A Stratified Methodology for Classifier and Recognizer Evaluation

Ross J. Micheals Terrance E. Boult

Vision and Software Technology (VAST) Lab
Computer Science & Engineering Department
Lehigh University Bethlehem, PA

Abstract

In this companion paper, we formally introduce STRAT, *a stratification centric methodology for the empirical evaluation of classification systems. The motivating criteria for* STRAT*'s development are discussed, as well as the potential consequences of departing from some common statistical assumptions made when applying more traditional methods.* STRAT *uses an established replicate statistical technique called balanced repeated replication, or BRR, that does not require the i.i.d. assumption needed for bootstrapping, jackknifing, or binomial techniques.*

1. Introduction

STRAT is a stratified, efficient, flexible, sampling design aware methodology for the empirical evaluation of classification and recognition systems. STRAT, which is short for both *stratification* and "**S**tratified **T**echniques for **R**ecognizer **A**ssessment and **T**esting," has been used by the authors for a variety of classifier evaluations. STRAT's preliminary concepts were introduced in [34]. In [4], it was used to determine if super-resolution techniques could improve face recognition rate. An abbreviated (and unnamed) form of the methodology was first published in [35] where it was shown how STRAT could be used to augment the well-known FERET face recognition algorithm evaluation [37].

This paper has two main objectives. First, for the first time, we formally introduce a complete version of STRAT with enough detail for implementation. Second, this paper is to serve as a position paper and to stimulate discussions of issues that are faced during an evaluation of classification systems. As will be shown, many of the choices made in classifier evaluation are subtle, surprisingly complex, and in some cases, have dramatic consequences.

This paper is organized as follows. In the next section, Section 2, we introduce STRAT's sampling and model viewpoints, as well as some nomenclature that we use throughout the paper. In Section 3, we discuss the fundamental criterion that helped shape the STRAT methodology. Section 4 discusses the ties between classification and clustered data. Section 5 is a survey of sampling issues and more traditional evaluations, and the consequences of departing from some common statistical assumptions. Section 6 is a short review of stratified sampling. This leads to Section 7, where we summarize the stratified replicate statistical technique known as balanced repeated replication, or BRR. After the conclusion in Section 8, there is a brief review of the background required to implement BRR — orthogonal arrays in Appendix A and Galois Fields in Appendix B. The bibliography closes the paper.

2. Sampling, Models, and Terminology

In this section, we briefly define a more formal terminology and system model for use in this paper. For the sake of clarity, and to separate STRAT from the connotations carried from more traditional methods, we will sometimes introduce some terminology that may already have well-known definitions experimental design or other fields. Also, since most of our previous research has been on face recognition evaluation, our examples will be geared towards this area. However, STRAT is flexible enough for a much wider variety of systems. We begin with an overview of classifier evaluation and systems.

2.1. Classifier Evaluation

Classifier evaluation may be viewed as a variety of samplings, each performed at their own layer of abstraction, with perhaps differing methods. The *sampling design*, a term from the statistical literature [6], [31], indicates the method by which the observations are made. We use the term *sampling level* to refer to the "stage" in which the sampling occurs. In this paper, we distinguish between four different designs *simple random*, *stratified*, *cluster* and *systematic* sampling and four sampling levels *population*, *system*, *model*, and *analysis*.

Note that there is the unfortunate potential for the word "population" to connote a group of individuals, as opposed to the more general definition [6] — "the aggregate of interest." Throughout the remainder of the paper, it should be kept in mind that the term population is used to describe the entire collection of potential observations. For example, in face recognition, our use of the term population refers to images of people, not the people themselves.

At each sampling stage there is a method, the *sampling design*, that dictates how our observations are made. In *simple random sampling* or **srs**, observations are made by selecting units where every unit has an equal chance of being drawn [6]. In *stratified sampling*, abbreviated **st**, the population is divided into smaller, non-overlapping subpopulations, or *strata*. If **srs** is performed within each stratum, then the observations were collected via stratified sampling. In *cluster sampling*, each sampling unit itself consists of a group of smaller units, or *subunits*, where the subunits are the elements of interest. When observations are made via regular or deterministic mechanisms (excluding pseudo-random methods) one has performed *systematic sampling*.

2.1.1 Population Level Sampling

Traditionally, sampling refers to the selection of units to observe from a larger *population*. We refer to the stage during which the original data is obtained as the *population level* sampling. For example, the choice of individuals (classes), lighting, pose, sensor parameters, and so on, are all made at this population level. If the data is synthetic, the choice of simulation parameters would be included in the population level sampling.

2.1.2 System Level Sampling

In most systems there exists a set of parameters, some discrete and other continuous, that steer the classifier's behavior. In face recognition systems, this may include thresholds for the minimum allowable proximity to an exemplar, face size normalization parameters, number of Eigenvectors, and so on. In a Principal Component Analysis (PCA) based system, the choice among different norms (i.e., L_1, L_2, or Mahanalobis distance) may also be considered a parameter choice. Obviously, it is impractical to run the classifier over every permutation of system parameters. Therefore, the particular selection of algorithm and sensor parameters used in an evaluation is itself a sampling at the *system level*. Typically, system parameters are selected via systematic sampling.

2.1.3 Model Level Sampling

Because most classifiers require an explicit training phase, we consider the selection of training and/or exemplars as part of the *model level* sampling. Often, system training may be further divided into multiple states as there is both *initial* and *iterative* or *update* training. The initial training refers to the system training that would be performed once, and "global" to the experiments performed. The *iterative* or *update* training refers to training that is specific to a single experiment or trial.

To illustrate, consider one variation of a PCA based face recognition system. Starting with a collection of face imagery, one could collect a set of "generic" bases. Then for a particular set of faces to identify, each exemplar would be projected onto the generic bases, creating the gallery, or collection of people to be identified. Each input image of an unknown subject would be compared to the gallery via projection onto the same set of bases. In this example, the generation of the bases would be considered the initial training, and the update training would consist of the projection of each gallery image. This has the advantage of not requiring explicit retraining when new exemplars are added. However, since information inherent to the classes to be discriminated are not absorbed into PCA basis, there is the potential for the classifier to lose classification accuracy. If the PCA has no initial training phase, i.e. consisting of update training only, it does not suffer from this potential information loss, but must retrain for every new gallery. Having a global training may be particularly disadvantageous to Linear Discriminant Analysis, or LDA, methods, since they may be more sensitive to differences in training. However, more empirical evidence is required to support this claim. Although it is not explicitly stated in their research, the FERET tests ([37] and [3]) and the evaluations of Beveridge et. al. ([2] and [1]) use both initial and iterative training during model level sampling.

2.1.4 Analysis Level Sampling

Finally, at the *analysis level* of sampling there is the review of the data and the synthesis of conclusions. Regardless of the metric, given a large number of probe statistics θ_p we ultimately desire:

- an unbiased estimator of the expected value of θ over all the classes represented in the images
- an unbiased estimator of the standard error, $v(\theta)$ or variance of θ, and
- the ability to state, with probabilistic confidence, the range of values of θ for use in hypothesis testing.

Consider the potential volume of data required for such an estimate. Suppose our goal is to estimate the expected value of some linear statistic defined over the population. Given a probe set of the population, we could obtain a single statistic, but this can provide neither standard error nor a confidence interval. Suppose, however, that instead of a single probe set, it is possible to obtain a set P of multiple probes, denoted as **P**. More formally, $\mathbf{P} = \{P_1, P_2, ..., P_{|P|}\}$ where $\ell(p_{i_j}) = \ell(p_{k_j})$, p_{i_j} is the jth image of probe set i, and i, j, k assume their obvious and reasonable values. Then, it would be possible to collect multiple estimates, one for each probe set. Given enough probe sets, the distribution of the statistic could be estimated or, for some statistics, the central limit theorem could be invoked. Unfortunately, the data requirements to get such a measurement are non-trivial — thirty to fifty images (at minimum) per subject may be required. Let us not lose sight of the fact that obtaining just a *single* estimate may itself be a difficult and time-consuming process, particularly if complex ground-truth is required. We also need a methodology that allows standard error and confidence intervals estimation from a minimal amount of data. In Section 5.2., we discuss many of the traditional techniques applied at this analysis level.

