J1,J);
              C=C+SDAR(J1);
```

```
108        DAR(J1)=MCAR(J1)-NCAR(J1);
109        ZERO=ZERO+DAR(J1);
110      END;
111      CALL MV(MCAR,N,MU,VAR);
112      PUT SKIP(2) EDIT ('POPUL. ( TRUE )',NS,MU,VAR)
                     (X(20),A(15),X(29),F(12),2(X(6),F(12,4)));
113      CALL MV(DAR,N,MU,VAR);
114      PUT SKIP(2) EDIT ('ZERO ( TRUE )',ZERO,MU,VAR)
                     (X(20),A(13),X(31),F(12),2(X(6),F(12,4)));
115      CALL MV(NDAR,N,MU,VAR);
116      PUT SKIP(2) EDIT ('INDIVIDUALS',S,MU,VAR)
                     (X(20),A(11),X(33),F(12),2(X(6),F(12,4)));
117      CALL MV(SDAR,N,MU,VAR);
118      PUT SKIP(2) EDIT ('CAPTURES',C,MU,VAR)
                     (X(20),A(8),X(36),F(12),2(X(6),F(12,4)));
119      PUT PAGE;
120      PUT SKIP(5) EDIT ('OBSERVED RECAPTURE FQ.')
                     (X(5),A(24));
121      M=PTR(J)+1;
122      N1=PTR(J+1);
123      LLEN=N1-M+1;
124      DO J1=M TO N1;
125        IF TGCAP(J1) > 0 THEN LAST=J1;
127      END;
128      PUT SKIP(3) EDIT ('(',ZERO,')')
                     (X(4),A(1),F(8),X(1),A(1));
129      PUT SKIP(2) EDIT ((TGCAP(J1) DO J1=M TO LAST))
                     (10(X(4),F(8)));
130      STU=FLOAT(S);
131      NG=STU*(C-1)/(C-STU);
132      P=STU/C;
133      Q=1-P;
134      ESTZ=NG-S;
135      MUO=C/STU;
136      VARO=0;
137      DO J1=1 TO LAST-M+1;
138        AIN=J1-MUO;
139        VARO=VARO+AIN*AIN*TGCAP(M+J1-1);
140      END;
141      VARO=VARO/STU;
142      PUT SKIP(4) EDIT ('MEAN','VAR.') (X(70),2(X(10),A(4)));
143      PUT SKIP(2) EDIT ('ZERO CLASS EXCLUDED',MUO,VARO)
                     (X(30),A(19),X(21),2(X(6),F(8,4)));
144      CALL MVO (TGCAP,M,N1,C,ESTZ,MUO,MU,VAR);
145      PUT SKIP(2) EDIT ('ZERO CLASS INCLUDED',MU,VAR)
                     (X(30),A(19),X(21),2(X(6),F(8,4)));
146      PUT SKIP(4) EDIT ('MAXIMUM LKD. FIT') (X(5),A(17));
147      PUT SKIP(3) EDIT ('(',ESTZ,')')
                     (X(4),A(1),F(8),X(1),A(1));
148      LAST=LLEN;
149      AIN=P*NG;
150      DO J2=1 TO LLEN;
151        AIN=AIN*Q;
152        IF AIN >= 1.0 THEN AEXP(J2)=AIN;
154                      ELSE
154                      DO;
155                        LAST=J2-1;
```

```
MUS: PROCEDURE OPTIONS (MAIN);

                                    GO TO MUM;
                                 END;
                      END;
                 MUM: PUT SKIP(2) EDIT ((AEXP(J1) DO J1=1 TO LAST))
                                       (10(X(4),F(8,1)));
                      CALL CHITEST (TGCAP,AEXP,LAST,CHIV,N1,M,G);
                      PUT SKIP(2) EDIT ('P  =  ',P) (X(5),A(6),X(8),F(8,2));
                      PUT SKIP(2) EDIT ('Q  =  ',Q) (X(5),A(6),X(8),F(8,2));
                      PUT SKIP(4) EDIT ('CHI VALUE IS  ',CHIV,G)
                              (X(5),A(14),F(8,2),X(10),F(3));
                      PUT SKIP(4) EDIT ('MEAN','VAR.') (X(60),2(X(14),A(4)));
                      G=0;
                      DO J1=1 TO N;
                        J2=SDAR(J1);
                        STU=FLOAT(NDAR(J1));
                        ITO=J2-STU;
                        IF ITO <= 0 THEN
                                DO;
                                  DAR(J1)=0;
                                  VNG(J1)=0;
                                  G=G+1;
                                END;
                             ELSE
                                DO;
                                    DAR(J1)=STU*(J2-1)/ITO;
