HATCHMARK2

A statistical tool to conduct an *a priori* analysis of the precision and accuracy of estimators that discriminate between hatchery- and natural-origin spawning escapement using observed marks

Richard A. Hinrichsen and Rishi Sharma

June 28, 2012

Introduction

Assessments of the status of endangered Columbia River salmon populations, which include estimation of extinction probabilities and long-term trend of natural-origin fish, require reliable estimates of the proportion of hatchery-origin spawners on the spawning grounds (McClure et al. 2003). Furthermore, an assessment of the degree of interbreeding of hatchery-origin with the wild-origin segment, which may reduce the genetic fitness of subsequent generations, also depends on this estimate (Waples 1991). To allow distinction between natural-origin and hatchery-origin salmon in the Columbia Basin, the U.S. Congress presently requires the US Fish and Wildlife Service to visibly mark all hatchery production intended for harvest.[1] Visible marking of hatchery releases is a widespread practice among hatchery operators in the Columbia River basin, though non-visible marking procedures are sometimes substituted for or added to visible marks.

Despite the importance of estimates of proportion of hatchery-origin fish on the spawning grounds, reliable estimation techniques are lacking. The statistical difficulty of estimating the proportion of hatchery-origin escapement when hatchery-fish that are not visibly marked are present has been recognized for over thirty years (Hankin 1982). In Hankin's (1982) paper, it was assumed that all hatchery fish that were not coded-wire tagged were visibly marked at the same rate regardless of their hatchery of origin. This assumption certainly simplifies the analysis, but is not always realized in the Columbia Basin where source hatcheries are known to use different visible marking rates (Hinrichsen et al. 2012).

[1] On June 27, 2007, the House passed (amended) H.R. 2643, including a provision requiring the U.S. Fish and Wildlife Service to implement a system of mass marking of salmonid stocks that are released from federal hatcheries

An estimator of the proportion of hatchery-origin fish is obtained for a program in which certain fractions of juveniles are visibly marked (VM), coded-wire tagged (CWT), or both VM and CWT. Some of the VM fish are then recovered as adults at a given spawning area along with fish that are the progeny of salmon spawning in the wild. The recoveries are fish that were sampled at the spawning grounds, usually in a carcass survey. Carcasses that have a VM are then checked for a coded wire tag and if one exists, the hatchery of origin is identified from the tag. In the case of a single hatchery input where VM fraction at that hatchery is known, it is a simple to derive an estimate of hatchery-origin spawners in the survey: the estimate is equal to the number of spawners that are not VM in the carcass survey divided the VM fraction. The estimate of the proportion of hatchery-origin spawners is then equal to this estimate of the number of hatchery-origin spawners in the survey divided by the number of carcasses surveyed. In the case of multiple hatchery inputs with different VM fractions, the estimator of proportion of hatchery-origin spawners is more complicated.

The goal of this work is to present a reliable estimator in the general case where there are spawners from multiple hatcheries and different source hatcheries may use different VM fractions. This general case becomes important whenever different VM fractions are applied for the hatcheries that supply inputs to spawning escapements in the wild. If a single VM fraction is applied to all of the hatcheries supplying inputs to spawning escapement, then this generalization is not needed. However, when different VM fractions are applied, it becomes necessary to estimate the number of spawners in the sample that come from each hatchery. This problem is solved using the method of moments which results in a generalized least squares (GLS) estimation problem (Kariya and Kurata 2004). This GLS estimator (GLSE) is then used as the basis for an *a priori* statistical analysis that gives the precision and relative bias of the estimate of the proportion of hatchery-origin spawners as a function of certain variables that may be controlled at the hatchery or in the spawning ground surveys: VM fraction, CWT fraction, and sampling rate. As these variables are increased, the precision and accuracy of the estimate of the proportion of hatchery-origin spawners also increase. For those interested in monitoring hatchery-origin fish at a given level of precision (e.g., 15% CV), the *a priori* statistical tool HATCHMARK2 may be used to determine how to choose VM fraction, CWT fraction, and sampling rate to deliver that precision.

Statistical code for the analysis, written in the R programming language, may be found in Appendix A. For convenience, in the mathematical descriptions and derivations, we use an abbreviated set of variable names. The variable names used in the R code and the webtool HATCHMARK2 are given in Appendix B along with their definitions.

Table 1. —Assumptions[1]

(A1)	(Fixed probabilities) Hatchery-specific VM fractions and escapement sample rate are known.
(A2)	(Fixed probabilities) Hatchery-specific CWT fractions are known.
(A3)	(Identically distributed) Every individual spawner has the same probability of being sampled.
(A4)	(Identically distributed) Every individual hatchery-origin spawner from the same hatchery has the same probability of having a mark.
(A5)	(Independence) Whether any individual is sampled has no effect on the probability that another individual is sampled.
(A6)	(Independence) Whether any individual hatchery-origin spawners is observed to have a mark has no effect on the probability that another individual will have a mark.

1 For convenience, we derived the estimators for releases grouped at the hatchery level. To split the data by release group instead, simply replace "hatchery" by "release" in the estimation method and interpret VM fractions and CWT fractions as release-specific. The GLSE we derived generalizes the estimator of Hankin (1982), who assumed two groups of releases: one that was VM and CWT at 100%, and another that used a constant VM fraction.

