Evaluation of the LVI software and its manual entitled “LVI Linear Discriminant Analyses: Beta Version” by Ian Moss, PhD, RPF, Tesera Systems Inc.

Prepared for: Yuan, Xiaoping, PhD, RPF
Forest Analysis and Inventory Branch
Ministry of Forests, Lands and Natural Resource Operations
7th Floor, 727 Fisgard Street
Victoria, B.C., Canada

Prepared by: Temesgen Hailemariam, PhD, RPF
254-5525 West Boulevard
West Boulevard
Vancouver, B.C., V6M 3W6
Canada

May 15, 2013

Table of Contents

Evaluation of the LVI software and its manual entitled "LVI Linear Discriminant Analyses: Beta Version" by Ian Moss, PhD, RPF, Tesera Systems Inc.

Executive summary

The LVI software is intended to provide tools which enable practitioners and analysts to improve vegetation inventory attributes by imputing attributes from sampled to non-sampled polygons. A peer-review of the software and its accompanying manual was conducted by Dr. Temesgen Hailemariam (the author of this report). The specific objectives of the peer-review were: to assess the strengths and weaknesses of the software; including the level of confidence in the software results; assess the credibility of the software and manual for operational use; and suggest improvements.

During an intensive week period in May, the author reviewed selected R scripts, Phyton codes, and manuals, communicated with Dr. Ian Moss of Teresa Inc. who wrote the software and its manual, and reviewed software outputs using independent data set to determine the functionality and accuracy of the software.

Some of the strengths of the software and manual include:

- Codes are written in R and Phyton, which are freely available and widely used by the remote sensing and forest biometrics community. The R scripts are short and easy to interpret.

- Clear and complete directions are available for installation. Hence, practitioners and analysts are able to install software with ease.

- The structure of the software is generic and modular. It can be easily modified to include other types of analysis in the future.

- CVS files are selected as the standard data format for the LVI software. This makes it easier for practitioners and analysts to transfer and share files.

- The methods used to classify and develop categories are straightforward, and the statistical analyses employed are appropriate. The *Fuzzy c-means classification* used in the software is an efficient method for cluster analysis of remotely sensed or image data. The procedures used for variable selection and assigning weights for identifying nearest neighbors are lucid and enlightening.

- The classification and discrimination questions addressed in the software could also be looked at through a parametric lens. It is refreshing that Dr. Moss did not focus on parametric methods that have been examined for several decades. I commend him for staying away from the beaten path and for examining the use of multiple discriminant analysis to assign varying weights for selected variables.

- The approach taken in the software can easily be extended to improve accuracies of different tree-list generation methods at varying scales. When combined with ground data, the advanced clustering and discriminant algorithms used for variable selection might improve the accuracy of tree-lists generated using nearest neighbor methods.

I certainly felt that the software and its manual are worthwhile endeavors, providing useful information to analysts, despite some **notable weaknesses listed below**.

- The number of R routines is unnecessarily large. It could be condensed by writing functions or by developing a user-interface. Data dictionary is lacking.
- Correlation coefficients between or among X-variables are manually calculated.
- Current version of the software does NOT include ensemble modeling (e.g., Random Forests) and simulated annealing to select X-variables.
- While the codes or routines provide data for the display of maps, post-classification to display maps are not yet possible.
- Over fitting is well known problem is statistics and in developing models using remotely sensed data. Lacking from the exposition is the efficiency of the selected variable selection procedures in: a) mitigating the problem of over fitting; and b) comparison to other variable selection strategies.
- Converting continuous variables into class variables is known to result in loss of precision, but Fuzzy-C-means classification might mitigate this problem.
- The LVI software can be used by analysts who have some programming skills in R and Phyton. As it stands, the software might be somewhat difficult to be used by practicing foresters without some computing and programming skills.

Taking the above strengths and weaknesses of the software and its manual together, I assert that the LVI software has a great potential to contribute to forest inventory and analysis. The Beta Version is on the right track to provide sound and defensible vegetation inventory estimates at polygon levels, which can be the basis for forest inventory in British Columbia.

Level of confidence in software results: I have a high level of confidence in the software's output for selecting appropriate nearest neighbors and summarizing their results. The values reported in the output files the author checked are correct.

Use of the software for forest inventory*:* I felt that once the software and its manual are revised, they could be effectively used to improve vegetation inventory estimates at polygons levels. To use the software efficiently and operationally, I identified 13 conditions for future enhancements. Seven of them can be addressed in six months to one year. The time required for the other six conditions will vary depending on the Forest Analysis and Inventory's future direction.

Overall, Dr. Moss has produced professional and functional software that handles the difficult problems of classification, discriminant analysis, and nearest neighbor identification. The thinking behind the software and its manual, and the approaches taken to improve procedures for estimating polygon attributes and to impute multivariate response variables are well developed—indeed commendable.

1.0 General comments:

Forest resource management is currently faced with a myriad of challenges. Solutions to these challenges require quantitative approaches in order to reach sound and broad-based solutions. Forestry has become increasingly quantitative in its approaches to research and sustainable management. Rising forest values increase the demand for accuracy and precision in quantitative approaches for management prescriptions and projected outcomes. Both trends have magnified the importance of precise classification of categories and imputation. The LVI software and its manual outlined selected classification, discriminant analysis, and nearest neighbor selection strategies to improve future forest inventory and mapping accuracy that are critical in tackling these challenges.

A large set of predictor variables are commonly acquired from remote sensing data. This poses an issue of how to select the optimal set of predictor variables to be included in a model. Variable selection has been addressed in several statistical methods including linear regression (Venables and Ripley, 2002), parametric regression (Murtaugh, 2009), nonparametric regression (Kulasekera, 2001), and additive models (Xue, 2009). Variable selection is not only an issue in linear regression, but it is also critical to nonparametric models, particularly in nearest neighbor (NN) imputation. Unfortunately, the procedures used for parametric methods cannot be used in the NN imputation because the NN methods are fundamentally different. For instance, model accuracy in training data does not automatically improve as more predictor variables are added, and the definition of model complexity is not straightforward. Yet, Packalen et al. 2012 asserted that in many cases variable selection is more important than the method used to select NN.

This project focused on the selection of predictor variables to NN imputation to predict both discrete and continuous variables. There are not many studies that address this issue, but there has been some research in machine learning for selecting variables that can be used in classification model (Guyon et al., 2004). A typical example of a classification task where variable selection plays an important role is that of gene selection from microarray data (Guyon and Elisseeff, 2003).

