Social Network Analysis and Mining (2021) 11:73
https://doi.org/10.1007/s13278-021-00760-0

ORIGINAL ARTICLE

A bootstrapping approach to social media quantification

Ashlynn R. Daughton[1] · Michael J. Paul[2]

Received: 10 January 2020 / Revised: 18 May 2021 / Accepted: 20 May 2021 / Published online: 9 August 2021
© The Author(s) 2021

Abstract
This work considers the use of classifiers in a downstream aggregation task estimating class proportions, such as estimating the percentage of reviews for a movie with positive sentiment. We derive the bias and variance of the class proportion estimator when taking classification error into account to determine how to best trade off different error types when tuning a classifier for these tasks. Additionally, we propose a method for constructing confidence intervals that correctly adjusts for classification error when estimating these statistics. We conduct experiments on four document classification tasks comparing our methods to prior approaches across classifier thresholds, sample sizes, and label distributions. Prior approaches have focused on providing the most accurate point estimate while this work focuses on the creation of correct confidence intervals that appropriately account for classifier error. Compared to the prior approaches, our methods provide lower error and more accurate confidence intervals.

Keywords Quantification · Classification bias · Confidence interval · Uncertainty · Social media · Public health

1 Introduction

Classifiers are often used in a pipeline toward a downstream task, i.e., the classifier's outputs are used as inputs in another step within a pipeline. This paper considers the downstream goal of obtaining a statistic, specifically, after classifying documents finding the proportion of positively classified instances. This scenario arises often, for example, when using a sentiment classifier on documents to measure overall sentiment of a corpus. This involves aggregating the individually classified messages to calculate an overall sentiment level, e.g., the percentage of messages classified as positive. This task is known as *quantification* in the data mining community (Forman 2008).

Quantification is challenging because errors introduced in the classification process will cause downstream statistics to be biased. Since no classifier is perfect, we seek to study and address how classification error affects this type of analysis. Moreover, a better understanding of the downstream effect of classifier error will provide a better understanding of the optimal tradeoff of error types when tuning a classifier (e.g., precision versus recall). We discuss these motivating issues in Sect. 2, as well as a discussion of prior work in this area.

Our paper has three main contributions over prior work:

- We characterize the estimator—bias, variance, and error—of sample proportions when the sample contains noisy classifications.
- We propose a more accurate method of constructing confidence intervals around class proportions estimated from noisy samples.
- We conduct experiments using our method across classifier thresholds, sample sizes, and class proportions.

We present these contributions from both a theoretical and empirical perspective. To understand the results empirically, we experiment with four datasets for the task of document classification of user-generated content: aggregating the sentiment in movie reviews, and measuring rates of vaccination from social media posts. This work expands on our prior work (Daughton and Paul 2019).

✉ Ashlynn R. Daughton
adaughton@lanl.gov

Michael J. Paul
michael.j.paul@colorado.edu

[1] Los Alamos National Laboratory, Los Alamos, USA

[2] University of Colorado, Boulder, USA

2 Background

2.1 Quantifying class proportions

The quantification problem was first described in seminal work by Forman, who showed that classification errors introduce systematic bias into the calculation of the number of positives (Forman 2005, 2006, 2008). He used the term "classify and count" to describe the naïve quantification approach of simply counting the number of positively classified instances and proposed several methods for adjusting the counts based on the true and false positive rates of the classifier, with some methods motivated specifically for data with imbalanced classes (Forman 2008). For a more comprehensive review of quantification see González et al. (2017). Here, we present a review of some methods for binary quantification to better contextualize this work.

Within the quantification literature, there are several methods that follow the "classify and count" dogma, but add a third step that corrects the estimate using some combination of normalization and classifier metrics. One of the most commonly cited is Forman's *Adjusted Count* which relies on the true positive rate (*tpr*) and false positive rate (*fpr*) to adjust the class proportion (see equation 2). These methods typically rely on the assumption that the true proportion positive (p) are the same in the training and test data, and that thus the true value of p relies only on the classifier (González et al. 2017). Others, like Bella et al. (2010), have proposed the use of the probability average of the classifier to adjust estimates (see equation 3). Their extension of this method to the scaled probability average simply further adjusts this estimate by taking into account the positive predictive values of the positive and negative classes (see equation 4).

Others have used more complex methods, including the expansion of the learning phase to include quantification estimation. Milli et al. (2013) proposed the use of modified decision trees, called *quantification trees* which are optimized for quantification rather than for classifying individual instances. Barranquero et al. (2013) investigated the use of $k-$nearest neighbor and weighted $k-$nearest neighbor algorithms for quantification with the observation that such algorithms benefit from more efficient methods to estimate classifier metrics.