Methods

We generalize the estimation problem of Hankin (1982) to handle multiple hatcheries with potentially different VM fractions and different CWT fractions. This problem is more complicated than the problem of a single hatchery because the number of VM fish from a given hatchery must be estimated: it may no longer be treated as known because only a fraction of the VM fish are given a unique tag identifying the hatchery of origin. Let λ_i represent the VM fraction that is applied to hatchery fish releases from hatchery *i* and that only a fraction ϕ_i are given a CWT that uniquely identifies the hatchery of origin. Further assume that each returning fish has the same probability of being sampled, θ. The outline of the mathematical derivations and the equation is presented here. The assumptions for this estimation problem are given in Table 1.

The assumptions in Table 1 allow us to express the joint distribution of escapement counts as a product of multinomial distributions:

$$f(\mathbf{x}) = \prod_{i=1}^{n} \binom{H_i}{x_{1,i}, x_{2,i}, x_{3,i}, x_{4,i}} (\theta\phi_i\lambda_i)^{x_{1,i}} (\theta\lambda_i(1-\phi_i))^{x_{2,i}} (\theta(1-\lambda_i))^{x_{3,i}} (1-\theta)^{x_{4,i}} \quad (1)$$

$$\times \binom{W}{x_5} \theta^{x_5} (1-\theta)^{W-x_5},$$

where H_i represents the hatchery-origin spawner escapement that originated in hatchery *i*, W represents the natural-origin spawner escapement, $x_{1,i}$ is the sampled and VM and CWT spawners from hatchery *i*, $x_{2,i}$ is the sampled and VM and not CWT spawners from hatchery *i*, $x_{3,i}$ is the number of sampled spawners from hatchery *i* that that are not visibly marked, $x_{4,i}$ is the unsampled spawners from hatchery *i*, x_5 is the sampled natural-origin spawners. Notice that $H_i = x_{i,1} + x_{i,2} + x_{i,3} + x_{i,4}$, which may be rearranged to give $x_{i,4} = H_i - x_{i,1} - x_{i,2} - x_{i,3}$.

Generalized least squares (GLS). —When there are multiple hatchery inputs, where not all VM fish are CWT, we apply the method of moments, which leads to an over-determined system of equations. This over-determined system is solved using GLS. To develop the system of equations, we use the method of moments, equating observed cell counts in the multinomial distributions to their expected values. Using this approach yields the following system of equations:

$$x_{1,i} = \theta\phi_i\lambda_i H_i \quad (2)$$

and

$$x_2 = \sum_{i=1}^{n} \theta\lambda_i(1-\phi_i)H_i \quad (3)$$

where $x_2 = \sum_{i=1}^{n} x_{2,i}$ is the total observed spawners that are VM but do not have a CWT to identify their hatchery of origin. Considering equations (2) and (3) over all input hatcheries forms a system of *n*+1 equations with *n* unknowns (the hatchery-specific escapements): an over-determined system. One approach to solving this over-determined system using GLS, which has a well-developed theory (Kariya and Kurata 2004). Because the observations are not all independent, we do not simply minimize the sum of squared differences between the observed observations and their expected values as in ordinary least squares, we instead minimize the square of the Mahalanobis distance given by

$$\left(\mathbf{x}-\mathbf{BH}\right)' \boldsymbol{\Sigma}^{-1}\left(\mathbf{x}-\mathbf{BH}\right) \tag{4}$$

where $\mathbf{x} = \begin{bmatrix} x_{11} & \dots & x_{1n} & x_2 \end{bmatrix}'$ is the vector of observations, $\mathbf{B}$ is a $(n+1)\times n$ matrix of cell count probabilities given by the partitioned matrix

$$\mathbf{B} = \begin{bmatrix} \mathbf{B}_1 \\ \mathbf{B}_2 \end{bmatrix} \tag{5}$$

where $\mathbf{B}_1 = \text{diag}(\theta\lambda_1\phi_1, \dots \theta\lambda_n\phi_n)$ is a diagonal $n\times n$ matrix, and $\mathbf{B}_2 = \left[\theta\lambda_1(1-\phi_1)\dots\theta\lambda_n(1-\phi_n)\right]$ is a row vector of dimension $1\times n$. $\boldsymbol{\Sigma}$ is the covariance matrix for the vector of observations, $\mathbf{x}$, given by the partitioned matrix

$$\boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{bmatrix} \tag{6}$$

where $\boldsymbol{\Sigma}_{11} = \text{diag}(H_1\theta\lambda_1\phi_1(1-\theta\lambda_1\phi_1), \dots, H_n\theta\lambda_n\phi_n(1-\theta\lambda_n\phi_n))$ is a diagonal $n\times n$ matrix, $\boldsymbol{\Sigma}_{12} = \left[H_1\theta^2\lambda_1^2\phi_1(1-\phi_1)\dots H_n\theta^2\lambda_n^2\phi_n(1-\phi_n)\right]'$ is an $n\times 1$ vector, $\boldsymbol{\Sigma}_{21} = \boldsymbol{\Sigma}_{12}'$, and