As just illustrated, one of the fundamental difficulties faced in classifier evaluation is the difficulty of acquiring sufficient data — system evaluation can be an enormously time consuming, tedious, and difficult process.

2.2. System Terminology

Let us briefly return to defining terminology. A *scene* is defined as some bounded space and time containing some objects or phenomenon that an experimenter wishes to investigate. Within this scene is some set of characteristics, or *properties* of interest. These properties take on some ideal states — their "true" values can only be estimated through *measurement* or *imaging* by a *sensor* and/or some algorithmic processing. Let a *context* represent a set of potential scenes coupled with a set of constraints that an experimenter attempts to enforce on that scene. A single *experiment* constrains the context to a *subcontext*, a specific set of parameters and constraints. An experiment, however, may consist of many *trials*, during which the scene undergoes some change. In the domain of evaluation, a context can be viewed as putting bounds on the genericity that will be used in the performance evaluation.

Within a context, there are both explicit and implicit properties. We consider any property over which an experimenter exercises direct control, an *explicit* property. All other properties are *implicit*. Typically, most experimenters try to vary the factors of interest, while controlling the variation in the other constraints, which must be accounted for. It is only in this manner that one can draw correspondences between particular variations in input to variations in output. Naturally, there are always a much larger number of implicit properties than explicit ones. One hopes that the vast majority of these implicit properties have negligible effect on the outcome of the experiments. Unfortunately for all experimenters, this is not always the case.

Assume our context defines a set W consisting of L non-overlapping classes of interest, or $W_1, W_2, \ldots, W_L$. Let S be a set of images where each image $s \in S$ belongs to some class W_i. Let $\#_i(S)$ represent the number of images in S belonging to class W_i. Then, we call sets S_1 and S_2 *equally representative* iff $\#_i(S_1) = \#_i(S_2)$ for all $1 \leq i \leq L$ and the union of S_1 and S_2 is empty. Further, we define a set S as *fully representative* with respect to W iff each class corresponds to at least one image in S, i.e., $\#_i(S) > 0$ for all $1 \leq i \leq L$. Finally, we call a set *uniquely representative* iff there is at most one image in S belonging to each class in W, i.e., $\#_i(S) \leq 1$ for all $1 \leq i \leq L$. It follows that a set S with one image per class from class W is both fully and uniquely representative.

A classifier can be considered an algorithm ϕ that, given an input (image) x, returns a class label i indicating that $x \in W_i$. With a human in the loop, a common practice is to relax the definition, outputting a set of *candidates* corresponding to the top n potential labels.

Let the set of images G represent a *training set* or *gallery*. Let P be a set of unknowns, (also referred to as *probes* or *test data*) that need classification. For simplicity, we assume that G and P are fully and uniquely representative. This greatly simplifies our discussion and allows us to substitute G for P as the training set without concerning ourselves with additional normalization issues. (We return to the consequences of swapping the training and test data later.) Most classifiers, given a training image $g \in G$ and probe image $p \in P$, can compute some bounded similarity metric $s_\phi(g,p)$ indicating the proximity, or degree to which p belongs in the class corresponding to g.

As mentioned previously, given an input p, a classifier emits a label i indicating the most likely class W_i to which the input belongs. Typically, this is simply the label of the gallery element g that produced the highest or lowest similarity score $s_\phi(g,p_i)$. In the case where many candidates are emitted, the labels may correspond to the top n similarity scores.

Let g_i represent a gallery image of class W_i, and let $\ell(x)$ represent the label for image x's true class. Given a probe p, a vector of similarity scores $s(g,p)$ can be calculated from all images $g \in G$. Sorting the similarity vector and finding the correct class' respective position along it determines the probe's *rank*. Specifically, a probe has a rank of n over gallery set G if in the similarity vector, there exist exactly n scores greater than or equal to $s(g_{\ell(p)},p)$. For normalization among evaluations with different numbers of subjects, given a rank r and m uniquely representative probes, we define the *relative rank* R as $R = r/m$. Note that evaluations with a greater number of probes enjoy a lower "best" possible relative rank.

With the system terminology behind us, we can now continue on to more interesting issues, such as the principles that have guided STRAT's development.

3. Guiding Principles

There are three fundamental criteria that have steered the development of this new methodology. In this section, we will briefly present the criteria here, and will support them throughout the remainder of the paper.

CRITERION 1 *Make minimal statistical assumptions whenever possible.*

Naturally, as we decrease the number and severity of an evaluation methodology's statistical requirements, the more flexible it becomes. Therefore, whenever possible, the authors advocate the use of methods that make assumptions that are either minimal or are well justified — even if this means sacrificing some statistical efficiency.

CRITERION 2 *Be able to account for clusters.*

Unfortunately, disregarding either natural or experimentally induced clustering of data can have serious consequences. Often, disregarding clusters occurs unexpectedly. For example, if it is assumed that the input data was collected via **srs**, and that classifier outputs are uncorrelated, then one has made the strong implicit statistical assumption that the classifier performs a whitening over natural clusters in the input data. Later, we will show a specific example of the difference in the conclusions made with and without recognizing clustered inputs.

CRITERION 3 *Use the proper estimators given the sampling design.*

A key contribution of the survey sampling community is the theoretical and empirical drive towards the use of estimators that properly reflect complex sampling designs. That is, given a sample knowingly obtained by a certain design, an appropriately adjusted estimator should be applied. Otherwise, as the sample departs from an estimator's underlying assumptions there is an increased chance of obtaining inaccurate results.

For illustration, consider the following overly simplistic example. Given a small population, sampling with replacement from an urn with a binary outcome is best modeled with the binomial distribution. If however, we sampled without replacement, a hypergeometric distribution would be a better choice. Likewise, we would not expect to be able to use the same estimator for both **srs** and **st** and obtain the same results.

4. Classification and Clustered Data

Criterion 2 states that our evaluation methodology should take clustered data into account. This is because classification systems are inextricable from the clustering of data. The very concept of classification assumes that different classes have their own unique distribution, and that samples within a class share some degree of homogeneity — i.e., there are consistent features of an input, that when accurately measured, allow correct labeling. Without the clustering assumptions, there is little merit to the classification problem itself. To illustrate, consider the problem of human identification via facial imagery. Certainly, we expect images of the same person to have some degree of homogeneity; it is this very consistency that allows us to distinguish one face from another. In fact, the authors expect clustered data to be present in any biometric evaluation.

Given a feature vector, a classifier returns a label corresponding to the most likely class to which the input belongs. Given a set of samples from a single class, running each input through a classifier defines an empirical distribution over the set of class labels. This distribution, which summarizes this class' behavior conditioned on the algorithm and training set, is, for the purposes of evaluation, further transformed into some error distribution, indicating some degree of "correctness" for a given class — one of the simplest transforms is simply the fraction of correct labels. Unfortunately, since we cannot expect that each class produces the same distribution of labels, we also cannot expect that each class produces the same error distribution. Therefore, given a set of samples from multiple classes, this final, conglomerate error distribution may be composed of many modes or clusters (numbering at most the number of classes). On the subject of clusters, Kish [26] states:

> The individuals in [clusters] tend to resemble each other — there is usually some *homogeneity* of characteristics, of attitudes, of behavior — but homogeneity is generally not complete[.] Because of this homogeneity, the use of these clusters for sampling units has definite consequences: it destroys the independence of the characteristics of the sample elements. The correspondence with the "well-mixed urn," inherent in the assumption of independence, is negated; and formulas that depend on that assumption fail to apply.