In addition to variable selection algorithms, predictors can be selected based on the correlation of X and Y or some other metric that ranks predictors (e.g., Z-scores). Selection of variables is often done before selecting NN, or the user may try different variable combinations in actual imputation. An assessment of different variable combinations is very time consuming, and automating the variable selection process makes the NN imputation process efficient. That is the context in which the LVI software is designed; to automate the variable selection and the NN selection and summarization processes.

2.0 Contents of current software and manual:

Based on the assumption that user's are competent in the use of computers and have some experience in programming, the manual is organized into 17 sections. It covers instructions to install R and Python models, gives a brief introduction on kNN and its integration with LVI, and detailed instructions for a number of specific analyses by specifying associated R codes and Python scripts. For example, Section 3 embodies the major part of Fuzzy C-Means Classification whereby the Y-variables are assigned to different classes, while Section 4 outlines the process used to identify the key Y-variables of interest and unique combinations of the X-variables that are likely to be useful in terms of explaining the similarities and differences amongst the observed combinations of Y-variables.

Sections 5 and 6 contain the procedures used to run discriminant analysis to produce Z-scores for single and multiple X-variables. Using the reference dataset to determine the best set(s) of X-variables for explaining similarities and differences amongst the Y-variables and identify appropriate variable weights, these sections culminate with Cohen's coefficient of agreement.

While Section 7 provides detailed instructions to identify Pearson's correlations amongst X- or Y-variables and to produce a unique X-variable sunset correlation matrix, Section 8 outlines procedures for visualizing data by producing box-plots and scatter grams.

Section 9 combines several output files produced in the previous sections into a single file for evaluation of results and for overall assessments. Section 10 outlines alternative variable selection methods that search for subsets which are optimal for imputing the response variables

Sections 11 and 12 apply the discriminant functions to the reference data set to produce Z-Scores and define the X-variables (respectively) to identify nearest neighbor, and generate Root Mean Squared Errors for k-Nearest Neighbors. Three distance metric (Euclidean, Mahalanobis, Absolute Difference) are applied to the selected X-variables, either before or after having applied the variable Z-scores. Using the reference data set, from 1 to k nearest neighbors is generated and root mean squared errors are calculated.

Section 13 provides the Python code used to use the Y-variables to identify nearest neighbor, and generate Root Mean Squared Errors for k-Nearest Neighbor. The summary of the analyses results including bias and root mean squares can be used to assess the quality of variable subsets as surrogates for a full data set and determine the reliability of the final results. By keeping the Y-variables constant across all variable sets, this step establishes a lower bound in terms of the Root Mean Squared Errors that can be obtained based on the sample population and therefore can be used as a benchmark against the results. *It represents the best results that can potentially be obtained given the sample population*.

Section 14 identifies k-NN in target data set for k = 1,2,3, … n, nearest neighbors associated with each target observation, based on the use of discriminat function Z-Scores. Following that Sections 15 and 16 summarize and format the Y-variable statistics for each target observation. A list of references in Section 17 is followed by Appendix I that lists 28 Python scripts and 40 R codes and Appendix II that provides a step-by-step demonstration of the "Standard LVI kNN process".

In sum, the manual describes several procedures and statistical methods used to evaluate various combinations of X- variables, variable weights, and distance metrics to determine the best combination for estimating the Y-variables. After selecting variables, variable weights, and distance metrics to match between 1 and k nearest neighbors from the reference set with each observation in the target set, the LVI software produce more than 48 input and output data files that can be imported in GIS database products to produce maps, compile inventory estimates, and conduct further analysis.

3.0 Technical analysis of components and assessments

3.1. Software

3.1.1. Testing routines and codes

The Tally Lake data set used by Stage and Crookston (2007) was used to examine some of the outputs produced by R codes and Phyton scripts in the LVI software. The data represented 8 response (Y set) and 21 predictor (X set) variables (Table 1). Summaries of attributes for 847 polygon-based reference stands (polygons) in Flathead National Forest, Montana, USA are given in Table 1.

Table 1. Summary of the attributes used to examine selected R and Python codes. Stage and Crookston (2007) used the data to demonstrate partitioning of error components and related statistics.

Variable	Attributes	Min	Mean	Median	Max	Std
Ground based measurements of trees (Y-variables):						
TopHt	Height of tallest trees (ft)	12	75.3	79	150	23.8
LnVolL	Log of the volume (ft^3/acre) of western larch	0	5.1	6	8.8	2.6
LnVolDF	Log of the volume (ft^3/acre) of Douglas-fir	0	5.6	6.7	8.7	2.7
LnVolLP	Log of the volume (ft^3/acre) of lodgepole pine	0	5	6	9.2	2.9
LnVolES	Log of the volume (ft^3/acre) of Engelmann spruce	0	3.1	2.7	8.8	3
LnVolAF	Log of the volume (ft^3/acre) of alpine fir	0	3.6	4.4	8.6	2.9
LnVolPP	Log of the volume (ft^3/acre) of alpine fir	0	0.1	0	8.1	0.8
CCover	Canopy cover (percent)	0	64.7	65	100	14.9
Geographic Location, Slope, and Aspect (X-variables):						
utmx	UTM easting at plot center	223476	230970	230546	240143	3543
utmy	UTM northing at plot center	5E+06	5363321	5363793	5376540	7088.6
elevm	Mean elevation (ft) above sea level over plot	940.6	1410.8	1431.5	1894	184
eevsqrd	(elevm - 1600)2	0.1	69599.6	29666.3	434764	90316.6
slopem	Mean slope (percent) over plot	0	13.2	13.1	32.2	5.1
slpcosaspm	Mean of slope (proportion) times the cosine of aspect	-0.5	0	0	0.5	0.1
slpsinaspm	Mean of slope (proportion) times the sine of aspect	-0.5	0	0	0.5	0.2
Additional X-variables:						
ctim	Mean of slope curviture over pixels in stand	0.1	0.9	0.8	3	0.6
tmb1m	Mean of LandSat band 1 over pixels in stand	46.8	50.7	50.2	61.7	2.2
tmb2m	Mean of LandSat band 2 over pixels in stand	32.4	38	37.5	52.1	3.4
tmb3m	Mean of LandSat band 3 over pixels in stand	23.5	29.7	28.6	51.2	4.2
tmb4m	Mean of LandSat band 4 over pixels in stand	45.6	70.5	69.2	112.8	12
tmb5m	Mean of LandSat band 5 over pixels in stand	26	46.1	43.8	103.6	12
tmb6m	Mean of LandSat band 6 over pixels in stand	16	26.4	24.9	61.7	6.7
durm	Mean of light duration over pixels in stand	2634.2	3697.4	3733.2	4152.1	239.3
insom	Mean of solar insolation over pixels in stand	799856	1188043	1189111	1422968	101040.8
msavim	Mean of AVI for pixels in stand	47.1	51.6	51.3	60.9	2.4
ndvim	Mean of NDVI for pixels in stand	32.1	38.4	38	50.6	3.3
crvm	Mean of slope curviture for pixels in stand	24	31.2	30.3	52.7	4.5
tancrvm	Mean of tangent curvature for pixels in stand	25.2	45.8	44	97	11.6
tancrvsd	Standard deviation of tangent curvature for pixels in sta	1.8	10.8	10.1	35.1	4.7