$\mathbf{\Sigma}_{22} = \sum_{i=1}^{n} H_i \theta\lambda_i (1-\phi_i)(1-\theta\lambda_i(1-\phi_i))$ is a scalar. With these definitions, it is then a matter of using the well-known solution to the minimization problem (treating the variance as fixed)

$$\hat{\mathbf{H}} = (\mathbf{B}'\Sigma^{-1}\mathbf{B})^{-1}\mathbf{B}'\mathbf{\Sigma}^{-1}\mathbf{x}\,. \tag{7}$$

The variance matrix for the GLSEs is given by

$$\text{var}(\hat{\mathbf{H}}) = (\mathbf{B}'\Sigma^{-1}\mathbf{B})^{-1}\,. \tag{8}$$

Using the above definitions, it may be shown that the GLSE for the number of hatchery-origin spawners from hatchery *i* is given by

$$\hat{H}_i = \frac{x_{1,i}}{\theta\lambda_i\phi_i} + \frac{\dfrac{H_i(1-\phi_i)}{\phi_i}}{\displaystyle\sum_{i=1}^{n}\left(\frac{H_i(1-\phi_i)\theta\lambda_i}{\phi_i}\right)}\left(x_2 - \sum_{i=1}^{n}\frac{x_{1,i}(1-\phi_i)}{\phi_i}\right), \tag{9}$$

where it is assumed that not all of the CWT fractions are 1, none of the CWT fractions is zero, none of VM fractions is zero, and sampling rate is not zero; otherwise, equation (9) is undefined. The special case of 100% CWT of all VM fish is handled below in equations (21)-(25). Continuing with the method of moments, the estimate of the total spawning escapement is

$$\hat{E} = \left(E_U + E_M\right)/\theta\,, \tag{10}$$

where E_U represents the observed spawners that are not VM, and E_M represents the observed VM spawners.

When all hatcheries use the same VM fraction, equation (9) reduces to

$$\hat{H} = \sum_{i=1}^{n} \hat{H}_i = \frac{E_M}{\theta\lambda}\,, \qquad (\text{assuming} \lambda_i = \lambda \text{ for } i = 1,\ldots,n) \tag{11}$$

where E_M is the observed number of VM spawners (with or without CWTs).

The proportion of hatchery-origin spawners may be estimated using the GLSEs of hatchery-origin escapement and total escapement as follows:

$$\hat{p} = \frac{\hat{H}}{\hat{E}}\,, \tag{12}$$

where $\hat{E} = \hat{H} + \hat{W}$ is the estimate of the total spawning population.

Using equation (8) along with the fact that $H = \sum_{i=1}^{n} H_i$, we may write

$$\text{var}(\hat{H}) = \sum_{i=1}^{n} \frac{H_i(1-\theta\lambda_i\phi_i)}{\theta\lambda_i\phi_i} - \frac{\left(\sum_{i=1}^{n} \frac{H_i(1-\phi_i)}{\phi_i}\right)^2}{\sum_{i=1}^{n} \frac{H_i(1-\phi_i)\theta\lambda_i}{\phi_i}}. \tag{13}$$

It was assumed in the above formula for $\text{var}\left(\hat{H}\right)$ that not all of the CWT fractions were equal to 1. In that event, the formula is undefined. The case of 100% CWT of all VM fish will be handled below in equations (21)-(25).

In the special case of a constant marking rate, the variance of the hatchery-origin escapement becomes

$$\text{var}(\hat{H}) = \sum_{i=1}^{n} \frac{H_i(1-\theta\lambda\ )}{\theta\lambda}\ , \qquad (\text{assuming}\lambda_i = \lambda \text{ for } i = 1,\ldots,n) \tag{14}$$

which involves no CWT fractions and is equivalent to the single-hatchery case.

We now return to the general case. The main focus on this work is not the variance of $\hat{H}$, but the variance of $\hat{p}$. For this we use the technique of a Taylor Series expansion of p about the GLSEs $\hat{H}$ and $\hat{E}$. It will then become apparent that the variance of $\hat{p}$ may be written as a function of the true value of p, E, the variances of $\hat{H}$ and $\hat{E}$ and their covariance. Using the multinomial distributions defined in equation (1), we may write

$$\text{var}(\hat{E}) = \text{var}\left(\sum_{i=1}^{n}\left(\frac{H_i - x_{4,i}}{\theta}\right) + \frac{x_5}{\theta}\right) = \sum_{i=1}^{n}\left(\frac{\text{var}(x_{4,i})}{\theta^2}\right) + \frac{\text{var}(x_5)}{\theta^2} \tag{15}$$

$$= \frac{H(1-\theta)\theta + W(1-\theta)\theta}{\theta^2}$$

$$= \frac{E(1-\theta)}{\theta}$$

$$\operatorname{cov}(\hat{H},\hat{E}) = \operatorname{cov}\left(\sum_{i=1}^{n} \frac{x_{1,i}}{\theta\lambda_i\phi_i} + \frac{m}{\theta}\left(x_2 - \sum_{i=1}^{n} \frac{x_{1,i}(1-\phi_i)}{\phi_i} \right), \sum_{i=1}^{n}\left(\frac{H_i - x_{4,i}}{\theta} \right) + \frac{x_5}{\theta} \right) \tag{16}$$

$$= \frac{-1}{\theta^2}\sum_{i=1}^{n}\left(\frac{1}{\lambda_i\phi_i} - m\frac{(1-\phi_i)}{\phi_i} \right)\operatorname{cov}(x_{1,i}, x_{4,i}) + m\operatorname{cov}(x_2, x_{4,i})$$

$$= \frac{(1-\theta)}{\theta} H \,,$$

where

$$m = \frac{\sum_{i=1}^{n} \frac{H_i(1-\phi_i)}{\phi_i}}{\sum_{i=1}^{n} \frac{H_i(1-\phi_i)\theta\lambda_i}{\phi_i}} \,. \tag{17}$$