Further, quantification methods are closely related to other concepts in the machine learning community. For example, the task *learning from label proportions* is the inverse of quantification: it applies when the class proportions are known, but individual labels are unknown (Kück and de Freitas 2005; Musicant et al. 2007; Quadrianto et al. 2009; Yu et al. 2013). Methods of learning from label proportions are used to learn to classify individual instances when only aggregate statistics are available. Some work has applied these techniques to social media tasks, including learning to classify user demographics (Ardehaly and Culotta 2017) and estimating political surveys (Benton et al. 2016).

There are a number of additional implications related to quantification. For example, some have extended the quantification algorithms to explore the effect of concept drift on quantification (Xue and Weiss 2009; Pérez-Gállego et al. 2017) and to count ordinal values (Da San Martino et al. 2016).

2.2 Quantification in practice

Quantification is an increasingly widespread application of user-generated content such as social media posts. For example, many have measured public sentiment and attitudes at a large scale by aggregating the results of sentiment models applied to individual messages online (O'Connor et al. 2010; Diakopoulos and Shamma 2010; Bollen et al. 2011; Wang et al. 2012; Mitra et al. 2016). Prior work has shown that the prevalence of influenza can be estimated from the number of tweets mentioning an influenza infection (Culotta 2010; Lamb et al. 2013; Sadilek et al. 2012). Others have used classifiers to study behavior in online communities (Yin et al. 2017; Mitra et al. 2016) or patterns in news coverage (West and Pfeffer 2017).

All of the studies cited above use what is known as the "classify and count" method of quantification (Forman 2008), though they did not refer to it as such; indeed, much work on aggregating user-generated content does not reference related work on quantification, even though quantification is implicitly being performed. We were able to find only a small number of studies that used adjustments when quantifying user-generated content, mostly in the domain of sentiment analysis in social media (Gao and Sebastiani 2015, 2016; Nakov et al. 2016; Sebastiani 2018).

To the best of our knowledge, no previous work has fully characterized the expected error of "classify and count" quantification. While Forman (2008) showed that it is biased, and when the bias is an overestimate versus an underestimate, he did not provide a complete formulation of the bias, nor has prior work derived the variance of the estimator (Forman 2008). We derive both in Sect. 4. In our experiments, we also show how the theoretical compares to the empirical error. This type of analysis can provide insight into the tradeoff of different error types.

Second, all previously proposed quantification methods have focused on producing point estimates of class proportions. We argue that for many quantification tasks it is useful to provide confidence intervals around the estimate; indeed, many of the social media studies we cited above constructed confidence intervals or similar statistics, but did not adjust for classification error. Our work tests if the point estimate produced by traditional bootstrapping (the average of all estimates) is more accurate than those produced using the entire sample. We then present an adjusted method for constructing bootstrap-based confidence intervals to correctly account for classification error, described in Sect. 5.

In our experiments, we show that naïvely-constructed confidence intervals are highly inaccurate, and our proposed algorithm is more accurate than simply constructing confidence intervals using statistics adjusted with Forman's methods. This approach can positively impact research that uses quantification. Even a low-performing classifier can be used in downstream analyses and hypothesis-testing, albeit with low statistical power, as long as the uncertainty is correctly quantified.

Our proposed confidence interval adjustment is somewhat related to other methods of accounting for measurement error (Stram et al. 1999; Barbiero and Manzi 2015; Buonaccorsi et al. 2018). Most similar to our work is that of Szpiro and Paciorek (2014), who adjust for errors in inferences that are used for downstream epidemiological analysis. However, their work is focused on complex exposure models rather than classifiers, and the error model is different from the classification errors we consider in this work.

3 Preliminaries

3.1 Binary classification

This paper focuses on binary classification, though the approach is straightforward to generalize to multiclass settings.

The training dataset contains N instances $X_i \in \mathbb{R}^M$ paired with labels $Y_i \in \{0, 1\}$. A classification function $f(X_i;\Theta)$ returns a predicted label $\hat{Y}_i \in \{0, 1\}$ for instances whose labels are unknown. We refer to the predicted labels as classifications, and the classification function as a classifier.

This paper will generally describe classifiers as traditional machine learning models, in which the parameters Θ are learned to optimize performance on training and validation data. However, our analyses also apply to classification functions using rule-based or dictionary-based approaches, which have been used in the some of the work cited above (O'Connor et al. 2010; Culotta 2010; West and Pfeffer 2017), as long as there is still labeled data on which the classifier can be evaluated.

3.2 Classification error

Various evaluation metrics are used to measure the reliability of a classifier, typically measured on held-out test data. In machine learning, the most common metrics are precision and recall. Most of our analyses in this paper use the true and false positive rates, where the true positive rate is the percent of positive instances correctly classified as positive (equivalent to recall), and the false positive rate is the percent of negative instances classified as positive. These correspond to the maximum likelihood estimates of $P(\hat{Y}_i = 1|Y_i = 1)$ and $P(\hat{Y}_i = 1|Y_i = 0)$, respectively.