While it is hard (if not impossible) to evaluate all codes or routines in a short period of time, I examined selected codes for their functionality and accuracy. My instinct tells me that if the basic statistics and attributes are correct, the chances of major flows in the software are lower. With that spirit, I examined selected codes and scripts (Appendix I) using the test data. Some of the basic statistics and output files examined include:

- UCORCOEF
- UNIQUEVAR
- VARMEANS
- XCOV
- MINMAXCORR
- NNTARGETxx
- YCOV
- ZERMSE
- XSCORE
- XSTAT
- XNNA
- XRMSE

The values obtained using the yaImpute and other R codes I wrote to test the LVI software are identical to those reported by the LVI software. Because of differences in selecting nearest neighbor between the yaImpute and the LVI software, there were minor differences in the selected nearest neighbors.

While analysis of coding efficiency, design, and organization are outside of the scope of this report, I have a high level of confidence in LVI software's output for selecting appropriate nearest neighbors and summarizing their values. The values reported in the output files the author checked are correct.

3.1.2. Technical analysis

A. To select the X-variable set, the Y-variables were classed into groups using Fuzzy C-Means Classification. Multiple discriminant analysis (MDA) was used to assign different weights to the selected X-variables, which are used to select nearest neighbors for each target polygon. While MDA maximizes *the ratio of the between and within group sum of squares by applying* varying weights, it is <u>unknown</u> if classification followed by discriminant analysis provides better results than methods that select variables directly. One can argue that converting continuous variables to discrete or class variables might lead to loss of information or lower precision. It is critical to <u>quantify</u> the gain or loss in precision by using classification followed by discriminant analysis versus direct variable selection.

B. One of the reasons for using Fuzzy C-Means Classification and MDA is to select variables and apply weights. The classification followed by discriminant analysis and nearest neighbor selection approach might improve the performance of NN imputation under some or most conditions. The benefits of using this approach over methods that select predictors based on minimum error or canonical correlation analysis (CCA) are not clearly stated. Questions that remain are:

- Could approaches that minimize RMSEs (e.g., Simulated Annealing, SA) by selecting different neighbors repeatedly achieve the same level of performance? Could a cost function that minimizes by taking into account the multivariate nature of the response and the X-variable set perform equally? Please see Packalen et al. 2012 for further details on some of the approaches used in selecting variables and nearest neighbors.

- When most response variables (Y-set) are continuous, could SA or other variable selection methods that maximize the canonical correlation perform equally well or better?

- What are some of the assumptions or conditions the classification and MDA require? Is multivariate normality required? Please specify assumptions if there are any.

C. The objective of this project was to select predictor variables for NN imputation with one or more discrete or continuous response variables. Up to k-Nearest neighbours were assigned from the reference dataset to each of the target observations, based on the use of discriminate function Z-Scores and a given number of variables. Lacking from the exposition is the issue of unrealistic model accuracy caused by potential overfitting. The problem of **overfitting** where a model adjusts to specific random features or noise of the training data but works poorly on other datasets. Overfitting is well known in statistic and machine learning (Reunanen, 2003; Hastie et al., 2009). Lacking in LVI software or its manual is the use of k-fold cross validation or separate test dataset to examine the performance of the approaches outlined in the software.

D. Dr. Moss completed a great deal of work in a very short period. His dedicated efforts over the last several months have resulted in a number of impressive approaches and outputs. Some of these include:

- Leave-one-out cross-validation is a special case of k-fold cross validation in which k is equal to the number of observations. Take-one-Leave one approach enables Cohen coefficients of agreement which can be used for classification accuracy statistics. Summary statistics for classification accuracy, including producer's and user's accuracy, are excellent. I find the values reported in CTABSUM.csv very useful in determining the number of X-variables to be used in identifying nearest neighbors.

- The focus of the project was prediction accuracy, not reproducing variance structure observed in the observations. Hence, the use of Z scores is appropriate and defensible.

- Accuracy measures provided on Page 21 are correct and useful.

- Producing a unique X-variable subset correlation matrix that yields min-max correlation matrix is helpful to do variable screening.

E. Graphics are well designed and rendered to enhance classification and imputation efforts. The *X- variables vs. the classification Box*, the *X-variable scatter plots*, and the *Y vs. Unique X-variables* scatter plots are informative and depict the fundamental data structure well.

3.1.3. Logic and consistency

A. The methods used to classify and develop categories are straightforward, and the statistical analyses employed are appropriate. The *Fuzzy C-Means Classification* used in the software is an efficient method for cluster analysis of remotely sensed or image data. The procedures used for variable selection and assigning weights for identifying nearest neighbors are lucid and enlightening.

B. Assessment methods including Z-scores are appropriate and suited to project goals.

C. Graphics are consistent, appropriate and designed to optimize learning. The R codes provided in Section 8 of the manual are straight forward and efficient for processing a large number of variables.

3.1.4. Adaptability and modularity

A. Most aspects of the software can be easily integrated into other classification and imputation routines.

B. The LVI software is flexible. The structure of the software is generic and modular. It can be easily modified to include other types of analysis (e.g., Random Forest, Systems of Equations, etc.) in the future.

C. Clear and complete directions are available for installation. Practitioners and analysts are able to install software with ease.

3.2. Manual

3.2.1. Contents

A. Content and context are accurate, current, and consistent with the theme. All information relates to the stated purpose and project goals.

B. Uses real examples to make the software relevant for practitioners/analysts. It might motivate practitioners or analysts to learn and master classification and imputation concepts.

C. The objectives and framework used in the LVI software and the yaImpute package (commonly used in the forestry literature) are different. Unlike the yaImpute package the LVI software selects variables before selecting nearest neighbors. Yet, having a section that outlines the main differences and similarities between them adds clarity and helps readers understand their potential and limitations.

D. Background and text are pleasing, compatible and easy to read. It is easy to navigate through the manual to find R codes and the Phyton script to find functions or routines.

3.2.2. Layout

A. Layout is clear and logical. Directions are clear and complete enough for practitioners or analysts to find what they need and perform required tasks.

B. Lists all prerequisite skills to run the software.

C. Layout is consistent. Paragraphs and sections have clear and accurate informative headings. Clear and clean fonts are used.