With these variance and covariance formulas it is now possible to derive the variance of the estimate of the proportion of hatchery-origin spawners. Using a first-order Taylor series expansion, we may write

$$\hat{p} \; - p \cong (\nabla p)' \begin{bmatrix} \hat{H} - H \\ \hat{E} - E \end{bmatrix}, \tag{18}$$

$$\text{var}\left(\hat{p}\ \right) \cong \left(\nabla p\right)' \text{var}\begin{bmatrix}\hat{H}\\ \hat{E}\end{bmatrix}\nabla p \tag{19}$$

$$= \frac{1}{E}\left\{\sum_{i=1}^{n}\frac{p_i(1-\theta\lambda_i\phi_i)}{\theta\lambda_i\phi_i} - \frac{\left(\sum_{i=1}^{n}\frac{p_i(1-\phi_i)}{\phi_i}\right)^2}{\sum_{i=1}^{n}\frac{p_i(1-\phi_i)\theta\lambda_i}{\phi_i}} - p^2\frac{(1-\theta)}{\theta}\right\},$$

where $p_i = H_i / E$. It was assumed in the above formula for $\text{var}\left(\hat{p}\right)$ that not all of the CWT fractions were equal to 1. In that event, the variance formula in equation (19) is undefined. The special case of CWT of all VM fish will be treated below in equations (21)-(25). The theoretical variance formula is useful because it shows clearly how the variance is related to the sampling rate, VM fractions, and CWT fractions.

Notice that in the special case where the VM fraction is constant, equation (19) reduces to

$$\text{var}\left(\hat{p}\right) = \frac{p}{E}\left\{\frac{(1-\lambda\theta)}{\lambda\theta} - p\ \ \frac{(1-\theta)}{\theta}\right\}. \qquad (\text{assuming}\ \lambda_i = \lambda \text{ for } i = 1,\ldots,n) \tag{20}$$

Special case (all VM fish given a CWT). —In the special case where all VM hatchery fish are given a coded wire tag, the hatchery-specific escapements are given by

$$\hat{H}_i = \frac{x_{1,i}}{\theta\lambda_i}. \qquad (\text{assuming}\ \ \phi_i = 1 \text{ for } i = 1,\ldots,n) \tag{21}$$

Using equation (1) it is easily shown that

$$\text{var}(\hat{H}_i) = \frac{\text{var}(x_{1,i})}{(\theta\lambda_i)^2} = \frac{H_i(1-\theta\lambda_i)}{\theta\lambda_i}, \qquad \text{(assuming } \phi_i = 1 \text{ for } i = 1,\ldots,n) \tag{22}$$

$$\text{var}(\hat{E}) = \text{var}\left(\sum_{i=1}^{n} \frac{H_i - x_{4,i}}{\theta} + \frac{x_5}{\theta}\right) = E\frac{(1-\theta)}{\theta}, \qquad \text{(assuming } \phi_i = 1 \text{ for } i = 1,\ldots,n) \tag{23}$$

$$\text{cov}(\hat{H}, \hat{E}) = \text{cov}\left(\sum_{i=1}^{n} \frac{x_{1,i}}{\theta\lambda_i}, \sum_{i=1}^{n} \frac{H_i - x_{4,i}}{\theta} + \frac{x_5}{\theta}\right) \qquad \text{(assuming } \phi_i = 1 \text{ for } i = 1,\ldots,n) \tag{24}$$

$$= H\left(\frac{1-\theta}{\theta}\right),$$

and

$$\text{var}(\hat{p}) \cong (\nabla p)' \, \text{var}\begin{bmatrix}\hat{H} \\ \hat{E}\end{bmatrix} \nabla p \qquad \text{(assuming } \phi_i = 1 \text{ for } i = 1,\ldots,n) \tag{25}$$

$$= \frac{1}{E}\left\{\sum_{i=1}^{n} \frac{p_i(1-\theta\lambda_i)}{\theta\lambda_i} - p^2\frac{(1-\theta)}{\theta}\right\}.$$

.

In all cases, the variance of the wild-origin escapement estimate is calculated using the equation

$$\text{var}\left(\hat{W}\right) = \text{var}(\hat{E} - \hat{H}) = \text{var}(\hat{E}) + \text{var}(\hat{H}) - 2\,\text{cov}(\hat{E}, \hat{H}). \tag{26}$$

Theoretical estimates of CV and SE in HATCHMARK2 were calculated using the theoretical variance formulas derived above. Alternatively, CV and SE are estimated using Monte Carlo simulation.