Let us examine this clustering more formally. Given a probe p_i, and an element of class W_i, we obtain, at different system layers, either a set of similarity measures, or a collection of potential labels. Let $\mathbf{p}_i$ represent a random variable that realizes probes belonging to class W_i, i.e., various values of p_i. Then, $\phi(\mathbf{p}_i)$ may be treated as a random variable that describes a distribution of similarity measures or labels potentially generated by running different values of p_i through the classifier. Similarly, if θ represents the metric which transforms the similarity matrix or labels to an error measure, $\theta(\phi(\mathbf{p}_i))$ may also be treated as a random variable. By nature of the classification problem, we cannot expect $\mathbf{p}_i$ and $\mathbf{p}_j$ to have the same statistical properties (for $i \neq j$, obviously). It follows that we cannot expect that $\theta(\phi(\mathbf{p}_i))$ and $\theta(\phi(\mathbf{p}_j))$, the error distributions of classes W_i and W_j, have the same properties either (for $i \neq j$). To do so would imply that $\theta \circ \phi$ also acts as a whitening filter which homogenizes the system output.

Therefore, given a set of samples from multiple classes, we expect this final, conglomerate error distribution to be composed of many modes or clusters.

In a classifier evaluation, we are usually concerned with the output of our system. However, given that we know there are clusters in the original input data, one cannot assume that the system output is not clustered. In order to use a technique that does not account for clustering with full confidence, one *needs to show* that the system properly decorrelates the data.

There is a possibility that the data is also clusterable by some other criterion or covariate — determining the "correct" or "best" stratification criterion is not always clear. This is implicitly reflected in Kish's previous quote when he states the homogeneity is "generally not complete." However, given that we generally expect the most similarity to occur between samples belonging to the same class, it seems only natural to stratify over class identity. In summary, taking a stratified sampling approach allows us to explicitly account for statistically correlated clusters or strata.

5. Sampling Issues

In this section, we discuss in detail, the kinds of assumptions that are often made at the different levels of sampling, and how they may effect an evaluation's results. For now, we will concentrate on *misclassification rate* — the metric most likely to be of greatest interest to classifier researchers.

5.1. Simple Random Sampling

The vast majority of traditional textbook estimation techniques are founded on the assumption that simple random sampling was used to obtain the underlying observations. Unfortunately, as discussed in the previous section it is not always clear that, given a particular experiment, the concept of **srs** over the system output is either feasible or even well defined. In experiments where the data has been collected from an outside source, there is an even greater potential for the sampling design to depart from the evaluator's assumptions.

Merging a binary outcome with the **srs** assumption leads directly to statistical tests and methods based on the binomial distribution (such as McNemar's test) [33]. At first examination, the binomial distribution is also particularly attractive because it is a simple, easily-understood, and well-studied distribution. Its use for evaluation has been discussed for classifiers [1], [10], [15], [41], [42] and for evaluation and tuning in a variety of other domains, including inductive learning systems [18], and speech recognition algorithms [19].

However, there can be direct and measurable consequences when the underlying data departs from **srs**. In [26], Kish discusses some quantitative consequences of violating the **srs** assumption, which we will briefly summarize here. He shows that even small departures from **srs** can have drastic effects on standard errors and confidence intervals. Typically, this distortion increases with the product $\rho(\overline{n}-1)$ where $\overline{n}$ is the average number of elements in a cluster and ρ is the *intraclass correlation coefficient* or **icc**. For example, Kish shows that even if this product is "just" 0.2, a 0.95 confidence interval which is reportedly incorrect with a frequency of 0.05 will, in actuality, be incorrect with a frequency of 0.074, nearly a 50% increase. Therefore, both ρ and $\overline{n}$ need to be small to maintain reasonable confidence intervals. However, Kish also notes that a low **icc** implies a characteristic of measure that is "fairly randomly" distributed within the clusters. Certainly, when it is the class that is both measured *and* the main cause for clustering, we cannot expect a negligible **icc**.

These departures follow directly from the statistics of a sequence of non-stationary Bernoulli trials, that is, where the probability of success p is not fixed. As shown in [5, pp. 135–136], if p varies *within* a trial (i.e., a trial consists of several sub-trials, each with their own p) standard error is reduced. Conversely, if p varies *between* trials (i.e., each *trial* has its own p) standard error in increased. In practice, it may not always be clear whether p is changing within trials, among trials, or both. However, as suggested by [26], if priors are available, a McNemar-style test may be used by making proper adjustments to the test's underlying variance estimators. Otherwise, there is no sound way to "fix" Bernoulli assumptions.

To illustrate the effects of clustered data, consider an example from Cochran [6] "illustrating just how erroneous the binomial formula may be."

> A group of 61 leprosy patients were treated with a drug for 48 weeks. To measure the effect of the drug on the leprosy bacilli, the presence of bacilli at six sites on the body of each patient was tested bacteriologically. Among the 366 sites, 153, or 41.8% were negative. What is the standard error of this percentage?

Using the traditional binomial formula the standard error is approximately 2.58%. However, if one uses the recommended cluster sampling proportion estimators, the standard error is estimated to be about 1.8 times larger, or 4.65% — quite a large difference.

> The binomial formula requires the assumption that results at different sites on the same patient are independent, although actually they have a strong positive correlation.

In some statistical literature, the term *overdispersion* is used to refer to situations where the true variance of a population is larger than that obtained via standard binomial estimators [9]. According to Collett [8, p. 190] the two major causes of overdispersion are (emphasis his) "**variation between the response probabilities** or **correlation between the binary responses**." In other words, when either 'i' of the i.i.d. assumption is violated.

With respect to system evaluation, there has been some limited recognition of the problems with the **srs** assumption as it relates to contingency tables, McNemar's test, and its related χ^2 statistics. Although they do not explicitly recognize the potential to drift from **srs**, Feelders and Verkooijen [16][1], state "whether or not conventional significant levels are appropriate in this kind of hypothesis test is debatable." Salzberg [41] considers the binomial test "a relatively weak test that does not handle quantitative differences between algorithms, nor does it handle more than two algorithms." Reich and Barai [39] note that none of their surveyed techniques "include improvements such as stratified methods." Regarding the similar two-class confusion matrices, James [22] states that "in practice the value of χ^2 [is] of little value."

The recognition of the effect of clustered data on confidence intervals is not widespread outside of the survey sampling literature. Kohavi [29] implicitly recognizes clusters by noting the superiority of *stratified cross-validation*, in which approximate proportions of the class labels are preserved. In his study, stratified cross-validation outperformed traditional cross-validation in terms of bias and variance. Kohavi notes the 0.632 bootstrap estimator [12] had low variance, but an "extremely large bias on some problems." In a more recent paper, Johnson and Keeves explicitly acknowledge Kish, but in the field of educational research [24]. The simulation community has also recognized the potential benefits of recognizing stratification, although in [28] it is used primarily as a variance reduction technique.

5.2. Traditional Evaluations

Although the drive towards sound evaluation methods is a fairly recent trend in the computer vision community, other disciplines have given experimental design and analysis considerably more attention. For example, Toussaint's 1974 survey, [45], includes over 185 citations concerning misclassification rate estimation. Despite this, and the volume of contributions made since then, as recently as 1996, Feelders and Verkooijen observe that "there is no consensus in the research community on how a [performance comparison] study [should be] performed in a methodologically sound way." [16]

As noted in the recent survey by Reich and Barai [39], the machine learning community has primarily used four major methods for evaluating classifier performance — *resubstitution*, *hold-out*, *cross-validation*, and *bootstrap*.

The most fundamental problem with holdout and cross-validation is that if data is shared between trials, then the trial results become interdependent ([10], [14], [29], [41]). Therefore, the underlying assumptions of a variety of statistical tests will become violated. Kohavi [29] gives a specific example where holdout methods yield significant inaccuracies.