3.2.3. Technical

A. The manual lacks some of the theoretical underpinnings for using classification followed by discriminant analysis and nearest neighbor selection. I suggest including some theoretical justifications in the manual, and providing specific page number(s) for the Dillon and Goldstein (1984) citation.

B. In Section 10, Step 8 requires that the "*RunLinearDiscriminantAnalysis_klaR_pvs"* code be run. The *pvs* command in *klaR* requires at least 5 observations per group to perform pairwise variable selection. This requirement needs to be stated as a caveat or be discussed in the manual. This would help analysts to consider the distributions of their data before they select the *pvs* or other methods for their discriminant analysis.

C. Page 1 of the manual asserts that "*The above mentioned process is a form of classification. At the extreme, when k = 1, each observation represents a single class. As k increases (k > 1), the number of classes remains the same (equal to the number of observations) but the classes are based on some degree of overlapping sets of observations (O) in the reference data calculated as follows:*

$$O = (k-1)/Nr$$

where, Nr is the total number of observations in the reference dataset."

I am uncertain if the equation holds in some circumstances. For example, when k=2 then O=1/Nr. Would this represent the degree of overlapping set of observations? If so, it can be easily influenced by Nr. If Nr is extremely large (as Nr→ ∞), O goes to 0. Would that mean there is no degree of overlapping when there is a large number of reference data. Please expound.

4.0 Statement of findings

4.1. Strengths of the software and its documentation:

Some of the strengths of the software and its manual include:

Software:

A. Codes are written in R and Phyton, which are freely available and widely used by the remote sensing and forest biometrics community.

B. Clear and complete directions are available for installation. Hence, practitioners/analysts are able to install software with ease.

C. The structure of the software is generic and modular. It can be easily modified to include other types of analysis (e.g., Random Forest, Systems of Equations, etc.) in the future.

D. CVS files are selected as the standard data format for the LVI software. This makes it easier for practitioners and analysts to transfer and share files.

E. The methods used to classify and develop categories are straightforward, and the statistical analyses employed are appropriate. The *Fuzzy c-means classification* used in the software is an efficient method for cluster analysis of remotely sensed or image data. The procedures used for variable selection and assigning weights for identifying nearest neighbors are lucid and enlightening.

F. Including the 'Take-one-Leave-one option" in the multiple linear discriminant analysis enables analysts to rate the quality of the variable set for the purpose of classifying and producing contingency tables. The posterior probabilities produced using 'Take-one-Leave-one option" are better than those produced with them.

G. The example data used in the analyses are reasonable. The input data included Landsat 5 TM imagery, VRI, and Predictive Ecosystem Mapping. The variable selection routines used to examine over 300 combinations of variables are logical and efficient.

H. The classification and discrimination questions addressed in the software could also be looked at through a parametric lens (e.g., logistic, etc.). It is refreshing that Dr. Moss did not focus on parametric methods that have been examined for several decades. I commend him for staying away from the beaten path and for examining the use of MDA to assign varying weights for selected variables.

I. Using LVI data collected in Quesnel, Dr. Moss demonstrated how the *Fuzzy c-means classification,* MDA, and nearest neighbor selection is performed using the LVI software. His pioneering work might enable practitioners to improve the prediction of Phase I photo variables which are the basis for forest inventory in British Columbia.

J. The approach taken in the software can easily be extended to improve accuracy of different tree-list generation methods at varying scales.

Manual:

K. After examining different distance metrics and different sets of Y and X variable sets, the codes or routines provide data sets that can be used to display maps.

L. The 17 chapters/sections are well written, with clear and concise presentations of the data, the methods, and the results. Building on each preceding chapter, the contents of the manual motivate practitioners and analysts to learn and master classification and imputation concepts.

M. The manual is logically arranged. The contents of the manual are clearly and sufficiently identifiable in each of the seventeen chapters and two appendices. The chapters compose clear and coherent topics.

4.2 Weaknesses of the software and its documentation:

The software and its manual are at their inception stages, and will evolve over time. Future enhancements of the software, its manual and the imputation process need to consider the following.

Software:

A. The number of R routines is unnecessarily large. It could be condensed by writing functions or by developing a user-interface. Data dictionary is lacking.

B. Correlation coefficients between or among Xs are manually calculated. It should be automated.

C. Current version of the software does NOT include ensemble modeling including Random Forests.

D. While the codes or routines provide data for the display of maps, post-classification to display maps are not yet possible.

E. Over fitting is a well-known problem in statistics and in developing models using remotely sensed data. Lacking from the exposition is the efficiency of the selected variable selection procedures in: a) mitigating the problem of over fitting; and b) comparison to other variable selection strategies (e.g., optimization-based variable selection methods). Some analysis or discussion on these topics will shed some light to practitioners and analysts.

F. Converting continuous variables into class variables is known to result in loss of precision (accuracy), but Fuzzy-C-means classification might mitigate this problem. The jury is out if classification followed by discriminant analysis and nearest neighbor selection provides better results than direct variable selection followed by nearest neighbor selection.

G. Bootstrapping is a potential method to impute categorical response variables, but it was not considered in the current version of the software or manual.

H. The LVI software can be used by analysts who have some programming skills in R and Phyton. As it stands, the software is somewhat difficult to be used by practicing foresters without some computing and programming skills.

Manual:

I. While the manual is complete and comprehensive, the structure of the document could be improved by reorganizing and rewriting some of the sections. The manual is lengthy and could be condensed. For example, Sections 5 and 6 can be combined without losing clarity.

J. The manual lacks comparative analysis that shows the gain made by using classification, discrimination and identification of nearest neighbors.

K. The manual, though in places somewhat tedious owing to the subject matter, is generally unambiguous and informative. Yet, the manual is cryptic and unclear at times, discouraging less savvy users and analysts from using the software.

Taking the above strengths and weaknesses of the software and its manual together, I assert that the LVI software has a great potential to contribute to forest inventory and analysis. The Beta Version is on the right track to provide sound and defensible vegetation inventory estimates at polygon levels, which can be the basis for forest inventory in British Columbia.

5.0 Future enhancement for the imputation process and accompanying software

While the LVI software is the state-of-the-art in combining and automating the process of classification, discriminant analysis, and nearest neighbor identification, it needs some revisions and thought. Software development is a continual exercise and process, not a discrete event. In updating the software, differentiate between versions and keep track of them in an automated fashion, using revision control or a documentation system.

Below are some issues to consider in enhancing the LVI software.