Monte Carlo simulations. — To evaluate the relative bias and precision of these GLSEs, Monte Carlo simulation was used. Using Monte Carlo simulation as an alternative to the theoretical estimates of precision can be important when sample size is low and the asymptotic properties may not apply. The idea is to assume some true values of the hatchery-origin and natural-origin escapements, simulate the data collection process again and again and use these simulated data and true values to gauge the precision and accuracy of the escapement estimates. The underlying assumptions are that a spawning fish is sampled with probability θ, the probability that an observed hatchery fish from hatchery *i* is VM is λ_i, and the probability that a VM fish from hatchery *i* is also CWT is ϕ_i. Using binomial random variables, we generate *M* Monte Carlo replications of the estimate the proportion of spawners that are of hatchery origin. The Monte Carlo replications are then used to determine that statistical properties of the GLSE, including: standard error, coefficients of variation, and relative bias. Bias is calculated as a relative bias, which is the true bias divided by the true value of the parameter estimated. Besides the special cases where (a) all the VM fish are given a coded wire tag or (b) all hatcheries mark the same fraction of releases, there two more special cases to consider in the estimation of $\hat{p}$. Whenever $x_{1,i} = 0;\ i = 1,\ldots,n$ and $x_2 = 0$, we use the estimate $\hat{p} = 0$. Assuming special cases (a) and (b) do not hold and $x_{1,i} = 0;\ i = 1,\ldots,n$ and $x_2 \neq 0$, $\hat{p}$ is assumed to be unestimable because there is no information in the data set that may be used to divide the x_2 returns by hatchery of origin.

Given the 10,000 Monte Carlo replications of the estimate of the hatchery proportion of spawning escapement, denoted by $\hat{p}_1^*, \hat{p}_2^*, \ldots \hat{p}_M^*$, where $M = 10{,}000$ represents the number of Monte Carlo replications[2], the Monte Carlo estimate of variance is given by

$$\mathrm{var}^*\left(\hat{p}\right) = \sum_{j=1}^{M} \frac{\left(\hat{p}_j^* - \bar{\hat{p}}^*\right)^2}{M-1}, \tag{27}$$

where $\bar{\hat{p}}^*$ represents the sample mean of the *M* Monte Carlo estimates. Bias is calculated as relative bias, namely,

[2] The number of Monte Carlo replications may be set by the user of HATCHMARK2.

$$bias^{*}\left(\hat{p}\right)=\frac{\left(\bar{\hat{p}}^{*}-\hat{p}\right)}{\hat{p}}, \tag{28}$$

where

$$SE^{*}\left(\hat{p}\right)=\sqrt{\mathrm{var}^{*}\left(\hat{p}\right)}, \tag{29}$$

The coefficient of variation is then estimated as

$$CV^{*}\left(\hat{p}\right)=\frac{SE^{*}\left(\hat{p}\right)}{p}. \tag{30}$$

Acknowledgements

The authors gratefully acknowledge the reviews of Charlie Paulsen, Tracy Hillman, Bruce Crawford, Michael Newsome, Craig Busack, Brian Maschhoff, and Tim Fisher. This work was supported by Bonneville Power Administration. The views expressed are solely those of the authors and are not intended to represent the views of any organization with which the authors are affiliated.

References

Hankin, D.G. 1982. Estimating escapement of pacific salmon: marking practices to discriminate wild and hatchery fish. Transpactions of the American Fisheries Society 111:286-298.

Hinrichsen, R.A., R. Sharma, T.R. Fisher. 2012. Precision and accuracy of estimators of the proportion of hatchery-origin spawners. Transactions of the American Fisheries Society 142:437-454.

Kariya, T. and H. Kurata. 2004. Generalized Least Squares. Wiley Series in Probability and Statistics. Wiley. New York, New York.

McClure, M.M., E.E. Holmes, B.L. Sanderson, and C.E. Jordan. 2003. A large-scale multispecial status assessment: anadromous salmonids in the Columbia River basin. Ecological Applications 13:964-989.

Waples, R.S. 1991. Genetic interactions between hatchery and wild salmonids: lessons from the Pacific Northwest. Canadian Journal of Fisheries and Aquatic Sciences. 48(Suppl. 1): 124-133.