The papers by Beveridge et. al., [1], [2] advocate a hybrid permutation and holdout approach. Their technique is partly motivated by their identification of resampling problems that are unique to evaluations involving a large number of classes, and a small number of exemplars or training data per class. Although their discussion is specific to bootstrap, it applies to cross-validation and holdout as well. Instead of simple random subsampling, Beveridge et. al. use a constrained permutation technique to generate the test and training sets for holdout trials. In his thesis, Jensen [23] uses similar randomization methods for the automated evaluation and tuning of classification trees. Beveridge's *balanced sampling*, Jensen's *conditional randomization* and Noreen's *stratified shuffling* [36] are all variations on the theme of preserving problem constraints.

Randomization tests, however, are fundamentally different than traditional parametric tests, and even bootstrap based tests. Noreen [36] observes

> When randomization is used, the null hypothesis is that the dependent variable is unrelated to the explanatory variable(s); or, more precisely, all permutations of the dependent variable relative to the explanatory variables were equally likely. When a conventional parametric method is used, the null hypothesis is that the data are a random sample from a population with certain specified characteristics.

This follows from the observation made by Cohen [7, pp. 175–176], that randomization does not produce sampling distribution in the traditional sense. In fact, Cohen and Noreen both state that randomization techniques should not be used when drawing an inference about a population parameter. Similarly, Efron and Tibshirani [13] lament "the standard deviation of the permutation distribution is *not* a dependable estimate of standard error for $\hat{\theta}$ (it is not indented to be), while the bootstrap standard deviation is."

In summary, we have seen that standard tests degrade, sometimes drastically, when departing from either postulate of the i.i.d. assumption. Many textbook methods, such as binomial tests, McNemar's test, and even the well-known resampling methods jackknife and bootstrap, however, *require* i.i.d. data. If there is correlation among subgroups, or, if the performance of the classifier varies between subgroups, then a method that specifically addresses the resulting clustering of the data is needed.

6. Stratified Sampling

We briefly review stratified sampling and how it may be used to estimate an expected value, or mean of a population statistic, denoted $\bar{y}$.[2] Suppose that given a population P of size N, P is divided into L mutually exclusive subpopulations, or *strata* of sizes $N_1, N_2, \dots, N_L$.

[1] [15] is the preliminary version of [16].

[2] The notation and definitions in this section are from [6].

After this division, or *stratification*, suppose that for stratum h of size N_h, we draw n_h samples. If n represents the total number of samples, then $n = n_1 + n_2 + \cdots + n_L$. Specifically, let $y_{(h,i)}$ represent the ith value drawn from stratum h. If the *stratum weight* (of stratum h) is defined as $W_h = N_h/N$ where $\sum_{h=1}^{L} W_h = 1$, then the stratified sampling estimate of the sample mean $\overline{y}_{\text{st}}$ is

$$\overline{y}_{\text{st}} = \sum_{h=1}^{L} W_h \overline{y}_h, \tag{1}$$

where $\overline{y}_h$ is the sample mean of stratum h, or

$$\overline{y}_h = (1/n_h) \sum_{i=1}^{n_h} y_{(h,i)}. \tag{2}$$

Note, it can be shown that $\overline{y}_{\text{st}}$ is an unbiased estimator of the population mean [47]. If, for all h, $n_h/n = N_h/N$ or $n_h/N_h = n/N$, then $\overline{y}_{\text{st}}$ simplifies to the traditional sample mean

$$\overline{y} = (1/n) \sum_{h=1}^{L} n_h \overline{y}_h. \tag{3}$$

6.1. Conditions and Advantages

In order for stratified sampling to be successful, two fundamental requirements must be met. First, each stratum must be *independent, but not necessarily identically distributed*. Stratifying our population by class satisfies this constraint because, by definition of the classification problem, we expect that each class is its own, independent, and unique distribution. Second, each stratum should be relatively homogeneous. That is, the variance of samples drawn from within the same stratum should be significantly less than the variance of samples drawn from multiple strata. This second requirement may be more difficult to fulfill, since it depends highly upon the nature of the underlying classes. Classes difficult to identify will yield an error distribution with a wide variance. Regardless, our previous experimentation has reflected Cochran's observation [6] that proper application of stratified methods rarely decreases an estimator's accuracy.

Aside from handling clustered data, there are many auxiliary benefits to stratified sampling ([6] and [31]). First and most importantly, stratification can help ensure consistency across experiments. This is particularly important in sensor evaluation — if our subcontexts are sufficiently well-defined and maintained across experiments then we help control the ability of scene changes to confound our results.

Second, stratification may help ensure that small subpopulations are explicitly included. This is particularly difficult to ensure given simple random sampling — a large number of samples, and therefore a large amount of corresponding ground truth, is required to ensure that unlikely classes are captured. If the collected data is not particularly representative of a population, then the stratum weights may be adjusted accordingly, making subsequent evaluations more meaningful.

Finally, stratification allows for *weighting*. Historically, these weights have been primarily used to take into account the relative difference in population size. However, they have also been used to account for cost, or “difficulty” in obtaining a sample, and in some cases may be used to determine the boundaries of the strata themselves [6].

One significant disadvantage of traditional stratified sampling is the lack of definitive methods for obtaining confidence intervals. In [6], a method is provided that can estimate the stratified sampling mean estimator's effective degrees of freedom, however, it requires that each of the $y_{(h,i)}$ are normal. What we require, therefore, is an alternate method of obtaining the stratified sampling estimates.

7. Balanced Repeated Replication

In this section, we briefly discuss a specific type of *balanced repeated replication*, or BRR.[3] As will be shown, BRR will eventually allow us to draw confidence intervals over our estimated statistics, reduce the amount of required data, and eliminate the need for more than one training.

It should be noted that BRR is not the only replicate statistics technique designed for stratified samples. The two major variations on BRR are *Fay's method* [25] and Sitter's *balanced orthogonal multi-array based BRR* [44]. In addition, there are the bootstrap and jackknife variants *bootstrap repeated replication* and *jackknife repeated replication* [17]. Out of all these techniques, BRR was selected because it is the most well-studied; whether another method is preferable remains an open question.

7.1. Full Half-Sampling

We use the same notation from Section 6. Assume we are given L strata and $n_h = 2$ units drawn from each stratum. Often n_h is referred to as the *primary sampling unit* or PSU. Then, if $y_{(h,i)}$ represents the ith unit from stratum h, then our data may be decomposed into two sets, one consisting of all of the first samples from each stratum, $y_1 = \{y_{(1,1)}, y_{(2,1)}, \ldots, y_{(L,1)}\}$, and the second composed of all of the second samples $y_2 = \{y_{(1,2)}, y_{(2,2)}, \ldots, y_{(L,2)}\}$. Let $\hat{\theta}_{\text{st}}$ represent some estimator of a general linear statistic applied over the population. As shown in [40], [26], BRR may also be applied to a much wider variety of statistics, including ratios, correlation coefficients [17], non-linear functions of population totals [46], and quartiles [43] as well as any non-linear function of subpopulation totals [46]. However, for simplicity, we limit our derivations here to linear statistics,

[3] BRR is not a new technique. The material presented in this section is adapted from [6], [11], [31], [32], and [47]

where we assume that the weights of the statistic have already been absorbed into the various stratum weights, W_h. Let $\hat{\theta}_{(\mathrm{st},y_1)}$ and $\hat{\theta}_{(\mathrm{st},y_2)}$ represent the estimator applied over the subsets y_1 and y_2 respectively. If we were interested in the population mean, and each stratum was given equal weight, then from Section 6, we know that an estimate of the population mean is $\bar{y}_{\mathrm{st}} = (\bar{y}_{(\mathrm{st},1)} + \bar{y}_{(\mathrm{st},2)}/2)$. Unfortunately, this estimate has only one degree of freedom, and as a consequence, lacks stability. Therefore, instead of traditional stratified sampling, suppose we were to generate a synthetic half-sample, or *replicate* by selecting a value from either y_1 or y_2 for each stratum. Having L strata and a PSU of 2 implies that there exist 2^L such half-samples. Given a single half-sample α, we could apply our linear estimator, yielding $\hat{\theta}_{(\mathrm{st},\alpha)}$. Letting: $\delta_{(h,1,\alpha)} = 1$ if $y_{(h,1)} \in$ half-sample α (0 otherwise), and $\delta_{(h,2,\alpha)} = 1 - \delta_{(h,1,\alpha)}$, the half-sample α estimate is

$$\hat{\theta}_{(\mathrm{st},\alpha)} = \sum_{h=1}^{L} W_h(\delta_{(h,1,\alpha)} y_{(h,1)} + \delta_{(h,2,\alpha)} y_{(h,2)}) \qquad (4)$$

where W_h represents an optional stratum weight. It follows that the replicate based sample population mean may be written

$$\hat{\theta}_{(\mathrm{st},2^L)} = \frac{1}{2^L} \sum_{\alpha=1}^{2^L} \hat{\theta}_{(\mathrm{st},\alpha)}. \qquad (5)$$