Table 1 Definitions of classification metrics used in our analyses, estimated as functions of the number of true and false positives (*TP* and *FP*) and true and false negatives (*TN* and *FN*)

Metric	Notation	Defn. (Probability)	Defn. (Estimate)
True positive rate	α	$P(\hat{Y} = 1\|Y = 1)$	$TP/(TP + FN)$
False positive rate	β	$P(\hat{Y} = 1\|Y = 0)$	$FP/(FP + TN)$
True negative rate	$1 - \alpha$	$P(\hat{Y} = 0\|Y = 0)$	$TN/(TN + FP)$
False negative rate	$1 - \beta$	$P(\hat{Y} = 0\|Y = 1)$	$FN/(TP + FN)$
Positive predictive value		$P(Y = 1\|\hat{Y} = 1)$	$TP/(TP + FP)$
Negative predictive value		$P(Y = 0\|\hat{Y} = 0)$	$TN/(TN + FN)$

Classifiers can be tuned to raise or lower the true and false positive rates. By lowering the threshold for a positive classification, more instances will be classified as positive, increasing recall while also increasing the false positive rate. This is often visualized as the receiver operating characteristic (ROC) curve, which shows the true positive rate against the false positive rate. This is similar to a precision-recall curve, which is more common in machine learning, where the false positive rate is replaced with precision.

Precision is also called the positive predictive value, which is the maximum likelihood estimate of $P(Y_i = 1|\hat{Y}_i = 1)$. This value is used to construct confidence intervals in Sect. 5.

Table 1 summarizes the measurements used in this paper.

3.3 Class proportions

After applying the classifier to data, we consider the question: how many instances were classified positive? We consider the proportion of positively classified instances as an estimator of the true proportion of positive instances:

$$\hat{p} = \frac{1}{N}\sum_{i=1}^{N} \hat{Y}_i \tag{1}$$

This can be generalized to multiple classes by treating the target class as positive and all others as negative. A proportion is a special case of the mean when the values are binary, so standard results of using sample means as estimators apply to the sample proportion. However, our analyses do not apply to means in general; they assume binary Y, so we will refer specifically to proportions.

4 Estimator properties

This section derives the bias, variance, and mean squared error of the estimator $\hat{p}$ (Eqn. 1) (Lehmann and Casella 1998). These properties depend on the true proportion p as well as the true positive rate of the classifier (denoted α in this section) and false positive rate (denoted β), where $1 \geq \alpha \geq \beta \geq 0$.

4.1 Bias

If $\hat{Y}_i$ is noise-free ($\hat{Y}_i = Y_i$), then the sample proportion is unbiased. However, in this section, we show that the estimator is biased when $\hat{Y}_i$ depends on classifier error.

Lemma 1 *Let $\hat{p}$ be the sample estimate of p, the proportion of positively labeled instances in a collection. Let α be the true positive rate of the classifier, and β be the false positive rate.*

The bias of the estimator $\hat{p}$ is: $\mathrm{E}[\hat{p} - p] = \alpha p + \beta(1-p) - p$.

Proof Assuming classifications $\hat{Y}_i \in \{0, 1\}$ are i.i.d., then $\mathrm{E}[\hat{p}] = \mathrm{E}[\hat{Y}_i]$, and $\mathrm{E}[\hat{Y}_i] = P(\hat{Y}_i = 1)$

$$\begin{aligned} &P(\hat{Y}_i{=}1|Y_i{=}1)P(Y_i{=}1){+}P(\hat{Y}_i{=}1|Y_i{=}0)P(Y_i{=}0) \\ &\quad = \alpha p + \beta(1-p), \end{aligned}$$

since $P(\hat{Y}_i = 1|Y_i = 1)$ corresponds to the true positive rate, $P(\hat{Y}_i = 1|Y_i = 0)$ corresponds to the false positive rate, and $P(Y_i = 1)$ corresponds to the true proportion of positive instances. □

From this, there are two straightforward corollaries about when the estimator is unbiased in two extreme cases: the classifier makes no mistakes, and the classifier is no better than random.

Corollary 1 *The estimator $\hat{p}$ is unbiased when $\alpha = 1$ and $\beta = 0$ (i.e., the classifier is perfect).*

Corollary 2 *The bias of estimator $\hat{p}$ is $\alpha - p$ when $\alpha = \beta$ (i.e., the classifier is no better than random). When $\alpha = \beta = p$, the estimator is unbiased.*

While not common in practice, these two scenarios intuitively demonstrate the properties of the estimator. If the classifier is perfect, then the estimator is simply the sample proportion, which is unbiased. If the classifier predictions are random, then it is unbiased if the probability of making a positive prediction is equal to the actual proportion of positive instances.