Appendix A R-code used in estimation of proportion of hatchery-origin spawners

```
#Program to calculate properties of the estimate of the proportion of hatchery-origin spawners
#using Monte Carlo simulation and theoretical results. This code treats the general case of inputs
#from several hatcheries with potentially different visible marking (VM) fractions and different
#fraction of VM fish given a CWT.

#Variables and parameters used in the analysis
#inputs
#Nsims = total number of bootstrap replications
#Nnos = true natural origin spawning escapement
#Nhos = true hatchery origin spawning escapement escapement (hatchery-specific)
#theta = sampling fraction
#lambda = marking rate (lambda) (hatchery-specific)
#pcwt=fraction of VM fish that are also CWT (hatchery-specific)
#
#
#intermediate variables
#phos = fraction of escapement that is of hatchery origin
#phosi (true values) calculated from Nhos and Nhos
#nhatch=number of hatcheries supplying spawners in the wild
#Ehatchsampled = Replications of number of hatchery-origin fish that are sampled
#Enatsampled = Replications of number of natural-origin fish that are sampled
#Em = Replications of number of VM spawners (hatchery-specific)
#Emcwt = Replications of number of VM and CWT spawners (hatchery-#specific)
#Eu = Replications of number of un-VM spawners
#Emtot= Replications of the total number of VM fish (summing over hatcheries)
#Nhoshat = Replications estimate of Nhos
#Ntothat = Replications of estimate of Ntot (totally number of spawners)


#output variables
#phos (true value) calculated from Nhos and Nnos
#SE.Nnoshat = standard error (SE) of Nnoshat
#CV.Nnoshat = Coefficient of variation of Nnoshat
#SE.Nhoshat = standard error (SE) of Nhoshat
#CV.Nhoshat = Coefficient of variation of Nhoshat
#SE.phoshat = standard error (SE)
#CV.phoshat = Coefficient of variation
#BIAS.phoshat = relative bias
#the following use theoretical formulas
#SE2.Nhoshat = standard error (SE) of Nhoshat
#CV2.Nhoshat = Coefficient of variation of Nhoshat
#SE2.Nnoshat = standard error (SE) of Nnoshat
```

```
#CV2.Nnoshat = Coefficient of variation of Nnoshat
#SE2.phoshat = standard error (SE)
#CV2.phoshat = Coefficient of variation

#uses Monte Carlo simulation for multiple hatcheries
#uses cwt ratios to help esimate fractions of
# fish that are not visibly marked that orginate from from hatchery i

#Use Monte Carlo simulation for results
phos.mhatch.estimates1<-function(Nsims=10000, Nnos=200, Nhos=c(100,100),theta=0.25,
  lambda=c(0.75,.25),pcwt=c(.5,.9)){

#check dimension of inputs
k1<-length(Nhos);k2<-length(lambda);k3<-length(pcwt)
mytest<-abs(k1-k2)+abs(k2-k3)
if(mytest>0) stop("dimensions of Nhos, lambda, and pcwt must match")
nhatch<-length(Nhos)
#check lambdas (if they are all the same, the analysis simplifies)
if(nhatch==1){mytest==TRUE}
if(nhatch>1){mytest<-var(lambda)<1.e-10}
if(mytest){
#phis don't matter at all – it's as if there were a single hatchery
 res<-phos.estimates1(Nsims,Nnos=Nnos,Nhos=sum(Nhos),theta=theta,lambda=mean(lambda))
 phos=sum(Nhos)/(sum(Nhos)+Nnos)
 myres<-list(Nsims=Nsims,
                    Nnos=Nnos,
                    Nhos=Nhos,
                    theta=theta,
                    lambda=lambda,
                    pcwt=pcwt,
              phos=phos,
              SE.Nhoshat=res$SE.Nhoshat,
              CV.Nhoshat=res$CV.Nhoshat,
              SE.Nnoshat=res$SE.Nnoshat,
              CV.Nnoshat=res$CV.Nnoshat,
                    SE.phoshat=res$SE.phoshat,
                    CV.phoshat=res$CV.phoshat,
                    BIAS.phoshat=res$BIAS.phoshat)

 return(myres)
}

#check phis (must all exceed zero)
if(sum(pcwt==0))stop("phis must all be greater than zero")
phitest<-FALSE
```

```
if(sum(pcwt==1)==nhatch)phitest<-TRUE

 phos<-sum(Nhos)/(sum(Nhos)+Nnos)
#generate synthetic data sets
 Ehatchsampled<-matrix(NA,nrow=Nsims,ncol=nhatch)