Through simple algebraic manipulation [47], it may be shown that

$$\hat{\theta}_{(\mathrm{st},2^L)} = \sum_{h=1}^{L} W_h(y_{(h,1)} + y_{(h,2)})(2^{L-1}/2^L) = \hat{\theta}_{\mathrm{st}} \qquad (6)$$

indicating $\bar{y}_{(\mathrm{st},2^L)}$ is an unbiased estimator of the population mean. We now move on to the second moment estimate. Let $d_h = y_{(h,1)} - y_{(h,2)}$ and

$$\delta_h^{(\alpha)} = \begin{cases} 1 & \text{if } y_{(h,1)} \in \text{half-sample } \alpha \\ -1 & \text{if } y_{(h,2)} \in \text{half-sample } \alpha, \end{cases} \qquad (7)$$

or, equivalently, $\delta_h^{(\alpha)} = 2\delta_{(h,1,\alpha)} - 1$. Then,

$$[\hat{\theta}_{(\mathrm{st},\alpha)} - \hat{\theta}_{\mathrm{st}}] = \sum_{h=1}^{L} W_h \delta_h^{(\alpha)} d_h/2. \qquad (8)$$

The 2^L replicate based variance estimate $v_{(2^L)}(\hat{\theta}_{\mathrm{st}})$ equals $[\hat{\theta}_{(\mathrm{st},\alpha)} - \hat{\theta}_{\mathrm{st}}]^2$ or

$$\sum_{h=1}^{L} W_h^2 d_h^2/4 + \sum_{h<h'}^{L} \delta_h^{(\alpha)} \delta_{h'}^{(\alpha)} W_h W_{h'} d_h d_{h'}/2 \qquad (9)$$

where the second summation is over all pairs of (h,h') such that $h < h' \leq L$. Unfortunately, even for moderate values of L, Equation 9 requires a large number of computations. Generating half-samples for an evaluation involving hundreds of stratum, therefore, becomes intractable. One potential speedup is to use some random, k-element subset of the 2^L half-samples. The corresponding variance estimator becomes

$$v_k(\hat{\theta}_{\mathrm{st}}) = \frac{1}{k(n_h - 1)} \sum_{\alpha=1}^{k} (\hat{\theta}_{(\mathrm{st},\alpha)} - \hat{\theta}_{\mathrm{st}})^2 \qquad (10)$$

Unfortunately, using simply random subsets yields a biased estimator. The goal, therefore, is to select a subset such that $v_{(k)}(\bar{y}_{\mathrm{st}}) = v(\bar{y}_{\mathrm{st}})$.

7.2. Balanced Half-Sampling

This brings us (finally) to the concept of *balanced half-sampling* or more generally, *balanced repeated replication* (often abbreviated BRR). In balanced half-sampling, we select a set of k replicates, such that $k < 2^L$ (typically, $k \ll 2^L$), and $v_{(k)}(\bar{y}_{\mathrm{st}})$ is unbiased. This can be accomplished if, for all $h < h' \leq L$ [32],

$$\sum_{\alpha=1}^{k} \delta_h^{(\alpha)} \delta_{h'}^{(\alpha)} = 0. \qquad (11)$$

If this criterion is met, then the half-samples are considered to be *balanced*, since all cross-stratum terms will cancel. Thus, $v_{(k)}(\bar{y}_{\mathrm{st}}) = v(\bar{y}_{\mathrm{st}})$.

Since, for each of the k replicates we must choose a sample from each of the h stratum, a natural representation for this set is a $k \times h$ array, which we denote B. Note that this array does not appear directly in Equation (10); it is used only to build the collection of half samples. If $B_{(\alpha,h)}$ is $+1$, then element $y_{(h,1)}$ should be included in replicate α. Otherwise, -1 indicates element $y_{(h,2)}$ should be in the αth half-sample. For example, if we abbreviate $+1$ and -1 with $+$ and $-$ respectively, then the following 8×7 orthogonal array could be used for eight replicates based on seven stratum.

$$B = \begin{bmatrix} + & + & - & + & - & - & - \\ + & - & + & - & - & - & + \\ - & + & - & - & - & + & + \\ + & - & - & - & + & + & - \\ - & - & - & + & + & - & + \\ - & - & + & + & - & + & - \\ - & + & + & - & + & - & - \\ + & + & + & + & + & + & + \end{bmatrix} \qquad (12)$$

For example, the 3rd half sample, constructed from the 3rd row of B is

$$\alpha_3 = \{y_{(1,2)}, y_{(2,1)}, y_{(3,2)}, y_{(4,2)}, y_{(5,2)}, y_{(6,1)}, y_{(7,2)}\}. \qquad (13)$$

However, when using balanced replicates

$$\hat{\theta}_{\mathrm{st}}^{(k)} = \frac{1}{k} \sum_{\alpha=1}^{k} \hat{\theta}_{(\mathrm{st},\alpha)} \qquad (14)$$

does *not* imply that the mean of the half-samples $\hat{\theta}_{\text{st}}^{(k)}$ equals $\hat{\theta}_{\text{st}}$. In order to have this desirable property, for each $1 < h \leq L$,

$$\sum_{\alpha=1}^{k} B_{(\alpha,h)} = 0 \qquad (15)$$

must be true. This makes sense intuitively, since this equation simply states that each sample must be selected an equal number of times. Half-samples satisfying both Equation (11) and Equation (15) are said to be in *full balance* [47]. Note that this orthogonal array could be used for any number of stratum $L^* < L$ since every collection of L^* rows are mutually orthogonal.

What is the proper value for k? Obviously, we desire a k large enough to provide a reasonable estimate, small enough to be tractable, and sufficient for balancing. Both Wolter [47] and Gurney & Jewett [20] suggest a construction originally from [38]: use 2^{β} replicates where β satisfies the inequality $L \leq (2^{\beta} - 1)$. For example, given $L = 483$, then 512 replicates ($\beta = 9$) suffice (Certainly, $512 \ll 2^{483}$.)

7.3. General BRR

So far, we have concentrated on the limiting case of two samples per stratum. As shown in [20] and [47], the BRR paradigm can be extended to any case where the PSU is a prime integer. Given n samples per stratum each replicate, or n^{-1}-sample contains a single element from each stratum, we redefine $\delta_{(h,1,\alpha)} = 1$ with respect to $1 < i < n$. That is, $\delta_{(h,i,\alpha)} = 1$ if sample $y_{(h,i)} \in$ replicate α and $\delta_{(h,i,\alpha)} = 0$, otherwise. Extending the definition of Equation (4) in the same fashion yields

$$\hat{\theta}_{\text{st}} = \sum_{h=1}^{L} W_h(\delta_{(h,1,\alpha)} y_{(h,1)} + \cdots + \delta_{(h,n,\alpha)} y_{(h,n)}) \qquad (16)$$

According to [47], just as in the half-sample case, we could use all n^L replicates to build the “textbook” equivalent population mean estimate

$$\hat{\theta}_{(\text{st},n^L)} = \sum_{\alpha=1}^{n^L} \hat{\theta}_{(\text{st},\alpha)} / n^L = \hat{\theta}_{\text{st}} \qquad (17)$$

and variance estimate

$$v_{(n^L)}(\hat{\theta}_{\text{st}}) = \frac{1}{n^L(n-1)} \sum_{\alpha=1}^{n^L} (\hat{\theta}_{(\text{st},\alpha)} - \hat{\theta}_{\text{st}})^2 = v(\hat{\theta}_{\text{st}}). \qquad (18)$$

Again, we need an orthogonal array, B, for construction of the k replicates. We use n^{β} replicates, where β satisfies the inequality $L \leq (n^{\beta} - 1e)$. For example, given $L = 256$ strata and $n = 3$ PSU, we will use 729 replicates since $L = 256 < (3^6 - 1)$. In the appendix, it is shown how to construct a large variety of orthogonal arrays that can be used to construct sets of fully balanced replicates.