5.1. Technical and analytical issues

A. It is important to select X-variables using more than one method. Other approaches worth examining include ensemble modeling in which RF importance is used to select variables, and optimization-based approaches (i.e., Simulated Annealing) which commonly minimize RMSE. In addition, comparison of the performance of these methods under varying number of predictor and response variables is warranted.

B. The variable selection process does not consider the underlying distribution of the data including the sampling, calibration, and testing routines used in separating datasets into calibration and testing datasets.

C. Examine the use of optimized distance metric in identifying nearest neighbor. This corresponds to learning a distance metric in which the different axes (predictors) are given different weights (Xing et al., 2002). A similar approach was used by Franco-Lopez et al., (2001) and Haapanen and Tuominen (2008), which minimized RMSE using the Nelder and Mead (1965) simplex search, while Tomppo and Halme (2004) used a Genetic Algorithm. While this type of metric is generally rare in NN imputation literature, it is worth examining if it improves the performance of the LVI software.

5.2. Software

5.2.1. Design

A. Add a routine to identify notably different observations with larger error. Tabulate notably different observations that show large differences between observed and imputed values. This will help to identify outlier observations or minor observations. Page 5 of the manual asserts that "*Finally there is one other significant consideration in nearest neighbour types of analyses: What to do with outliers? Certain reference observations can often be identified as making a proportionately large contribution to errors of estimation of the Y-variables in the final outcome. These observations should be removed from the reference dataset*." While the above assertion is mostly correct, there are scenarios where legitimate observations might be much different from the rest of the population. These scenarios might include unique stands with minor species, or polygons with unique attributes ("rare polygons"). In these instances, the use of a routine that identifies notably different observations with larger error will help to identify whether an observation is an outlier or not.

B. Add a routine to identify most commonly used reference observations. Tabulate references most often used in imputation – a matrix of reference observations that are used most often as sources of imputation might help in assessing the quality of imputation used.

C. For large data sets, Steps 11 and 12 (pages 12 and 13) are computationally slow. One can ask if there is a need to produce all the tables reported in these steps.

D. Automate the computation of correlation coefficients between or among X-variables.

5.3. Manual

A. The manual needs to document some theoretical underpinnings on classification, discriminant analysis, and nearest neighbor identification.

B. Use a common example data set like the IRIS data to make the manual easier to follow and to make it friendlier for practitioners and analysts with limited programming skills.

C. Include an index for quick searches of keywords.

5.4. Publication in peer-reviewed journals

The contents reported or computed in the LVI software are lead-edge research topics and undoubtedly contribute to the existing body of knowledge of classification, discriminant analysis and imputation. I suggest the author consider publishing part of the software and associated expositions in peer-reviewed journal and contribute to the existing body of knowledge.

6.0 Suggested priorities for the imputation process and accompanying software

6.1. Short term

A. Given that Section 10 (*Alternative Variable Selection Procedure*) is under development, the variable selection process should be considered in tandem with sampling, calibration, and testing routines where the calibration and testing datasets are separated.

B. It is important to select X-variables using more than one method. There is a need to adopt or develop an estimate of variable importance and also consider other methods including ensemble modeling and simulated annealing approaches**.**

C. The process for dealing with correlation amongst the variables should be automated. In particular, consideration should be given to regressing each X-variable on the remaining X-variables within a dataset and removing those where the regression correlation coefficient is above (or below if negative) a certain user-defined threshold.

D. The various routines should be wrapped up into single routines to reduce the number of steps. This might also involve development of an easier to use user-interface.

E. A process should be developed to better facilitate the construction of data dictionaries used in the process, particularly as it relates to the reference and target datasets. In line with this topic, the data dictionaries that are to remain unchanged should be separated from those involving the target and reference datasets.

F. Add a routine to identify notably different observations with larger error. Tabulate notably different observations that show large differences between observed and imputed values. This will help to identify those observations in the target dataset that are outside of the range contained within the reference dataset, and so too, those types or combinations that are under-represented in the reference dataset relative to those in the Target dataset. This would be very useful in identifying locations suitable for establishment of additional samples so as to ensure adequate representation of the range of variation.

G. The manual should be re-organized. The content is mostly there, but it should be rearranged so that the workflows are presented in a more orderly fashion.

6.2. Long term

A. The software has a modular approach. This allows the possibility of extending the software in a number of different directions with more options and basis for comparisons of results with the goal of producing the best map possible.

B. There are some recognized problems with k-NN methods. They tend to be highly biased at the edge of the data cloud because prediction sites will likely be paired with a more central sample value due to asymmetric neighborhoods. Extremely small values and extremely high values will be over- and underestimated, respectively, if the sample data do not cover the whole range of variability (Packalen and Maltamo 2007). Bias can also be a problem in the interior of the data cloud if the covariates are nonuniformly distributed (Stage and Crookston 2007). Thus, examinations of geo-statistical approaches (e.g., universal kriging) that are unbiased and consider spatial autocorrelation are warranted.

C. In k-NN, a substitute observation or polygon is found using the covariates that are available for every site. As the number of samples increases, there will be a higher chance of getting an exact match in the covariates. However, this only guarantees an exact match for the response variables if there is perfect (or very strong) correlations with the covariates. As sample size increases, there is no guarantee that the mean of the response variables will approach the true mean and hence NN methods are not statistically consistent (LeMay and Temesgen 2005). Hence, it is important to extend the list of attributes used as X-variables, particularly those relating to species and site productivity, including topographic index (relating to soil moisture), slope position at different scales, and climate indices: http://www.genetics.forestry.ubc.ca/cfcg/ClimateWNA/ClimateWNA.html

D. With a small degree of modification the LVI software could be made completely generic such that it could be applied generally to processes involving classification, discriminant analysis, and nearest neighbor types of analyses, either alone or in combination. The approach used in the LVI software could be equally well applied to generate tree lists.

E. Relevant accuracy statistics for assessing the quality of predictions of categorical variables are still lacking (Tomppo et al 2009). However, there are recent proposals for a variance estimator that incorporates spatial correlation (McRoberts et al. 2007), model-based estimators of the uncertainty (Magnussen et al. 2009) and design-based approaches to derive the statistical properties of the k-NN predictions (Baffetta et al. 2009). The criteria for variable selection listed on Page 13 are very informative and appropriate, but they might fail for zero-inflated data or for attributes with highly skewed distributions. Hence, some investigation to assess the quality of variable selection under this scenario is warranted.

7.0 Overall Recommendation:

The software and its manual are interesting and solid. Building on each preceding chapter, the manual has contributed to the existing body of knowledge. Through its demonstration data sets and open-source codes, the software is manageable to run. The work has exceeded my expectations.