 for(jj in 1:nhatch){
  Ehatchsampled[,jj] <-rbinom(Nsims,size=Nhos[jj],prob=theta)
 }
 Enatsampled <-rbinom(Nsims,size=Nnos,prob=theta)
 Em<-matrix(NA,nrow=Nsims,ncol=nhatch)
 Emcwt<-matrix(NA,nrow=Nsims,ncol=nhatch)

 for(ii in 1:Nsims){
  for(jj in 1:nhatch){
  Em[ii,jj]<-rbinom(1,size=Ehatchsampled[ii,jj],prob=lambda[jj])
  Emcwt[ii,jj]<-rbinom(1,size=Em[ii,jj],prob=pcwt[jj])
 }}

#total fish that are not visibly marked (summing over all hatcheries)
 Emtot<-apply(Em,c(1),sum)
 Eu<-apply(Ehatchsampled,c(1),sum)-Emtot+Enatsampled

 Nhoshat<-rep(NA,Nsims)
#Replications of estimates
if(!phitest){
 for(ii in 1:Nsims){
  Nhoshat[ii]<-get.nhoshat(x2=sum(Em[ii,]-Emcwt[ii,]),x1=Emcwt[ii,],theta,lambda,phi=pcwt)
}}else{
 for(ii in 1:Nsims){
  Nhoshat[ii]<- sum(Emcwt[ii,]/(theta*lambda))
}}

 Ntothat<-Eu*(1/theta)+Emtot*(1/theta)
 Nnoshat<-Ntothat-Nhoshat
 phoshat<-Nhoshat/Ntothat

#properties of phos estimator
SE.Nnoshat<-sqrt(var(Nnoshat,na.rm=T))
CV.Nnoshat<-SE.Nnoshat/Nnos
SE.Nhoshat<-sqrt(var(Nhoshat,na.rm=T))
CV.Nhoshat<-SE.Nhoshat/sum(Nhos)
SE.phoshat<-sqrt(var(phoshat,na.rm=T))
CV.phoshat<-SE.phoshat/phos
BIAS.phoshat<-(mean(phoshat,na.rm=T)-phos)/phos
```

```
myres<-list(Nsims=Nsims,
            Nnos=Nnos,
            Nhos=Nhos,
            theta=theta,
            lambda=lambda,
            pcwt=pcwt,
            phos=phos,
            SE.Nnoshat=SE.Nnoshat,
            CV.Nnoshat=CV.Nnoshat,
            SE.Nhoshat=SE.Nhoshat,
            CV.Nhoshat=CV.Nhoshat,
            SE.phoshat=SE.phoshat,
            CV.phoshat=CV.phoshat,
            BIAS.phoshat=BIAS.phoshat)
 return(myres)
}

#Theoretical results
phos.mhatch.estimates2<-function(Nnos=200,Nhos=c(100,100), theta=0.25,
  lambda=c(0.75,.25),pcwt=c(.5,.9)){

#check dimension of inputs
k1<-length(Nhos);k2<-length(lambda);k3<-length(pcwt)
mytest<-abs(k1-k2)+abs(k2-k3)
if(mytest>0) stop("dimensions of Nhos, lambda, and pcwt must match")
nhatch<-length(Nhos)
#check lambdas (if they are all the same, the analysis simplifies)
if(nhatch==1){mytest==TRUE}
if(nhatch>1){mytest<-var(lambda)<1.e-10}
if(mytest){
#phis don't matter at all – it's as if there were a single hatchery
 res<-phos.estimates2(Nnos=Nnos,
                      Nhos=sum(Nhos),
                       theta=theta,
                       lambda=mean(lambda))
 phos=sum(Nhos)/(sum(Nhos)+Nnos)
 myres<-list(Nnos=Nnos,
             Nhos=Nhos,
             theta=theta,
             lambda=lambda,
             pcwt=pcwt,
             phos=phos,
            SE2.Nnoshat=res$SE2.Nnoshat,
            CV2.Nnoshat=res$CV2.Nnoshat,
```

```
                SE2.Nhoshat=res$SE2.Nhoshat,
                CV2.Nhoshat=res$CV2.Nhoshat,
                SE2.phoshat=res$SE2.phoshat,
                CV2.phoshat=res$CV2.phoshat)
 return(myres)
}#mytest

#check phis (must all exceed zero)
if(sum(pcwt==0))stop("phis must all be greater than zero")
phitest<-FALSE
if(sum(pcwt==1)==nhatch)phitest<-TRUE
 phos<-sum(Nhos)/(sum(Nhos)+Nnos)

#theoretical formula for variance of phoshat
Ntot<-sum(Nhos)+Nnos
phosi<-Nhos/Ntot
if(!phitest){
 sum1<-sum(phosi*(1-theta*lambda*pcwt)/(theta*lambda*pcwt))
 sum2<-sum(phosi*(1-pcwt)/pcwt)
 sum3<-sum(phosi*(1-pcwt)*theta*lambda/pcwt)
 phos.var<-(1/Ntot)*(sum1-sum2*sum2/sum3-phos*phos*(1-theta)/theta)
 sum1<-sum(Nhos*(1-theta*lambda*pcwt)/(theta*lambda*pcwt))
 sum2<-sum(Nhos*(1-pcwt)/pcwt)
 sum3<-sum(Nhos*(1-pcwt)*theta*lambda/pcwt)
 Nhos.var<-sum1-sum2*sum2/sum3
 Nnos.var<-Ntot*(1-theta)/theta+Nhos.var-2*(1-theta)*sum(Nhos)/theta
}else{
 sum1<-sum(phosi*(1-theta*lambda)/(theta*lambda))
 phos.var<-(1/Ntot)*(sum1-phos*phos*(1-theta)/theta)
 Nhos.var<-sum(Nhos*(1-theta*lambda)/(theta*lambda))
 Nnos.var<-Ntot*(1-theta)/theta+Nhos.var-2*(1-theta)*sum(Nhos)/theta
}
 SE2.phoshat<-sqrt(phos.var)
 CV2.phoshat<-SE2.phoshat/phos
SE2.Nnoshat<-sqrt(Nnos.var)