Statistically, using a larger number of PSU can have a significant advantage (as demonstrated later). Usually, the leading $1/(n_h - 1)$ term in Equation (10) helps yield smaller variance, and therefore, tighter confidence intervals. The degree to which this occurs, of course, varies according to both the nature of the estimator, the underlying distributions and the samples.

For both the PSU $= 2$ and PSU > 2 case, an orthogonal array is required for the selection of a set of balanced half-samples. Although orthogonal arrays can be tedious to generate by hand, there are two viable alternatives. First, there exist a number of Internet resources with free galleries of orthogonal arrays. Even if a desirable orthogonal array is missing from a gallery, a mathematical software package that supports Galois Fields may be used. Maple source code that may be used to generate an arbitrarily large orthogonal array is provided in Appendix B.

7.4. Confidence Intervals

To the best of the authors' knowledge, [17] is the earliest work to discuss, in detail, the use of BRR for the application of confidence intervals. Frankel states that “the distribution of the ratio of the first-order sample estimate minus its expected value, to its estimated standard error is reasonably approximated by Student's-*t* within symmetric intervals.” In other words, the normalized distribution of the statistics of the BRR estimates can be approximated by the Student-*t* distribution when using two-sided confidence intervals. Frankel shows, empirically, that by using L (the number of strata) degrees of freedom, d, makes this assumption quite reasonable. Another important empirical study by Kish and Frankel [27], to paraphrase [30], “found the *t* approximation to be adequate for confidence intervals for a variety of population parameters with as few as 6 or 12 strata.”

In 1981, Krewski and Rao [30] provided an analytic proof that the normalized distribution of the first-order BRR estimates approaches a normal distribution as the number of strata goes to infinity. A later paper by Rust and Rao [40] suggest that the true value of d is “somewhat smaller” than the number of strata. However, in practice, given a large enough number of strata, even a large difference between the true and estimated values of d does not significantly alter the confidence interval. This last property is one of the reasons why the methodology is well suited for evaluation of systems with a large number of classes.

8. Conclusions and Future Work

In this paper, we introduced STRAT, a methodology for classifier evaluation. STRAT is flexible enough to handle a wide variety of statistics, efficient enough for practical use, and based on sound stratification principles. In addition, we discussed not only the criteria that guided the development of STRAT, but also discussed the consequences of departing from these requirements. The life of STRAT as an evaluation methodology is merely beginning. Currently, STRAT is based on BRR, a replicate statistics method that does *not* require the i.i.d. assumption that limits the applicability of other techniques to empirical classifier evaluation.

In the future, we hope to investigate how STRAT may be extended to other computer vision systems. We have begun investigating how STRAT can be used for sensor evaluation. This is a particularly challenging domain since, unlike algorithm evaluation, one can no longer guarantee identical inputs. We also hope to perform a meta-evaluation, to find some of the practical strengths and weakness of STRAT as compared to other existing methodologies.

A. Orthogonal Arrays

This appendix is concerned with the definition and fundamentals of orthogonal arrays. So that orthogonal arrays are not confused with the similarly named concept of orthogonal matrices, we use the term "array" in places where "matrix" may otherwise be preferred. The precise definition of an orthogonal array varies across different literature, although its spirit (usually) remains intact.

We begin with a formal definition of an orthogonal array. Let S represent some finite set of n symbols, $S = \{s_1, s_2, \ldots, s_n\}$. Let A represent some $m \times n$ array composed of symbols of S, i.e.

$$a_{(i,j)} \in S, \forall (1 \leq i \leq m \text{ and } 1 \leq j \leq n). \tag{19}$$

Let $a_{(\cdot,j)}$ represent the jth column of array A. Given two columns of A, $a_{(\cdot,q)}$ and $a_{(\cdot,r)}$ $(1 \leq q, r \leq n,\ q \neq r)$, if each ordered pair (a_{iq}, a_{ir}), for all $(1 \leq i \leq m)$, appears either zero or the same number (usually, but not necessarily, $m/2^{|S|}$), then A is an *orthogonal array*. For example, Equation (12) is an orthogonal array, since for any two columns, the ordered pairs $(+,+)$, $(+,-)$, $(-,+)$, $(-,-)$ each appear the same number of times (twice, in the above array).

The following orthogonal array has special significance

$$H_2 = \begin{bmatrix} - & + \\ + & + \end{bmatrix} \tag{20}$$

It follows that H_2 is indeed orthogonal since the ordered pairs $(-1,1)$ and $(1,1)$ each appear once, and all other ordered pairs, only $(1,-1)$ and $(-1,-1)$ in this case, appear zero times.

The matrix H_2 can be used to generate (some, but not all of the) elements of the set of *Hadamard matrices*, which are a special subset of orthogonal arrays. If an orthogonal array A is square and has the symbol set $S = \{-1, 1\}$, then A is a Hadamard matrix.

Hadamard matrices are quite easy to construct, since they can be defined recursively. That is, given Hadamard matrix H_n, then

$$H_{n^2} = \begin{bmatrix} H_n & H_n \\ -H_n & H_n \end{bmatrix} \tag{21}$$

is also a Hadamard matrix. It is trivial to see that if H is a Hadamard matrix, then $-H$ is as well.

As discussed in Appendix B, construction of general orthogonal arrays is more difficult, but quite tractable, nevertheless. Regardless of how they are derived, however, once they are constructed, orthogonal arrays can be reused across a large number of experiments.

B. Galois Fields

In this appendix, we review the definition of Galois Fields,[4] their properties, and generation within a replicate statistics context. The material presented in this section is adapted from [20]. Although it may be somewhat non-standard, the notation used in this section is designed for clarity.

B.1. Definition

Simply put, a Galois Field is a special field of polynomials. Suppose we are given a set P of at most degree n polynomials with integer coefficients of at most p, where $p > 0$ and p is prime. Polynomial addition and multiplication are defined in the usual manner, except that each of the coefficients c of the result of the operation are reduced to c **mod** p. In this appendix, we will use **coefmod** to denote this operation. For example,

$$\begin{aligned} & \left[(2x^3 + x^2 + x) + (2x^3 + 2x^2 + 1)\right] \ \textbf{coefmod}\ 3 \\ = \ & (4x^3 + 3x^2 + x + 1)\ \textbf{coefmod}\ 3 \\ = \ & x^3 + x + 1 \end{aligned} \tag{22}$$

or

$$\begin{aligned} & \left[(x^2 + 2x + 1)(x + 1)\right] \ \textbf{coefmod}\ 3 \\ = \ & (x^3 + 3x^2 + 3x + 1)\ \textbf{coefmod}\ 3 \\ = \ & x^3 + x \end{aligned} \tag{23}$$

We define a polynomial q as Q-irreducible if, for $q \in Q$, q cannot be expressed as a product of two polynomials $q = rs$ where neither q nor s is 1 and both $r \in Q$ and $s \in Q$. Let g represent a P-irreducible polynomial where P is as defined in the previous paragraph. Then, the set P modulo the polynomial g forms a *Galois Field* of p^n elements. In other words, if **coefmod**ed multiplication operation yields a polynomial of degree p or greater, then the result should be divided by g, and replaced with the remainder of this division. For example, given $n = 5$ and $p = 3$,

$$\begin{aligned} & x^5\ \textbf{mod}\ (x^5 + 2x + 1)\ \textbf{coefmod}\ 3 \\ = \ & (x^5 + 2x + 1) + (x + 2)\ \textbf{mod}\ (x^5 + 2x + 1)\ \textbf{coefmod}\ 3 \\ = \ & (x + 2)\ \textbf{coefmod}\ 3 \\ = \ & x + 2 \end{aligned} \tag{24}$$

We now require a process than can be used to systematically generate the elements of a Galois Field.