                        VNG(J1)=DAR(J1)*(STU*J2-J2+STU)/(J2-STU)**2;
                                END;
                        SDAR(J1)=DAR(J1)-MDAR(J1);
                        AAXP(J1)=FLOAT(ABS(SDAR(J1)))/MDAR(J1);
                      END;
                      Q=1-CAPROB;
                      PP=KK/(1-Q*K);
                      QQ=(1-Q)*K/(1-Q*K);
                      CALL MV (DAR,N,MU,VAR);
                      PUT SKIP(2) EDIT ('NG',MU,VAR,'REJECT',G,'/',N)
                                       (X(38),A(2),X(20),2(X(6),F(12,4)),
                                       X(5),A(6),X(1),F(3),X(1),A(1),X(1),F(3));
                      CALL MV (SDAR,N,MU,VAR);
                      PUT SKIP(2) EDIT ('NG-N',MU,VAR)
                                       (X(38),A(4),X(18),2(X(6),F(12,4)));
                      CALL MV (VNG,N,MU,VAR);
                      PUT SKIP(2) EDIT ('VAR. NG',MU,VAR)
                                       (X(38),A(7),X(15),2(X(6),F(12,4)));
                      CALL MIV (AAXP,N,MU,VAR);
                      PUT SKIP(2) EDIT ('(NG-N)/N',MU,VAR)
                                       (X(38),A(8),X(14),2(X(6),F(12,4)));
                      CALL THEO (AEXP,S,PP,QQ,LLEN,LAST);
                      PUT SKIP(4) EDIT ('THEORETICAL FIT') (X(5),A(16));
                      PUT SKIP(2) EDIT ((AEXP(J1) DO J1=1 TO LAST))
                                       (10(X(4),F(8,1)));
                      CALL CHITEST(TGCAP,AEXP,LAST,CHIV,N1,M,G);
                      PUT SKIP(2) EDIT ('CHI VALUE IS  ',CHIV,G) (X(5),A(14),
                                                 F(8,2),X(10),F(3));
                      PUT PAGE;
                      DO J1=M TO N1;
                        IF TSTAY(J1) > 0 THEN LAST=J1;
                      END;
```

```
MUS: PROCEDURE OPTIONS (MAIN);

204   TZ=FLOAT(TGCAP(M))/S;
205   T1=1-TZ;
206   VAR,AIN=0;
207   DO J1=M+1 TO LAST;
208     AIN=AIN+(J1-M)*TSTAY(J1);
209   END;
210   AIN=AIN/S;
211   DO J1=M TO LAST;
212     MU=(J1-M)-AIN;
213     VAR=VAR+MU*MU*TSTAY(J1);
214   END;
215   VAR=VAR/S;
216   PUT SKIP(5) EDIT ('OBSERVED DURATION FREQUENCY ',
     ' (TOTAL) ','MEAN = ',AIN,'VAR. = ',VAR)
          (X(5),A(29),A(10),X(5),2(X(5),A(7),X(2),F(9,6)));
217   PUT SKIP(3) EDIT ((TSTAY(J1) DO J1=M TO LAST))
                    (10(X(4),F(8)));
218    KM=1-(T1/AIN);
219    QM=(KM-T1)/(KM*TZ);
220    PM=(1-KM)/(1-QM*KM);
221    CALL HOL (AEXP,QM,KM,PM,LLEN,LAST,S);
222   PUT SKIP(4) EDIT ('MAXIMUM LKD FIT')
                    (X(5),A(15));
223   PUT SKIP(2) EDIT ('Q =  ',QM) (X(12),A(5),F(8,6));
224   PUT SKIP(2) EDIT ('K =  ',KM) (X(12),A(5),F(8,6));
225   PUT SKIP(3) EDIT ((AEXP(J1) DO J1=1 TO LAST))
                    (10(X(4),F(8,1)));
226   TEPK=0;
227   DO J1=1 TO LAST;
228     TEPK=TEPK+AEXP(J1);
229   END;
230   IF S > TEPK THEN TEPK=S-TEPK;
232               ELSE TEPK=0;
233   PUT SKIP(2) EDIT ('REMAINDER',TEPK)
             (X(10),A(9),X(5),F(8,1));
234   CALL CHITEST(TSTAY,AEXP,LAST,CHIV,N1,M,G);
235   PUT SKIP(2) EDIT ('CHI VALUE IS  ',CHIV,G) (X(5),A(14),
                                   F(8,2),X(10),F(3));
236   G=0;
237   DO J1=1 TO N;
238     STU=NDAR(J1);
239     TZ=PHO(J1,J)/STU;
240     T1=1-TZ;
241     AIN=TME(J1,J)-(TVE(J1,J)*TZ)/(TME(J1,J)*T1);
242     IF AIN <=0.0 THEN DO;
244                          G=G+1;
245                          AVN(J1)=0;
246                       END;
247                   ELSE AVN(J1)=FSCAP(J1,J)/AIN;