More generally, we consider the relationship between α and β as a third corollary.

Corollary 3 *The estimator $\hat{p}$ is unbiased when*:

$$\alpha = 1 + \beta - \frac{\beta}{p}, \qquad \beta = \frac{p - \alpha p}{1-p}.$$

To show this relationship more clearly, the top row of Fig. 1 shows the bias at various values of α, β, and p. For readability, we only show a few values of β, rather than the full range of values of α and β. We show β as a function of α; one in which β is close to α, one in which it is much smaller than α, and one in between.

For the bias to be zero, α should increase as p increases. We see in the figure that there is a diagonal band of near-zero values that moves upward along α values as p increases. However, the position of the band depends on the value of β. As β decreases, the band moves toward the upper left, favoring larger α values even at low p.

4.2 Variance

Next, we examine variance of the estimator $\hat{p}$.

Lemma 2 *Let $\hat{p}$ be the sample estimate of p, with a sample size of n. Let α be the true positive rate of the classifier, and β be the false positive rate. The variance of the estimator $\hat{p}$ is*: $\mathrm{Var}(\hat{p}) = \frac{1}{n}\left[(\alpha p + \beta(1-p))[1 - (\alpha p + \beta(1-p))]\right]$.

Proof By standard results, $\mathrm{Var}(\hat{p}) = \frac{1}{n}\mathrm{Var}(\hat{Y}_i)$, and

$$\begin{aligned} \mathrm{Var}[\hat{Y}_i] &= \mathrm{E}[(\hat{Y}_i - \mathrm{E}[\hat{Y}_i])^2] \\ &= (1 - \mathrm{E}[\hat{Y}_i])^2 P(\hat{Y}_i = 1) + (0 - \mathrm{E}[\hat{Y}_i])^2 P(\hat{Y}_i = 0) \\ &= (1 - (\alpha p + \beta(1-p)))^2(\alpha p + \beta(1-p)) \\ &\quad + (-(\alpha p + \beta(1-p)))^2(1 - (\alpha p + \beta(1-p))) \\ &= (\alpha p + \beta(1-p))[1 - (\alpha p + \beta(1-p))], \end{aligned}$$

where we have that $\mathrm{E}[\hat{Y}_i] = P(\hat{Y}_i = 1) = \alpha p + \beta(1-p)$ from Lemma 1, and $P(\hat{Y}_i = 0) = 1 - P(\hat{Y}_i = 1)$. □

Corollary 4 *The variance of the estimator $\hat{p}$ is minimized when $(\alpha p + \beta(1-p)) = 0$ or $(\alpha p + \beta(1-p)) = 1$. This condition is satisfied when any of the following relationships are true*:

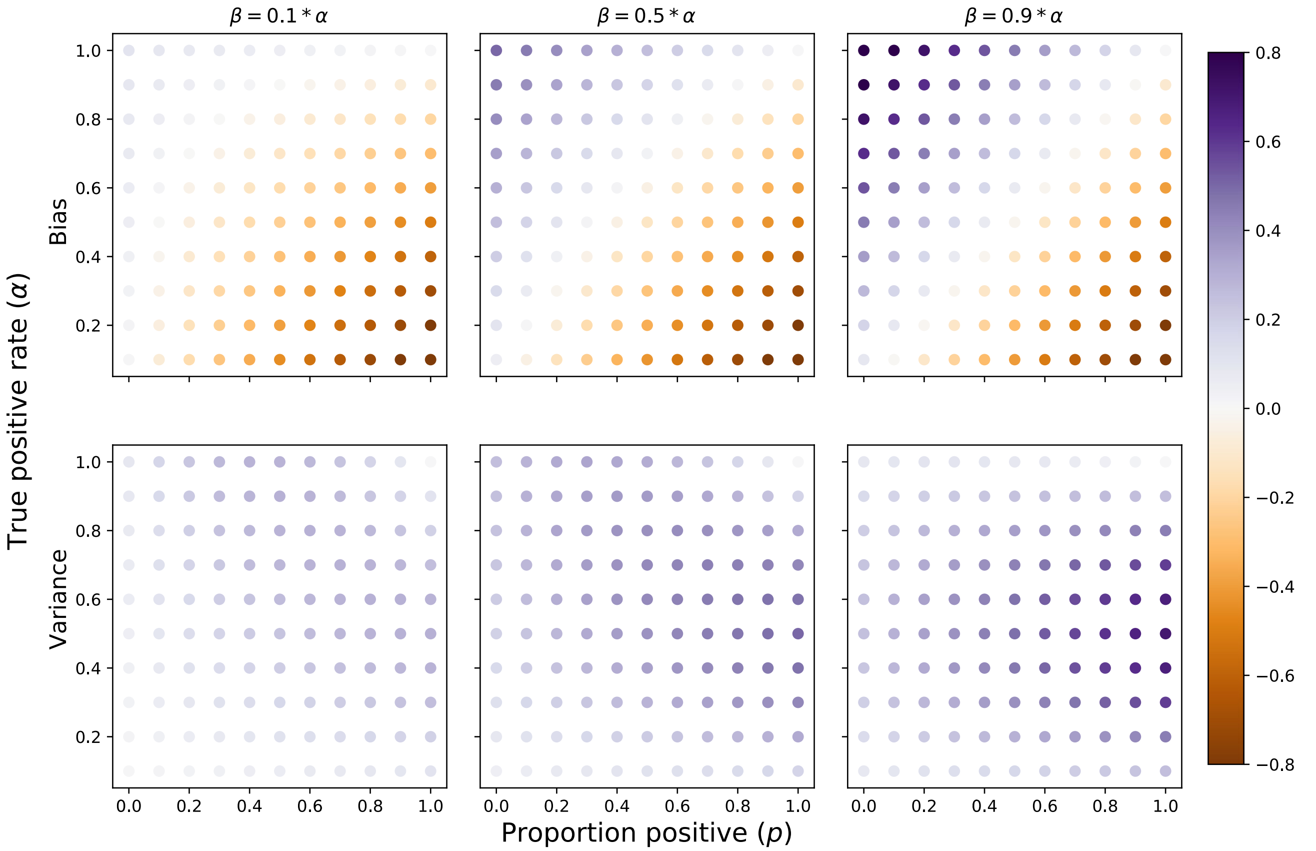


Fig. 1 Bias (top) and variance (bottom) of the estimated proportion of positive instances ($\hat{p}$) as a function of the true positive rate (α), false positive rate ($\beta \leq \alpha$), and the true proportion of positive instances (p)

$$\alpha = \frac{-\beta(1-p)}{p}, \alpha = \frac{1-\beta(1-p)}{p},$$
$$\beta = \frac{-\alpha p}{1-p}, \beta = \frac{1-\alpha p}{1-p}.$$

The bottom row of Fig. 1 shows the variance when $n = 1$. The variance tends to shrink as p and α shrink, with a stronger pattern when β is smaller. For most values of p, variance tends to be lower when α is lower (i.e., lower recall, but fewer false positives).

4.3 Error

Finally, the mean squared error (MSE) of the estimator is given by: $\text{MSE}(\hat{p}) = \text{Bias}(\hat{p})^2 + \text{Var}(\hat{p})$.

We experimentally compare this expected error to the actual error on real datasets. In practice, estimating the theoretical error requires knowledge of p, which is unknown. We suggest that for the purpose of estimating the error, p can be set to its value in historical data.

5 Confidence intervals

It is important to be able to quantify the certainty of an estimate, for example with a confidence interval of the estimate. Traditionally, confidence intervals are a function of sample size and variability in the data. However, when estimating statistics from classifiers, an additional layer of uncertainty is introduced, as not all instances will be labeled correctly. Here, we present a nonparametric approach to constructing a confidence interval of $\hat{p}$ based on bootstrapping.

5.1 Bootstrapping: Review

Bootstrapping, or bootstrap resampling, is a procedure to simulate the statistics one would obtain when sampling from a distribution (Efron 1979; Efron and Tibshirani 1993). A bootstrapped estimate (for example, an estimate $\hat{p}$) is obtained by sampling N instances with replacement from the original dataset of size N, then calculating the statistic (e.g., $\hat{p}$) on the set of sampled instances. This procedure can be

Algorithm 1: Error-adjusted bootstrap resampling

```
Data: Set of $N$ instances classified as $\hat{Y}_i \in \{0,1\}$
Input: Number of bootstrap samples, $T$
Output: $S$, a set of $T$ estimates of $\hat{p}$
$S = \{\}$
for $1 \leq t \leq T$ do
    $\mathbf{y} = []$
    for $1 \leq i \leq N$ do
        Sample instance $j \in \{1, 2, \ldots, N\}$;
        if $\hat{Y}_j = 1$ then
            Sample $y \sim P(Y_j = y | \hat{Y}_j = 1)$;
        else
            Sample $y \sim P(Y_j = y | \hat{Y}_j = 0)$;
        end
        $\mathbf{y}$ += $[y]$;
    end
    $\hat{p} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{y}_i$;
    $S = S \cup \{\hat{p}\}$;
end
return $S$
```

repeated many times to obtain many bootstrapped estimates, providing a distribution over estimates.

To construct a $c\%$ confidence interval, the bootstrapped estimates can be sorted, and the range of the middle $c\%$ of values can be taken as the interval.

5.2 Error-adjusted bootstrapping

If bootstrapping is applied to noisy classifications rather than true labels, then the samples will not be drawn from the correct distribution. We propose an adjustment to the sampling procedure that draws from the actual distribution of the data.