[4] According to the online version of Merriam-Webster's Collegiate Dictionary (`http://m-w.com/`) fields, Galois is pronounced "gal-**wä**"

	x^4	x^3	x^2	x^1	1
g^1 **coefmod** 3 **mod** $g = 2x^4 + 2x^3 + 2$	2	2	0	0	2
g^2 **coefmod** 3 **mod** $g = 2x^4 + x^3 + 2$	2	1	0	0	2
g^3 **coefmod** 3 **mod** $g = x^4 + 2x^3 + 2x + 2$	1	2	2	0	2
	$\vdots$				
g^{241} **coefmod** 3 **mod** $g = 2x^4 + 2x^3 + x^2 + x + 1$	2	2	1	1	1
g^{242} **coefmod** 3 **mod** $g = 1$	0	0	0	0	1

Figure 1: Galois Fields can be generated from successive powers of the generating element g.

B.2. Generation

If g is a P-irreducible polynomial, $g^{(p^n-1)} = 1$, and $g \neq 1$, then a Galois Field over P, denoted $GF(p^n)$ can be generated by taking successive powers of g, and performing the appropriate **mod** and **coefmod** operations. The polynomial g is often referred to as a *primitive root* or *generating element*. Consider the generation of GF(3^5) (i.e., $p = 3$ and $n = 5$) given generating element $g = (2x^4 + 2x^3 + 2)$. Note that there may be many generating elements for a particular Galois Field. For instance $g = (x + 1)$ is also a generating element of GF(3^5).

The most difficult part of Galois Field generation is the selection of the generating polynomial g. This can be done with a process such as the one described in [21, pp. 156–158]. Fortunately, the mathematical software package **Maple**™ contains facilities. The code snippet found in Figure 2 can be used to generate full Galois Fields and save it to a file.

B.3. Orthogonal Array Generation

Given a Galois Field in the above form, we may generate an orthogonal array of, at largest, size $p^n \times (p^n - 1)$. Choose any column of the Galois Field array and make this the first column of the orthogonal array, with the exception of the bottommost element which should be set to zero. Successive columns of the matrix can be generated by rotating the elements of the previous column with (again) the exception of the bottommost element. Assuming $m_{(i,j)}$ represents the element at row i, column j of matrix M, then

$$\begin{aligned} m_{(i,j+1)} &= m_{(i+1,j)} && \text{if } i < p^n - 1, \\ m_{(i,j+1)} &= m_{(1,j)} && \text{if } i = p^n - 1, \\ m_{(i,j)} &= 0 && \text{otherwise.} \end{aligned} \tag{25}$$

```
gen_matrix := proc (modulo, degrees)
  local G, prim, tmp, i, j, fname;
  fname := ‘GF-‘ . modulo . ‘-‘ . degrees . ‘.dat‘;
  writeto(fname);
  printf(‘GF(%gˆ%g), ‘, modulo, degrees);
  G := GF(modulo, degrees);
  prim := G[PrimitiveElement](‘?‘);
  printf(‘generator = %s\n‘, \
        convert(subs(‘?‘=a,G[ConvertOut](prim)),string));
    for i from 1 by 1 to G[size] - 1 do
      tmp := G[ConvertOut](G[‘ˆ‘](prim,i));
      printf(‘%4g   ‘, i);
      for j from 0 by 1 to degrees-1 do
        printf(‘%g ‘, coeff(tmp, ‘?‘, j));
      od;
     printf(‘\n‘);
  od;
```

Figure 2: **Maple**™ (Version 5 Release 3) source code for Galois Field generation. Note that successive runs of this function given the same parameters may result in different, but isomorphic, fields.

Based on these rules of construction, the following is one potential orthogonal matrix generated from GF(3^3).

$$\begin{bmatrix}
0&2&1&1&2&1&0&1&0&0&2&2&2&0&1&2&2&1&2&0&2&0&0&1&1&1\\
2&1&1&2&1&0&1&0&0&2&2&2&0&1&2&2&1&2&0&2&0&0&1&1&1&0\\
1&1&2&1&0&1&0&0&2&2&2&0&1&2&2&1&2&0&2&0&0&1&1&1&0&2\\
1&2&1&0&1&0&0&2&2&2&0&1&2&2&1&2&0&2&0&0&1&1&1&0&2&1\\
2&1&0&1&0&0&2&2&2&0&1&2&2&1&2&0&2&0&0&1&1&1&0&2&1&1\\
1&0&1&0&0&2&2&2&0&1&2&2&1&2&0&2&0&0&1&1&1&0&2&1&1&2\\
0&1&0&0&2&2&2&0&1&2&2&1&2&0&2&0&0&1&1&1&0&2&1&1&2&1\\
1&0&0&2&2&2&0&1&2&2&1&2&0&2&0&0&1&1&1&0&2&1&1&2&1&0\\
0&0&2&2&2&0&1&2&2&1&2&0&2&0&0&1&1&1&0&2&1&1&2&1&0&1\\
0&2&2&2&0&1&2&2&1&2&0&2&0&0&1&1&1&0&2&1&1&2&1&0&1&0\\
2&2&2&0&1&2&2&1&2&0&2&0&0&1&1&1&0&2&1&1&2&1&0&1&0&0\\
2&2&0&1&2&2&1&2&0&2&0&0&1&1&1&0&2&1&1&2&1&0&1&0&0&2\\
2&0&1&2&2&1&2&0&2&0&0&1&1&1&0&2&1&1&2&1&0&1&0&0&2&2\\
0&1&2&2&1&2&0&2&0&0&1&1&1&0&2&1&1&2&1&0&1&0&0&2&2&2\\
1&2&2&1&2&0&2&0&0&1&1&1&0&2&1&1&2&1&0&1&0&0&2&2&2&0\\
2&2&1&2&0&2&0&0&1&1&1&0&2&1&1&2&1&0&1&0&0&2&2&2&0&1\\
2&1&2&0&2&0&0&1&1&1&0&2&1&1&2&1&0&1&0&0&2&2&2&0&1&2\\
1&2&0&2&0&0&1&1&1&0&2&1&1&2&1&0&1&0&0&2&2&2&0&1&2&2\\
2&0&2&0&0&1&1&1&0&2&1&1&2&1&0&1&0&0&2&2&2&0&1&2&2&1\\
0&2&0&0&1&1&1&0&2&1&1&2&1&0&1&0&0&2&2&2&0&1&2&2&1&2\\
2&0&0&1&1&1&0&2&1&1&2&1&0&1&0&0&2&2&2&0&1&2&2&1&2&0\\
0&0&1&1&1&0&2&1&1&2&1&0&1&0&0&2&2&2&0&1&2&2&1&2&0&2\\
0&1&1&1&0&2&1&1&2&1&0&1&0&0&2&2&2&0&1&2&2&1&2&0&2&0\\
1&1&1&0&2&1&1&2&1&0&1&0&0&2&2&2&0&1&2&2&1&2&0&2&0&0\\
1&1&0&2&1&1&2&1&0&1&0&0&2&2&2&0&1&2&2&1&2&0&2&0&0&1\\
1&0&2&1&1&2&1&0&1&0&0&2&2&2&0&1&2&2&1&2&0&2&0&0&1&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0
\end{bmatrix} \tag{26}$$

The **Maple**™ code shown in Figure 2 generates the first column of an orthogonal matrix. The output is comma delimited so that it may be included in C header files.