248     TEPK=1-(T1/MB(J1,J));
249     ADAR(J1)=STU/T1*TEPK;
250     SDAA(J1)=ADAR(J1)-MDAR(J1);
251     AAXP(J1)=ABS(SDAA(J1))/MDAR(J1);
252     KAR(J1)=TEPK;
253   END;
254   PUT SKIP(4) EDIT ('MEAN','VAR.','S.D.')
                                   (X(60),3(X(14),A(4)));
```

```
255                       CALL MIV (ADAR,N,MU,VAR);
256                       PUT SKIP(2) EDIT ('NH',MU,VAR)
                                    (X(38),A(2),X(20),2(X(6),F(12,4)));
257                       CALL MIV (SDAA,N,MU,VAR);
258                       PUT SKIP(2) EDIT ('NH-N',MU,VAR)
                                    (X(38),A(4),X(18),2(X(6),F(12,4)));
259                       CALL MIV (AAXP,N,MU,VAR);
260                       PUT SKIP(2) EDIT ('(NH-N)/N',MU,VAR)
                                    (X(38),A(8),X(14),2(X(6),F(12,4)));
261                       DO J1=1 TO N;
262                        ADAR(J1)=AVN(J1)-MCAR(J1);
263                       AAXP(J1)=ABS(ADAR(J1))/MCAR(J1);
264                       END;
265                       CALL MIV (AVN,N,MU,VAR);
266                       PUT SKIP(2) EDIT ('NB',MU,VAR,'REJECT',G,'/',N)
                                    (X(38),A(2),X(20),2(X(6),F(12,4)),
                                    X(5),A(6),X(1),F(3),X(1),A(1),X(1),F(3));
267                       CALL MIV (ADAR,N,MU,VAR);
268                       PUT SKIP(2) EDIT ('NB-N',MU,VAR)
                                    (X(38),A(4),X(18),2(X(6),F(12,4)));
269                       CALL MIV (AAXP,N,MU,VAR);
270                       PUT SKIP(2) EDIT ('(NB-N)/N',MU,VAR)
                                    (X(38),A(8),X(14),2(X(6),F(12,4)));
271                       MU=KK*KK*PP;
272                       DO J1=1 TO N;
273                         ADAR(J1)=KAR(J1)-K;
274                         AAXP(J1)=ABS(ADAR(J1))/K;
275                         SDAA(J1)=SQRT(MU/NCAR(J1));
276                       END;
277                       CALL MIV(KAR,N,MU,VAR);
278                       MUO=SQRT(VAR);
279                       PUT SKIP(2) EDIT ('KH',MU,VAR,MUO)
                                    (X(38),A(2),X(20),3(X(6),F(12,4)));
280                       CALL MIV (ADAR,N,MU,VAR);
281                       MUO=SQRT(VAR);
282                       PUT SKIP(2) EDIT ('KH-K',MU,VAR,MUO)
                                    (X(38),A(4),X(18),3(X(6),F(12,4)));
283                       CALL MIV (AAXP,N,MU,VAR);
284                       MUO=SQRT(VAR);
285                       PUT SKIP(2) EDIT ('(KH-K/K)',MU,VAR,MUO)
                                    (X(38),A(8),X(14),3(X(6),F(12,4)));
286                        CALL MIV(SDAA,N,MU,VAR);
287                       PUT SKIP(2) EDIT ('S. D. OF K',MU,VAR)
                                   (X(38),A(10),X(12),3(X(6),F(12,9)));
288                        CALL HCL(AEXP,Q,K,PP,LLEN,LAST,S);
289                       PUT PAGE;
290                       PUT SKIP(10) EDIT ('THEORETICAL FIT')
                                    (X(5),A(15));
291                       PUT SKIP(2) EDIT ('K =   ',K) (X(12),A(5),F(8,6));
292                       PUT SKIP(3) EDIT ((AEXP(J1) DO J1=1 TO LAST))
                                    (10(X(4),F(8,1)));
293                       TEPK=0;
294                       DO J1=1 TO LAST;
295                         TEPK=TEPK+AEXP(J1);
296                       END;
297                       IF S > TEPK THEN TEPK=S-TEPK;