In sum, the classification, discrimination, and imputation questions addressed in the software could also be looked through a parametric lens (e.g., logistic regression or stepwise variable selection, etc.). It is refreshing that Dr. Moss did not focus on parametric methods that have been examined for several decades. I commend him for staying away from the beaten path and for using *Fuzzy c-means classification* to classify and then use discriminant analysis and k-NN methods for imputing polygon attributes. The software and its manual will help to improve procedures for estimating polygon attributes and to impute multivariate response variables. In light of this, I strongly recommend that Dr. Moss be granted additional resources to make the software operational.

Thank you for the opportunity to review this enlightening report and software. I look forward to seeing its updated version in the near future.

References

Baffetta F, Fattorini L, Franceschi S, Corona P. 2009. Design-based approach to k-nearest neighbours technique for coupling field and remotely sensed data in forest surveys. Remote Sensing of Environment 113: 463–475.

Guyon, I., and Elisseeff, A. 2003. An introduction to variable and feature selection. The Journal of Machine Learning Research, Vol. 3, pp. 1157_1182.

Guyon, I.M., Gunn, S.R., Ben Hur, A., and Dror, G. 2004. Result analysis of the NIPS 2003 feature selection challenge. In Advances in Neural Information Processing Systems 17. Edited by Saul, L.K., Weiss, Y., and Bottou, L. MIT Press, Cambridge, MA, pp. 545_552.

Goerndt, M., V. Monleon and H. Temesgen. 2011. A comparison of small-area estimation techniques to estimate selected stand attributes using LiDAR-derived auxiliary variables. *Can. J. For. Res.* 41:1189-1201.

Kim, H.-J. and E. Tomppo. 2006. Model-based prediction error uncertainty estimation for k-NN method. Remote Sensing of Environment 104: 257-263.

Kulasekera, K.B. 2001. Variable selection by stepwise slicing in nonparametric regression. Statistics & Probability Letters, Vol. 51, pp. 327_336. doi: 10.1016/S0167-7152(00)00167-X.

LeMay, V. and H. Temesgen. 2005. Comparison of nearest neighbor methods for estimating basal area and stems per ha using aerial auxiliary variables. Forest Science 51:109-119.

McRoberts, R., E.O. Tomppo, A.O. Finley and J. Heikkinen. 2007. Estimating areal means and variances of forest attributes using the k-nearest neighbors technique and satellite imagery. Remote Sensing of Environment 111:466-480.

Magnussen, S., R.E. McRoberts and E.O. Tomppo. 2009. Model-based mean square error estimators for k-nearest neighbour predictions and applications using remotely sensed data for forest inventories. Remote Sensing of Environment 113: 476–488.

Murtaugh, P.A. 2009. Performance of several variable-selection methods applied to real ecological data. Ecology Letters, Vol. 12, No. 10, pp. 1061_1068. doi: 10.1111/j.1461-0248.2009.01361.x.

Packalen, P., Temesgen, H., and Maltamo, M. 2012. Variable Selection Strategies for Nearest Neighbor Imputation in Remote Sensing Based Forest Inventory. Canadian Journal of Remote Sensing. 38(5): 557-569.

Packalén, P. and M. Maltamo. 2007. The k-MSN method for the prediction of species-specific stand attributes using airborne laser scanning and aerial photographs. Remote Sensing of Environment 109: 328-341.

Stage, A.R.; Crookston, N.L. 2007. Partitioning error components for accuracy-assessment of near neighbor methods of imputation. For. Sci. 53(1):62-72.

Temesgen, H. 2003. Estimating tree-lists from aerial information: a comparison of parametric and most similar neighbor approaches. Scandinavian Journal of Forest Research 18:279-288.

Tomppo E, Gagliano C, De Natale F, Katila M, McRoberts R (2009) Predicting categorical forest variables using an improved k-nearest neighbor and landsat imagery. Remote Sensing of Environment 113: 500–517.

Venables, W.N., and Ripley, B.D. 2002. Modern Applied Statistics with S-plus, 4th edition. Springer-Verlag, New York. 495 p.

Wang, T., A, Hamann, D. Spittlehouse and S. Aitken. 2006. Development of scale-free climate data for western Canada for use in resource management. International Journal of Climatology 26: 383-397.

Xue, L. 2009. Consistent variable selection in additive models. Statistica Sinica, Vol. 19, pp. 1281_1296.

Appendix I – R codes used to examine selected outputs, summary statistics, and codes.