 CV2.Nnoshat<-SE2.Nnoshat/Nnos
 SE2.Nhoshat<-sqrt(Nhos.var)
 CV2.Nhoshat<-SE2.Nhoshat/sum(Nhos)


myres<-list(Nnos=Nnos,
            Nhos=Nhos,
            theta=theta,
            lambda=lambda,
            pcwt=pcwt,
```

```
                phos=phos,
                SE2.Nnoshat=SE2.Nnoshat,
                CV2.Nnoshat=CV2.Nnoshat,
                SE2.Nhoshat=SE2.Nhoshat,
                CV2.Nhoshat=CV2.Nhoshat,
                SE2.phoshat=SE2.phoshat,
                CV2.phoshat=CV2.phoshat)
 return(myres)
}


#use iteration in x=g(x) method
#in general the estimate depends on the true values
#of escapement, so use iteration until the estimate converges
get.nhoshat<-function(x2,x1,theta,lambda,phi){
  etol<-1.e-5
  nhatch<-length(x1)
  Nhos0<-x1/(theta*lambda*phi)
  if(sum(c(x1,x2))<1.e-10)return(0.0)
  if((sum(x1)<1.e-10)&(x2>0))return(NA)
  run1<-sum(x1*(1-phi)/phi)
#initial guess
  Nhos<- Nhos0
  mynorm1<-sqrt(sum(Nhos*Nhos))
  err<-2.*etol*(mynorm1+etol)
  iter<-0
  while(err>etol*(mynorm1+etol)){
   rise<-Nhos*(1-phi)/(phi*theta)
   run<-sum(lambda*Nhos*(1-phi)/phi)
   Nhos<-Nhos0+(rise/run)*(x2-run1)
   mynorm2<-sqrt(sum(Nhos*Nhos))
   err<-abs(mynorm2-mynorm1)
   mynorm1<-mynorm2
   iter<-iter+1
   if(iter>100)stop("too many iterations in get.nhoshat")
  }
#  print(iter)
  return(sum(Nhos))
}


#special case where all lambdas are the same (Monte Carlo Results)
phos.estimates1<-function(Nsims=10000,Nnos=100,Nhos=100,theta=0.25,lambda=0.75)
{
 Ntot<-Nhos+Nnos
```

```
phos<-Nhos/Ntot
Ehatchsampled <-rbinom(Nsims,size=Nhos,prob=theta)
Enatsampled <-rbinom(Nsims,size=Nnos,prob=theta)
Em<-rep(NA,Nsims)
for(ii in 1:Nsims){
 Em[ii]<-rbinom(1,size=Ehatchsampled[ii],prob=lambda)
}
Eu<-Ehatchsampled-Em+Enatsampled

Nhoshat<-Em*(1/theta)*(1/lambda)
Ntothat<-Eu*(1/theta)+Em*(1/theta)
Nnoshat<-Ntothat-Nhoshat
phoshat<-Nhoshat/Ntothat
SE.Nhoshat<-sqrt(var(Nhoshat,na.rm=T))
CV.Nhoshat<-SE.Nhoshat/Nhos
SE.Nnoshat<-sqrt(var(Nnoshat,na.rm=T))
CV.Nnoshat<-SE.Nnoshat/Nnos
SE.phoshat<-sqrt(var(phoshat,na.rm=T))
CV.phoshat<-SE.phoshat/phos
BIAS.phoshat<-(mean(phoshat,na.rm=T)-phos)/phos

myres<-list(Nsims=Nsims,
            Nnos=Nnos,
            Nhos=Nhos,
            theta=theta,
            lambda=lambda,
            phos=phos,
            SE.Nnoshat=SE.Nnoshat,
            CV.Nnoshat=CV.Nnoshat,
            SE.Nhoshat=SE.Nhoshat,
            CV.Nhoshat=CV.Nhoshat,
            SE.phoshat=SE.phoshat,
            CV.phoshat=CV.phoshat,
            BIAS.phoshat=BIAS.phoshat)

return(myres)
}

#special case where all lambdas are the same (theoretical results)
phos.estimates2<-function(Nnos=100,Nhos=100,theta=0.25,lambda=0.75)
{
Ntot<-Nhos+Nnos
phos<-Nhos/Ntot
var.Nhoshat<-Nhos*(1-lambda*theta)/(lambda*theta)
var.Nnoshat<-Nnos*(1-theta)/theta+Nhos*(1-lambda)/(theta*lambda)
```

```
var.phos<-(phos/Ntot)*((1-lambda*theta)/(lambda*theta)-phos*(1-theta)/theta)
SE2.Nhoshat<-sqrt(var.Nhoshat)
CV2.Nhoshat<-SE2.Nhoshat/Nhos
SE2.Nnoshat<-sqrt(var.Nnoshat)
CV2.Nnoshat<-SE2.Nnoshat/Nnos
SE2.phoshat<-sqrt(var.phos)
CV2.phoshat<-SE2.phoshat/phos

myres<-list(Nnos=Nnos,
            Nhos=Nhos,
            theta=theta,
            lambda=lambda,
            phos=phos,
            SE2.Nnoshat=SE2.Nnoshat,
            CV2.Nnoshat=CV2.Nnoshat,
            SE2.Nhoshat=SE2.Nhoshat,
            CV2.Nhoshat=CV2.Nhoshat,
            SE2.phoshat=SE2.phoshat,
            CV2.phoshat=CV2.phoshat)

return(myres)
}
```

Appendix B Names of equivalent variables used in Rcode and the mathematical derivations.

Table B.1. —Names of equivalent variables.

R code	Mathematical derivation	Definition
Nhos	H	Total number of hatchery-origin spawners
phos	p	Proportion of hatchery-origin spawners
Nnos	W	Total number of wild-origin spawners
theta	θ	Sampling rate
pcwt	ϕ	CWT fraction
lambda	λ	VM fraction
Ntot	E	Total number of spawners (total escapement)