299                                   ELSE TEPK=0;
```

```
300                          PUT SKIP(2) EDIT ('REMAINDER',TEPK)
                                 (X(10),A(9),X(5),F(8,1));
301                          CALL CHITEST(TSTAY,AEXP,LAST,CHIV,N1,M,G);
302                          PUT SKIP(2) EDIT ('CHI VALUE IS  ',CHIV,G) (X(5),A(14),
                                                    F(8,2),X(10),F(3));
303                END;
304                END;
305                END;
306                END;
307                DONE: PROCEDURE (INI,SPAN,CAPROB,SCAP,NCAP,GCAP,ESTAY,INT,PER,PERI,
                                 PTR,JP);
308                    DCL (SCAP(*),NCAP(*),GCAP(*),ESTAY(*),MARK,INT,LCAP,FCAP,
                          PER,INI,PERI,M,JJ,JK,IBI,JM,PTR(*),J,JZ,JP)
                            BINARY FIXED (31),
                         (REAL,CAPRCB,SPAN) BINARY FLOAT (31);
309                    DCL RANDOM ENTRY (BINARY FIXED (31),BINARY FLOAT (31));
310                    MARK=0;
311                    DO J=INI TO SPAN;
312                      CALL RANDOM (INT,REAL);
313                      IF REAL < CAPROB THEN
314                      DO;
315                        IF MARK=0 THEN
316                          DO;
317                            NCAP(J)=NCAP(J)+1;
318                            FCAP=J;
319                            IF J <= JP THEN JJ=1;
321                                       ELSE JJ=(J-JP+PER-1)/PER+1;
322                          END;
323                          ELSE
323                          DO;
324                            IF J <= JP THEN JK=1;
326                                       ELSE JK=(J-JP+PER-1)/PER+1;
327                            IF JK > JJ THEN
328                              DO;
329                                M=LCAP-FCAP+1;
330                                DO JZ=JJ TO JK-1;
331                                  JM=PTR(JZ);
332                                  GCAP(JM+MARK)=GCAP(JM+MARK)+1;
333                                  ESTAY(JM+M)=ESTAY(JM+M)+1;
334                                END;
335                                JJ=JK;
336                              END;
337                          END;
338                        MARK=MARK+1;
339                        SCAP(J)=SCAP(J)+1;
340                        LCAP=J;
341                      END;
342                    END;
343                    IF MARK > 0 THEN
344                      DO;
345                        IF LCAP <= JP THEN JK=1;
347                                     ELSE JK=(LCAP-JP+PER-1)/PER+1;
348                        M=LCAP-FCAP+1;
349                        DO JZ=JK TO PERI;
350                          JM=PTR(JZ);
351                          GCAP(JM+MARK)=GCAP(JM+MARK)+1;
352                          ESTAY(JM+M)=ESTAY(JM+M)+1;
```

```
353                          END;
354                        END;
355                    END DONE;
356             TMV: PROCEDURE (DUM,TOL,MU,VAR,LEN,STA);
357                    DCL (DUM(*),TOL,LEN,STA,J) BINARY FIXED (31),
                           (MU,VAR,AIN) BINARY FLOAT (31);
358                    VAR,TOL=0;
359                    DO J=STA TO LEN;
360                      TOL=TOL+DUM(J);
361                    END;
362                    MU=FLOAT(TOL)/LEN;
363                    DO J=STA TO LEN;
364                      AIN=DUM(J)-MU;
365                      VAR=VAR+AIN*AIN;