For each bootstrap sample, after selecting the instances (sampled with replacement), we randomly sample the labels of the instances according to the confusion matrix of the classifier. If an instance is classified positive, we sample the label according to $P(Y|\hat{Y}_i = 1)$; if an instance is classified negative, we sample the label according to $P(Y|\hat{Y}_i = 0)$. In this way, rather than treating the classifications as labels directly, we sample labels based on the probability that the classifier predicted an incorrect label. This procedure simulates the classification process in addition to the sampling process when obtaining an estimate.

We refer to this approach as *error-adjusted* bootstrapping. The steps for obtained a set of error-adjusted bootstrapped samples are detailed in Algorithm 1.

5.2.1 Correctness of algorithm

The underlying assumption of bootstrap resampling is that the instances are i.i.d. and that uniformly sampling an instance is a draw from $P(Y)$. If the distribution of classifications $P(\hat{Y})$ is different from the distribution of labels $P(Y)$, then randomly sampling from the classifier outputs will not correctly draw from $P(Y)$.

Our approach uses the distribution $P(\hat{Y})$ and predictive values $P(Y|\hat{Y})$ to correctly calculate $P(Y)$: $P(Y_i{=}y) = P(Y_i{=}y|\hat{Y}_i{=}0)P(\hat{Y}_i{=}0) + P(Y_i{=}y|\hat{Y}_i{=}1)P(\hat{Y}_i{=}1)$.

As a generative process, sampling from this marginal distribution corresponds to the following steps for each instance i: (i) Sample $\hat{y}_i \sim P(\hat{Y})$; (ii) Sample $y_i \sim P(Y|\hat{Y}_i = \hat{y}_i)$. This matches Algorithm 1, which thus samples a label y according to the true label distribution $P(Y)$ rather than the classification distribution $P(\hat{Y})$.

5.2.2 Predictive value estimates

As described so far, we assume the positive predictive value $P(Y|\hat{Y} = 1)$ and negative predictive value $P(Y|\hat{Y} = 0)$ are known. We propose two approaches to estimating these values. The first is to use cross-validation to provide point estimates of the positive and negative predictive values at each threshold of interest. This is the same approach used in prior work (Forman 2008).

The second approach extends Algorithm 1 to use a posterior distribution over predictive values. We do this by fitting a beta distribution to the individual estimates from cross-validation. We then draw a new estimate of the predictive values before sampling each label y_j during bootstrapping. We refer to this in experiments as the **extended** algorithm.

6 Experiments

We now experiment with estimating class proportions using document classifiers on two datasets. There are two goals of these experiments: first, to validate the above theory experimentally, and second, to show how class estimates vary in practice as the classification threshold is adjusted.

6.1 Datasets and classification details

We experiment with binary document classification on four datasets of online user-generated content:

- **Flu Vaccination:** A set of 10,000 tweets labeled with if the tweet indicates that someone has received an influenza vaccination (i.e., a seasonal flu shot) (Huang et al. 2017) from 2013-2016. The dataset spans four years, and approximately 31% of tweets are labeled positive. 15% of tweets were reserved for testing. The aggregation task is to calculate the percent of tweets that indicate vaccination each month.
- **Flu Infection:** A set of 1,017 tweets from (Lamb et al. 2013) from 2009 labeled as indicating flu infection. The original dataset included 5,000 tweets, but most are no longer available for download. The aggregation task is to calculate the percent of tweets indicating flu infection each week of available data. Again, 15% of tweets were reserved for testing.
- **IMDB:** A set of 50,000 movie reviews labeled with positive or negative sentiment (Maas et al. 2011). The dataset is balanced so that there is an equal number of reviews for positive and negative sentiment, and contains reviews for 2,780 movies (average 18 reviews per movie). The IMDB data come split 50/50 into training and testing. The aggregation task is to calculate the percentage of reviews that are positive for each movie.
- **Yelp:** A set of 6,752,287 reviews across 192,632 businesses on Yelp. The full dataset is available here.[1] Reviews with greater than 3 stars were labeled "positive" reviews, and ≤ 3 stars were considered negative reviews. Because this dataset was so large, we created a smaller dataset using a random sample of 10% of the businesses. Of these reviews 15% were reserved for testing (584,841 reviews in the training dataset and 103,208 in the test dataset). Approximately 77% of the reviews were positive.

Classification was done using binary logistic regression classifiers implemented with `scikit-learn` (Pedregosa and others 2011). Grid search using fivefold cross-validation on the training data was used to tune the ℓ_2 regularization parameter. For all classifiers, unigrams were used to build feature sets. While more extensive feature engineering or feature selection techniques might result in higher performing classifiers, we constructed the experiments this way to create a simple and equitable comparison between the datasets. ROC curves are shown in Figure S1. We note that classification performance is extremely high for the Yelp dataset (area under the ROC curve is nearly 1), while error rates are higher for the Twitter datasets.