Running Section 4 of the LVI manual

```
#Set Working Directorty for each session
setwd("F:\\Rwd")
getwd()
save.image(file="OpenSession.RData")

#source('F:\\Rwd\\RScript\\installPackages.r')
source('F:\\Rwd\\RScript\\LoadDatasetAndAttachVariableNames.r')
nLviRows

#Optional
#source('F:\\Rwd\\RScript\\SelectObservationSubset.r')
#nLviRows

source('F:\\Rwd\\RScript\\DeclareClassificationVariableAsFactor.r')
nClasses

source('F:\\Rwd\\RScript\\SelectXVariableSubset_v1.r')
XVARSEL

source('F:\\Rwd\\RScript\\Loadsubselect-R-Package.r')

source('F:\\Rwd\\RScript\\RunLinearDiscriminantAnalysis_subselect_ldaHmat.r')

source('F:\\Rwd\\RScript\\Run-ldaHmat-VariableSelection-Improve.r')

source('F:\\Rwd\\RScript\\ExtractVariableNameSubsets.r')

source('F:\\Rwd\\RScript\\WriteDataframeToCsvFile.r')

system('python F:\\Rwd\\Python\\EXTRACT_RVARIABLE_COMBOS.py')        #To import a Phyton
executable. Yet, you stilll need to run phyton fron C:\ directory or programs
```

```
#Running Section 5 of the LVI manual
#Set Working Directorty for each session
setwd("F:\\Rwd")
getwd()
save.image(file="OpenSession.RData")

source('F:\\Rwd\\RScript\\LoadDatasetAndAttachVariableNames.r')
#source('F:\\Rwd\\RScript\\SelectObservationSubset.r')
source('F:\\Rwd\\RScript\\DeclareClassificationVariableAsFactor.r')
source('F:\\Rwd\\RScript\\ComputeUniformPriorClassProbilityDistribution.r')
source('F:\\Rwd\\RScript\\ComputeSamplePriorClassProbabilityDistribution.r')

source('F:\\Rwd\\RScript\\SelectXVariableSubset_v1.r')
source('F:\\Rwd\\RScript\\SelectXVariableSubet_v2.r')

source('F:\\Rwd\\RScript\\LoadMASS-R-Package.r')
source('F:\\Rwd\\RScript\\RunLinearDiscriminantAnalysis_MASS_lda.r')
source('F:\\Rwd\\RScript\\RunLinearDiscriminantAnalysis_MASS_lda_TakeOneLeaveOne.r')
```

```
#Running Section 6 of the LVI manual

#Set Working Directorty for each session
setwd("F:\\Rwd")
getwd()
save.image(file="OpenSession.RData")

source('F:\\Rwd\\RScript\\LoadDatasetAndAttachVariableNames.r')
#source('F:\\Rwd\\RScript\\SelectObservationSubset.r')
source('F:\\Rwd\\RScript\\DeclareClassificationVariableAsFactor.r')

#source('F:\\Rwd\\RScript\\ComputeUniformPriorClassProbilityDistribution.r')
source('F:\\Rwd\\RScript\\ComputeSamplePriorClassProbabilityDistribution.r')

source('F:\\Rwd\\RScript\\WritePriorDistributionToFile.r')
source('F:\\Rwd\\RScript\\LoadMASS-R-Package.r')

source('F:\\Rwd\\RScript\\SelectXVariableSubset_v2.1.r')
source('F:\\Rwd\\RScript\\RunMultipleLinearDiscriminantAnalysis_MASS_lda_TakeOneLeaveOne.r')
source('F:\\Rwd\\RScript\\WriteMultipleLinearDiscriminantAnalysis_MASS_lda_TOLO_to_File.r')

source('F:\\Rwd\\RScript\\RunMultipleLinearDiscriminantAnalysis_MASS_lda.r')
source('F:\\Rwd\\RScript\\WriteMultipleLinearDiscriminantAnalysis_MASS_lda.r')

system('python F:\\Rwd\\Python\\COHENS_KHAT.py')
```

```
#Running Section 7 of the LVI manual
#Set Working Directorty for each session
setwd("F:\\Rwd")
getwd()
save.image(file="OpenSession.RData")

source('F:\\Rwd\\RScript\\LoadDatasetAndAttachVariableNames.r')
#source('F:\\Rwd\\RScript\\SelectObservationSubset.r')

#UNIQUEVAR = read.csv('F:\\Rwd\\Rdata\\Archived\\LVI\\Demo\\OutputFiles\\UNIQUEVAR.csv')

source('F:\\Rwd\\RScript\\SelectUniqueXVariableSubset.r')  #modified to load data

source('F:\\Rwd\\RScript\\CompileUniqueXVariableCorrelationMatrixSubset.r')
source('F:\\Rwd\\RScript\\CreateUniqueVarCorrelationMatrixFileForPrinting.r')
source('F:\\Rwd\\RScript\\SelectXVariableSubset_v2.1.r')        #modified to load data
source('F:\\Rwd\\RScript\\AddVariableSubsetIndicatorsToCorrelationMatrix.r')
source('F:\\Rwd\\RScript\\WriteUniqueVarCorrelationMatrix.r')
```

#Running Section 8 of the LVI manual – X variable vs Classification Box Plots & X Variable Scatter Plots

```
#
#Set Working Directorty for each session
setwd("F:\\Rwd")
getwd()
save.image(file="OpenSession.RData")

source('F:\\Rwd\\RScript\\LoadDatasetAndAttachVariableNames.r')
#source('F:\\Rwd\\RScript\\SelectObservationSubset.r')
source('F:\\Rwd\\RScript\\DeclareClassificationVariableAsFactor.r')


source('F:\\Rwd\\RScript\\SelectUniqueXVariableSubset.r')  #modified to load data
source('F:\\Rwd\\RScript\\CreateUniqueVariableClassificationBoxPlots.r')
source('F:\\Rwd\\RScript\\CreateUniqueVariableScatterPlots.r')
```