366                    END;
367                    VAR=VAR/LEN;
368                  END TMV;
369             MVO: PROCEDURE (ARA,ST,LEN,S,Z,MU,MUO,VAR);
370                    DCL (ARA(*),ST,LEN,Z,S,J) BINARY FIXED (31),
                           (MU,MUO,VAR,VAL,INI,AIN) BINARY FLOAT (31);
371                    INI=S+Z;
372                    MUO=MU*S/INI;
373                    VAR=MUO*MUO*Z;
374                    AIN=0;
375                    DO J=ST TO LEN;
376                      AIN=AIN+1;
377                      VAL=AIN-MUO;
378                      VAR=VAR+VAL*VAL*ARA(J);
379                    END;
380                    VAR=VAR/INI;
381                  END MVO;
382             CHITEST: PROCEDURE (OBS,EXPECT,LAST,VALUE,SAMI,FST,M);
383                        DCL (OBS(*),LAST,SAMI,FST,J,JJ,M,PT,FT) BINARY FIXED (31),
                               (EXPECT(*),VALUE,IDIF,INT) BINARY FLOAT (31);
384                        FT=FST-1;
385                        M=LAST;
386                        DO J=1 TO LAST;
387                          IF EXPECT(J) < 5 THEN
388                            DO;
389                              M=J;  INT=0;
391                              DO JJ=M TO LAST;
392                                INT=INT+EXPECT(JJ);
393                              END;
394                            IF INT < 5 THEN
395                              DO;
396                                M=M-1;
397                                EXPECT(M)=EXPECT(M)+INT;
398                              END;
399                              ELSE EXPECT(M)=INT;
400                              GO TO BG;
401                            END;
402                        END;
403                  BG: PT=M+FT;
404                        INT=0;
405                        DO JJ=PT TO SAMI;
406                        INT=INT+OBS(JJ);
407                        END;
```

```
                    AG: IF INT < 5 THEN
                                   DO;
                                     PT=PT-1;
                                     INT=INT+OBS(PT);
                                     M=M-1;
                                     EXPECT(M)=EXPECT(M)+EXPECT(M+1);
                                     GO TO AG;
                                   END;
                        OBS(PT)=INT;
                        VALUE=0;
                        DO J=1 TO M;
                          IDIF=OBS(J+PT)-EXPECT(J);
                          IF IDIF¬= 0 THEN VALUE=VALUE+(IDIF*IDIF)/EXPECT(J);
                        END;
                        END CHITEST;
                THEO: PROCEDURE (ARY,NUM,PP,QQ,LEN,N);
                        DCL (ARY(*),PP,QQ,VAL) BINARY FLOAT (31),
                            (NUM,LEN,N,J) BINARY FIXED (31);
                        ARY(1),VAL=NUM*PP;
                        N=LEN;
                        DO J=2 TO LEN;
                          VAL=VAL*QQ;
                          IF VAL < 1.0 THEN
                            DO;
                              N=J-1;
                              GO TO FINI;
                            END;
                          ARY(J)=VAL;
                        END;
                FINI: END THEO;
                HOL: PROCEDURE (ARY,Q,K,KK,LEN,LAST,N);
                       DCL (ARY(*),Q,K,KK,AIN) BINARY FLOAT (31),
                           (J,LEN,LAST,N) BINARY FIXED (31);
                       LAST=LEN;