C. Acknowledgments

The authors would like to thank **Herve Abdi** (University of Texas at Dallas), **Ross Beveridge** (Colorado State University), **Meghan Blackford** (Lehigh University), **Michael Eckmann** (Lehigh University), **Patrick Grother** (DARPA ITO), **Jonathan Phillips** (DARPA ITO), **Stefan Leigh** (NIST), **Visvanathan Ramesh** (Seimens Corporate Research), **Andrew Rukhin** (NIST), and **Gao Xiang** (Lehigh University) for their support and input.

References

[1] J. Ross Beveridge, Bruce Draper, Kai She, and Geof H. Givens. Parametric and nonparametric methods for the statistical evaluation of human id algorithms. Technical report, Colorado State University, 2001. Available through `http://www.cs.colostate.edu/evalfacerec/`.

[2] J. Ross Beveridge, Kai She, Bruce Draper, and Geof H. Givens. A nonparametric statistical comparison of principal component and linear discriminant subspaces for face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2001)*, Hawaii, December 11–13 2001.

[3] D. M. Blackburn, M. Bone, and P. J. Phillips. Facial recognition vendor test 2000. http://www.dodcounterdrug.com/facialrecognition/, Dec 2000.

[4] Terrance E. Boult, Ming-Chao Chiang, and Ross J. Micheals. *Super-Resoluation Imaging*, chapter Super-Resolution via Image Warping. Kluwer Academic Publishers, 2001.

[5] George E. P. Box, William G. Hunter, and J. Stuart Hunter. *Statistics for Experimenters: An Introduction to Design, Data Analysis, and Model Building.* John Wiley & Sons, 1978.

[6] William G. Cochran. *Sampling Techniques.* John Wiley & Sons, third edition, 1977.

[7] Paul R. Cohen. *Empirical Methods for Artificial Intelligence.* MIT Press, 1995.

[8] D. Collett. *Modelling Binary Data.* Chapman & Hall, 1991.

[9] D.R. Cox and E. J. Snell. *Analysis of Binary Data.* Chapman and Hall, second edition, 1989.

[10] Thomas G. Dietterich. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Compuation*, 10(7):1895–1924, 1998.

[11] Bradley Efron. *The Jackknife, the Bootstrap, and Other Resampling Plans*, volume 38 of CBMS-NSF Regional Conference Series in Applied Mathematics. SIAM, Philadelphia, Pennsylvania, 1982.

[12] Bradley Efron. Estimating the error rate of a prediction rule: Improvement on cross-validation. *Journal of the American Statistical Association*, 78(382):316–331, June 1983.

[13] Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap.* Chapman & Hall, 1993.

[14] Floriana Esposito, Donato Malerba, and Giovanni Semeraro. A comparitiv analysis of methods for pruning decision trees. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(5), May 1997.

[15] A. Feelders and W. Verkooijen. Which method learns most from data? In *Preliminary papers on the Fifth International Workshop on Artificial Intelligence and Statistics*, pages 219–225, Fort Lauderdale, Florida, January 4–7 1995.

[16] A. Feelders and W. Verkooijen. *Learning from Data: AI and Statistics V*, chapter On the Statistical Comparison of Inductive Learning Methods, pages 271–279. Springer-Verlag, 1996.

[17] Martin R. Frankel. *Inference from Survey Samples: An Empirical Investigation.* Litho Crafters, Inc., 1971.

[18] Olivier Gascuel and Gilles Caraux. Statistical significance in inductive learning. In *10th European Conference on Artificial Intelligence (ECAI 92)*, pages 435–439. John Wiley & Sons, 1992.

[19] L. Gillick and S. J. Cox. Some statistical issues in the comparison of speech recognition algorithms. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 532–535, Glasgow, May 1992.

[20] Margaret Gurney and Robert S. Jewett. Construction orthogonal replications for variance estimation. *Journal of the American Statistical Association*, 70(532):819–821, December 1975.

[21] I. N. Herstein. *Topics in Algebra.* Xerox College Publishing, 2nd edition, 1975.

[22] Mike James. *Classification Algorithms.* John Wiley & Sons, 1985.

[23] David D. Jensen. *Induction with Randomization Testing: Decision-Oriented Analysis of Large Data Sets.* PhD thesis, Washigton University, May 1992.

[24] Trevor Johnson and John Keeves. Spending on the selling of wisdom. *Issues in Educational Research*, 10(1):39–54, 2000. Also available at `http://education.curtin.edu.au/lier/lier10/johnson.html`.

[25] David R. Judkins. Fay's method for variance estimation. *Journal of Official Statistics*, 6(3):223–239, 1990.

[26] Leslie Kish. Confidence intervals for clustered samples. *American Sociological Review*, 22(2):154–165, April 1957.

[27] Leslie Kish and Martin R. Frankel. Inference from complex samples. *Journal of the Royal Statistical Society B*, 36(1):1–37, 1974.

[28] Jack P. C. Kleijnen. *Statistical Techniques in Simulation*, volume 1. Marcel Dekker, 1974.

[29] Ron Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the International Join Conference on Artificial Intelligence (IJCAI)*, 1995.

[30] D. Krewski and J. N. K. Rao. Inference from statified samples: Properties of the linearization, jackknife and balanced repeated replication methods. *The Annals of Statistics*, 9(5):1010–1019, 1981.

[31] R. Lehtonen and E. J. Pahkinen. *Practical Methods for Design and Analaysis of Complex Surveys.* Statistics in Practice. John Wiley & Sons, 1995.

[32] P. J. McCarthy. *Replication: An Approach to the Analysis of Data From Complex Surveys.* U.S. Government Printing Office, April 1966.

[33] Quinn McNemar. *Psychological Statistics.* Wiley & Sons, 1949.

[34] Ross J. Micheals and Terrance E. Boult. Replicate statistics for efficient vision system evaluation. Technical report, Lehigh University, December 2000. Available through `http://www.eecs.lehigh.edu/˜rjm2`.

[35] Ross J. Micheals and Terrance E. Boult. Efficient evaluation of classification and recognition systems. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2001)*, Hawaii, December 11–13 2001.

[36] Eric W. Noreen. *Computer Intensive Methods for Testing Hypotheses.* John Wiley & Sons, 1989.

[37] P. Jonathon Phillips, Hyeonjoon Moon, Syed A. Rizvi, and Patrick J. Rauss. The FERET evaluation methodology for face-recognition algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10):1090–1104, October 2000.

[38] R. L. Plackett and J. P. Burman. The design of optimum multifactorial experiments. *Biometrika*, 33(4):305–325, June 1946.

[39] Yoram Reich and S. V. Barai. Evaluating machine learning models for engineering problems. *Artificial Intelligence in Engineering*, 13(3):257–272, 1999.

[40] K. F. Rust and J. N. K. Rao. Variance estimation for complex surveys using replication techniques. *Statistical Methods in Medical Research*, 5:283–31, 1996.

[41] Steven L. Salzberg. On comparing classifiers: Pitfalls to avoid and a recommended approach. *Data Mining and Knowledge Discovery*, 1(3), 1997.

[42] Steven L. Salzberg. On comparing classifiers: A critique of current research and methods. *Data Mining and Knowledge Discovery*, 1:1–12, 1999.

[43] Jun Shao and C. F. J. Wu. Asymptotic properties of the balanced repeated replication method for sample quantiles. *Annals of Statistics*, 20(3):1571–1593, September 1992.

[44] R. R. Sitter. Balanced repeated replications based on orthogonal multi-arrays. *Biometrika*, 80(1):211–221, March 1993.

[45] Godfried T. Toussaint. Bibliograph on estimation of misclassification. *IEEE Transactions on Information Theory*, IT-20(4):472–479, July 1974.

[46] Richard Valliant, Alan H. Dorfman, and Richard M. Royall. *Finite Population Sampling and Inference.* John Wiley & Sons, 2000.

[47] Kirk M. Wolter. *Introduction to Variance Estimation.* Springer series in statistics. Springer-Verlag, 1985.