#Running Part of Section 8 of the LVI manual - Test Y versus Unique X Variable Scatter Plots

```
setwd("F:\\Rwd")
getwd()
save.image(file="OpenSession.RData")

source('F:\\Rwd\\RScript\\LoadDatasetAndAttachVariableNames.r')
source('F:\\Rwd\\RScript\\SelectUniqueXVariableSubset.r')  #modified to load data
source('F:\\Rwd\\RScript\\SelectYVariableSubset_v1.r')  #DATA in XVARSELV1.csv
source('F:\\Rwd\\RScript\\CreateYvXVariableScatterPlots.r')
```

```
#Running Section 9 of the LVI manual
#Set Working Directorty for each session
#setwd("F:\\Rwd")
#getwd()
#save.image(file="OpenSession.RData")

system('python F:\\Rwd\\Python\\COMBINE_EVALUATION_DATASETS.py')

data <- read.csv('F:\\Rwd\\Rdata\\Archived\\LVI\\Demo\\OutputFiles\\ASSESS.csv')

dim(data)
names(data)
data$KHAT
```

```
#Running Section 10 of the LVI manual
#Set Working Directorty for each session
setwd("F:\\Rwd")
getwd()
save.image(file="OpenSession.RData")

source('F:\\Rwd\\RScript\\LoadDatasetAndAttachVariableNames.r')
#source('F:\\Rwd\\RScript\\SelectObservationSubset.r')
source('F:\\Rwd\\RScript\\DeclareClassificationVariableAsFactor.r')

#source('F:\\Rwd\\RScript\\ComputeUniformPriorClassProbilityDistribution.r')
source('F:\\Rwd\\RScript\\ComputeSamplePriorClassProbabilityDistribution.r')

source('F:\\Rwd\\RScript\\SelectXVariableSubset_v1.r')

source('F:\\Rwd\\RScript\\LoadklaR-R-Package.r')
source('F:\\Rwd\\RScript\\LoadCombinat-R-Package.r')
source('F:\\Rwd\\RScript\\RunLinearDiscriminantAnalysis_klaR_pvs.r')    #There should be at least 5 observations per group to perform pairwise variable selection
```

```
#Running Section 11 through 14 of the LVI manual
#Set Working Directorty for each session
setwd("F:\\Rwd")
getwd()
save.image(file="OpenSession.RData")

#Section 11
system('F:\\Rwd\\Phyton\\NN_ZSCORE.py')


#Section 12
system('F:\\Rwd\\Phyton\\NN_XDIST.py')

#Section 13
system('F:\\Rwd\\Phyton\\NN_YDIST.py')

data13 <- read.csv('F:\\Rwd\\Rdata\\Archived\\LVI\\Demo\\OutputFiles\\ASSESS.csv')


#Section 14
system('F:\\Rwd\\Phyton\\NN_TARG_REF.py')

data14 <- read.csv('F:\\Rwd\\Rdata\\Archived\\LVI\\Demo\\OutputFiles\\NNTARGET64.csv')
dim(data14)

#Section 15
system('F:\\Rwd\\Phyton\\COMPILE_NN_TARGET_YSTATS.py')


#Section 16
system('F:\\Rwd\\Phyton\\PIVOT_TABLE.py')
```

```
#Comapre_predictions.r
#load necessary packages
library(yaImpute)
library(randomForest)
library(vegan)

data(TallyLake)
data1 <- data.frame(TallyLake)
xsets <- data1[,9:29]              #all observation

tardat = read.csv('F:\\Rwd\\QTARGET.csv')
refdat=read.csv('F:\\Rwd\\LVINEW.csv')

# set work directory --> this where file with results will be written to
setwd('F:\\Rwd\\')

names(tardat)
names(refdat)

l.targ <- dim(tardat)[1]
l.ref <- dim(refdat)[1]

xr <- refdat[,
c("ctim","elevm","eevsqrd","slopem","slpcosaspm","slpsinaspm","tmb1m","tmb2m","tmb3m","tmb4m","
tmb5m","tmb6m","durm","insom","utmx","utmy","msavim","ndvim","crvm","tancrvm","tancrvsd")]

xt <- tardat[,
c("ctim","elevm","eevsqrd","slopem","slpcosaspm","slpsinaspm","tmb1m","tmb2m","tmb3m","tmb4m","
tmb5m","tmb6m","durm","insom","utmx","utmy","msavim","ndvim","crvm","tancrvm","tancrvsd")]

# define x variables
x <- rbind(xr,xt)

l.comb1 <- dim(x)[1]
# define y variables
y <- refdat[, c("TopHt","LnVolL","LnVolDF","LnVolLP","LnVolES","LnVolAF","LnVolPP","CCover")]

##Compare LVI and yaImpute outputs
euc <- yai(x = x, y = y, method = "euclidean", ann=FALSE)     #method can be one of: msn, msn2,
mahalanobis, ica, euclidean, gnn, randomForest, raw, random
# need to adjust number based on number of Y variables
euc.imp <- impute(euc)[,1:8]
euc.tmp <-
rbind(cbind(rownames(euc$neiIdsRefs),euc$neiIdsRefs,euc$neiDstRefs),cbind(rownames(euc$neiIdsTrgs),
euc$neiIdsTrgs,euc$neiDstTrgs))

#Compare LVI and yaImpute outputs based on thhe five closest nearest neighbors
euc5 <- yai(x = x, y = y, method = "euclidean", k=5,ann=FALSE)     #method can be one of: msn, msn2,
mahalanobis, ica, euclidean, gnn, randomForest, raw, random
euc5.imp <- impute(euc5)[,1:8]
```

```
euc5.tmp <-
rbind(cbind(rownames(euc5$neiIdsRefs),euc5$neiIdsRefs,euc5$neiDstRefs),cbind(rownames(euc5$neiIds
Trgs),euc5$neiIdsTrgs,euc5$neiDstTrgs))


#####################################################################################
############################
# MSN
msn <- yai(x = x, y = y, method = "msn", ann=FALSE)
# need to adjust number based on number of Y variables
# if you have 8 Y variables it should be [,1:8]
msn.imp <- impute(msn)[,1:8]
msn.tmp <-
rbind(cbind(rownames(msn$neiIdsRefs),msn$neiIdsRefs,msn$neiDstRefs),cbind(rownames(msn$neiIdsTr
gs),msn$neiIdsTrgs,msn$neiDstTrgs))

# write out observed and imputed values and distances to file outImp1.csv
#kk.msn <- cbind(rep,msn.tmp,msn.imp,obs)
#write(t(kk.msn), file = "F:\\Rwd\\msn20.csv", ncol=dim(kk.msn)[2], append=T, sep = ",")

#####################################################################################
############################
# k-MSN with k=5
msn5 <- yai(x = x, y = y, method = "msn",k=5, ann=FALSE)

msn5.imp <- impute(msn5,observed=FALSE,method="mean")
msn5.tmp <-
rbind(cbind(rownames(msn5$neiIdsRefs),msn5$neiIdsRefs,msn5$neiDstRefs),cbind(rownames(msn5$neiI
dsTrgs),msn5$neiIdsTrgs,msn5$neiDstTrgs))


#####################################################################################
############################
# RandomForest
rf <- yai(x = x, y = y, method = "randomForest",rfMode="dont", ann=FALSE)

# need to adjust number based on number of Y variables
# if you have 8 Y variables it should be [,1:8]
rf.imp <- impute(rf)[,1:8]
rf.tmp <- rbind(cbind
#Test_LVI.r – test output produced by the LVI software
#load necessary packages
library(yaImpute)
library(randomForest)
library(vegan)

data(TallyLake)
data1 <- data.frame(TallyLake)
# xsets <- data1[,9:29]               #all observation

#dataCor <- cor(xsets)
#dataCov <- cov(xsets)

tardat = read.csv('F:\\Rwd\\QTARGET.csv')
refdat=read.csv('F:\\Rwd\\LVINEW.csv')

xsets <- refdat[,10:30]
dataCor <- cor(xsets)
#1. Check correlation coefficients against UCORCOEF.csv
write.csv(dataCor, file='F:\\Rwd\\Tests\\Cor_Tally.csv')

dataCov <- cov(xsets)
#2. Check covariances for each pair of variables across all the X-variables selected against XCOV.csv
write.csv(dataCov, file='F:\\Rwd\\Tests\\Cov_Tally.csv')

setwd("F:\\Rwd")
getwd()

#source('F:\\Rwd\\RScript\\installPackages.r')
source('F:\\Rwd\\RScript\\LoadDatasetAndAttachVariableNames.r')
#lvinew
nLviRows

source('F:\\Rwd\\RScript\\DeclareClassificationVariableAsFactor.r')
nClasses
uniqueClasses

source('F:\\Rwd\\RScript\\SelectXVariableSubset_v1.r')
XVARSEL
xDataset
xNames

###Check correlation coefficients here
cor(xDataset)

source('F:\\Rwd\\RScript\\Loadsubselect-R-Package.r')

source('F:\\Rwd\\RScript\\RunLinearDiscriminantAnalysis_subselect_ldaHmat.r')
lviHmat

source('F:\\Rwd\\RScript\\Run-ldaHmat-VariableSelection-Improve.r')
lviVariableSets
```

```
source('F:\\Rwd\\RScript\\ExtractVariableNameSubsets.r')
SOLSUM

source('F:\\Rwd\\RScript\\WriteDataframeToCsvFile.r')
#File VARSELCT.csv is written out at 5:04 - it overwrites the previous version

system('python F:\\Rwd\\Python\\EXTRACT_RVARIABLE_COMBOS.py')
```