                       ARY(1)=KK*N;
                       ARY(2)=ARY(1)*(1-Q)*K;
                       DO J=3 TO LEN;
                         AIN=ARY(J-1)*K;
                         IF AIN < 1.0 THEN
                                       DO;
                                         LAST=J-1;
                                         GO TO FINE;
                                       END;
                         ARY(J)=AIN;
                       END;
                 FINE: END HOL;
                MV: PROCEDURE (ARY,LEN,MEAN,VAR);
                      DCL (ARY(*),LEN,J) BINARY FIXED (31),
                          AY2 (LEN) BINARY FIXED (31),
                          (MEAN,VAR,TEM,TM) BINARY FLOAT (31);
                      MEAN,VAR=0;
                      TEM=ARY(1);
                      DO J=1 TO LEN;
                        TM=ARY(J)-TEM;
                        AY2(J)=TM;
                        MEAN=MEAN+TM;
                      END;
```

```
                MEAN=MEAN/LEN;
                DO J=1 TO LEN;
                  TM=AY2(J)-MEAN;
                  VAR=VAR+TM*TM;
                END;
                MEAN=MEAN+TEM;
                VAR=VAR/LEN;
              END MV;
           MIV: PROCEDURE (ARY,LEN,MU,VAR);
                  DCL (ARY(*),AIN,VAR,MU,TM,TEM) BINARY FLOAT(31),
                      YY(LEN) BINARY FLOAT (31),
                      (J,LEN) BINARY FIXED (31);
                  TEM=ARY(1);
                  AIN,VAR=0;
                  DO J=1 TO LEN;
                    TM=ARY(J)-TEM;
                    YY(J)=TM;
                    AIN=AIN+TM;
                  END;
                  MU=AIN/LEN;
                  DO J=1 TO LEN;
                    AIN=YY(J)-MU;
                    VAR=VAR+AIN*AIN;
                  END;
                  MU=MU+TEM;
                  VAR=VAR/LEN;
                END MIV;
           PRTR: PROCEDURE (I,CA,SAM,N,JP);
                   DCL (I,NUM,SAM,N,JP) BINARY FIXED (31),
                       CA BINARY FLOAT (31);
                   NUM=(I-1)*SAM+JP;
                   PUT PAGE;
                   PUT LINE (10) EDIT ('MODEL') (X(64),A(5));
                   PUT SKIP(0) EDIT ('_____') (X(64),A(5));
                   PUT SKIP(3) EDIT ('IMMIGRATION  AND  DEATH') (X(55),A(23));
                   PUT SKIP(3) EDIT ('PARAMETERS') (X(62),A(10));
                   PUT SKIP(0) EDIT ('__________') (X(62),A(10));
                   PUT SKIP(3) EDIT ('NO. OF SAMPLING PERIODS  : ',NUM)
                                    (X(51),A(27),F(4));
                   PUT SKIP(2) EDIT ('NO. OF SIMULATIONS       : ',N)
                                    (X(51),A(27),F(4));
                   PUT SKIP(2) EDIT ('CAPTURE PROBABILITY      : ',CA)
                                    (X(51),A(27),F(4,2));
                 END PRTR;
           END;
```

SAMPLE OUTPUT FROM THE

IMMIGRATION AND DEATH MODEL

MODEL

IMMIGRATION AND DEATH

PARAMETERS

NO. OF SAMPLING PERIODS : 80

NO. OF SIMULATIONS : 50

CAPTURE PROBABILITY : 0.05

	TOTAL	MEAN	VAR.
POPUL. (TRUE)	44905	898.1000	916.3300
ZERO (TRUE)	31013	620.2600	634.9924
INDIVIDUALS	13892	277.8400	260.6144
CAPTURES	19630	392.6000	602.9200

OBSERVED RECAPTURE FQ.

(31013)

9808	2899	842	243	77	20	3

	MEAN	VAR.
ZERO CLASS EXCLUDED	1.4130	0.5704
ZERO CLASS INCLUDED	0.5208	0.5277

MAXIMUM LKD. FIT

(33630)

9830.6	2873.6	840.0	245.5	71.8	21.0	6.1	1.8

P = 0.71

Q = 0.29

CHI VALUE IS 1.90 6

	MEAN	VAR.	