We experiment with different classification thresholds, meaning we set $\hat{y}_i = 1$ if $P(y_i = 1|x_i) > \tau$ for a threshold τ. Increasing the threshold will lower the true positive rate α while also lowering the false positive rate β, thus trading off different error types.

For the adjustment methods, we calculate the error rates and predictive values using cross-validation. In the extended version of Algorithm 1, we additionally sample the predictive values based on the cross-validation distribution.

6.1.1 Bootstrapping benefits

Before testing estimate adjustment methods, we first experiment to see if bootstrapping provides benefits over standard methods that provide a single point estimate. To do this, we compare the error when obtaining a point estimate to the average estimate obtained from a bootstrapped estimate (see Fig. 2).

Error rates between the two methods are extremely close. In many instances, the lines are close to entirely overlapping. Bootstrapping alone may provide a benefit, but it is extremely small. The most obvious benefit is in the flu infection dataset which is also the smallest. We also compare error between an adjusted bootstrapped estimate to the point estimate (see Fig. 2). For three out of four datasets, our adjustment reduces error across all thresholds. However, in the fourth, the flu infection dataset, error is actually larger in the adjusted estimate. We hypothesize that this is because this is the smallest dataset, so the estimates of the error rates may be less accurate.

6.2 Bootstrapped experiments

6.2.1 Baseline

We experimentally compare to the "adjusted counts" method from Forman (2008). Here, the true positive rate (α) and the false positive rate (β) are used to obtain an adjusted estimate of the percent of positive instances:

$$p \approx \frac{\hat{p} - \beta}{\alpha - \beta}, \tag{2}$$

[1] https://www.yelp.com/dataset

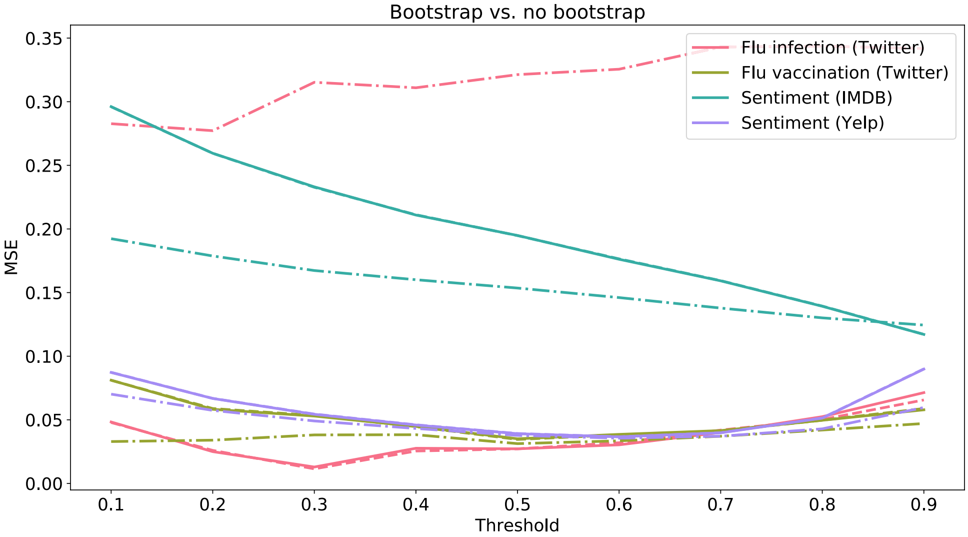


Fig. 2 Bootstrap importance. Colors show each dataset, dashed lines (- - -) show the unweighted bootstrap method – the point estimate is the average of the bootstrapped estimates. Solid lines show the point estimate without using bootstrapping. Dashed-dot lines (- .) show the point estimate when using Algorithm 1 to adjust the bootstrap sample

where $\hat{p}$ is the fraction estimated positive by the classifier. The estimate must be truncated to the range [0, 1]. While Forman (2008) introduced multiple methods for estimating p, the adjusted count method was selected for use as a baseline because it consistently performed well in general.[2] In our experiments we calculate the adjusted counts within each bootstrapping iteration, and then construct confidence intervals of the adjusted counts.

In Bella et al.'s scaled probability estimate (SPA) method, the probability estimates of the classifier are taken advantage of (Bella et al. 2010). The probability average (PA) is the average of the probabilities given by the classifier on the test dataset:

$$PA = \frac{\sum_i p_i}{n} \tag{3}$$

This is then scaled using the positive predictive value of the positive class and the positive predictive value of the negative class so that the final value is between [0,1]:

$$SPA = \frac{PA - PPV_{(-)}}{PPV_{(+)} - PPV_{(-)}} \tag{4}$$

6.3 Estimator error

We then calculate the mean squared error (MSE) of the classifier estimate of $\hat{p}$ on each test dataset, compared to the true proportion given by the labels. We then additionally apply bootstrapping to each group and estimate $\hat{p}$ as the mean of the proportions across the bootstrap samples. Doing so allows us to investigate if error-adjusted bootstrapping produces better estimates of $\hat{p}$ while also looking at our original motivation of producing more accurate confidence intervals. 100 bootstrap samples are collected in all experiments.