NG	955.0600	9194.2564	REJECT 0 / 50
NG-N	56.9600	7314.1584	
VAR. NG	8216.9200	4819107.1136	
(NG-N)/N	0.0923	0.0042	

THEORETICAL FIT

9415.6	3034.0	977.6	315.0	101.5	32.7	10.5	3.4	1.1

CHI VALUE IS 73.44 6

OBSERVED DURATION FREQUENCY (TOTAL) MEAN = 2.684855 VAR. = 38.602526

9808	439	398	342	306	277	270	243	211	186
161	132	114	131	85	63	68	78	66	54
39	44	48	38	35	22	21	20	26	33
15	12	11	9	9	6	15	4	8	6
1	4	5	2	8	2	2	1	1	0
1	3	1	1	1	3	0	0	0	2
0	0	0	0	0	0	0	1		

MAXIMUM LKD FIT

Q = 0.948800

K = 0.890504

9808.0	447.2	398.2	354.6	315.8	281.2	250.4	223.0	198.6	176.8
157.5	140.2	124.9	111.2	99.0	88.2	78.5	69.9	62.3	55.5
49.4	44.0	39.2	34.9	31.1	27.7	24.6	21.9	19.5	17.4
15.5	13.8	12.3	10.9	9.7	8.7	7.7	6.9	6.1	5.5
4.9	4.3	3.9	3.4	3.1	2.7	2.4	2.2	1.9	1.7
1.5	1.4	1.2	1.1						

REMAINDER 8.7

CHI VALUE IS 55.96 41

	MEAN	VAR.	S.D.
NH	848.3024	10571.6125	
NH-N	-49.7976	8723.7810	
(NH-N)/N	0.0933	0.0050	
NB	991.9154	145984.9566	REJECT 0 / 50
NB-N	93.8154	140135.0987	
(NB-N)/N	0.2988	0.0883	
KH	0.8898	0.0001	0.0102
KH-K	-0.0151	0.0001	0.0102
(KH-K/K)	0.0168	0.0001	0.0110
S. D. OF K	0.004706139	0.000000019	

THEORETICAL FIT

K = 0.904837

9415.6	426.0	385.4	348.8	315.6	285.5	258.4	233.8	211.5	191.4
173.2	156.7	141.8	128.3	116.1	105.0	95.0	86.0	77.8	70.4
63.7	57.7	52.2	47.2	42.7	38.6	35.0	31.6	28.6	25.9
23.4	21.2	19.2	17.4	15.7	14.2	12.9	11.6	10.5	9.5
8.6	7.8	7.1	6.4	5.8	5.2	4.7	4.3	3.9	3.5
3.2	2.9	2.6	2.3	2.1	1.9	1.7	1.6	1.4	1.3
1.2	1.1								

REMAINDER 10.0

CHI VALUE IS 261.67 47

BIBLIOGRAPHY

Arnason, A. N. and P. W. Chan (1971). The properties of the geometric and Holgate estimators applied to capture-recapture data. (In preparation.)

Burnham, K. P. and W. S. Overton (1969). A simulation study of livetrapping and estimation of population size. Oregon State University Technical Report No. 14. Oregon State University, Corvallis.

Brass, W. (1958). Simple methods of fitting the truncated negative binomial distribution. Biometrika, 45: 59-68.

Calhoun, J. B. (1963). The social use of space. Physiological Mammology edition. V. W. Mayer and R. G. Gilder. Academic Press.

Cormack, R. M. (1966). A test for equal catchability. Biometrics, 22: 330-342.

Cox, D. R. and H. D. Miller (1965). The theory of stochastic processes. 1st edition, Methuen, London.

Eberhardt, L. L. (1969). Population estimates from recapture frequencies. J. Wildl. Mgmt. 33: 28-39.

Holgate, P. (1964). A modified geometric distribution arising in trapping studies. Acta Theriologica, 9: 353-389.

Jolly, G. M. (1965). Explicit estimates from capture-recapture data with both death and immigration - stochastic model. Biometrika, 52: 225-246.

Tanton, M. T. (1965). Problems of live-trapping and populations estimation of the wood mouse, Apopedmus sylvaticus (L.). J. Anim. Ecol. 34: 1-22.

Tanton, M. T. (1969). The estimation and biology of population of the bank vole (Clethrionomys glareolus (SCHR.)) and wood mouse (Apodemus sylvaticus (L.)). J. Anim. Ecol., 38: 511-528.