The top row of each panel of Fig. 3 show the observed MSE (orange) with and without making error adjustments during bootstrapping.

In general, differences in error are quite small. We also find that patterns vary across different datasets. Algorithm 1 results in a lower error than baseline methods in the flu infection dataset, but the Forman-adjusted method is the lowest in the flu vaccination and IMDB dataset, and the Bella-adjusted method is the lowest in the Yelp dataset (with Algorithm 1 coming in a close second).

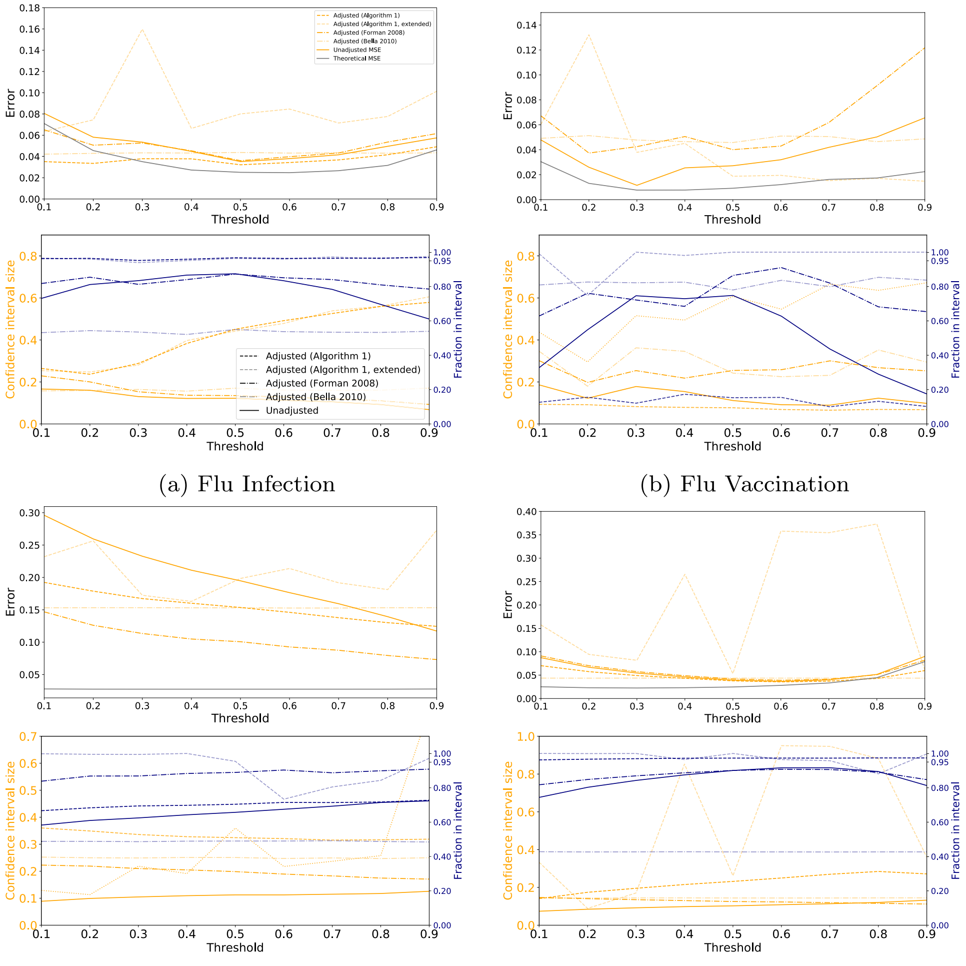
We also plot the theoretical MSE calculated using the cross-validation estimates of α and β, with p estimated from the training data. While the magnitude of the theoretical error does not perfectly match the observed error, the shape mimics the observed (unadjusted) error very closely.

6.4 Confidence intervals

We examine the empirical characteristics of 95% confidence intervals constructed using bootstrap sampling, with and without making various error adjustments. We look at two characteristics: the fraction of times that the true value is contained in the interval (which should be 95%, asymptotically), as well as the size of the intervals.

The bottom row of each panel in Fig. 3 show these characteristics. The blue lines show the fraction of correct values contained in the 95% confidence intervals. There is some variation across datasets, but we in general see that the confidence intervals constructed using error-adjusted bootstrapping correctly capture the true values around 95% of the time. There are two instances (the IMDB dataset, and the flu vaccination dataset) where the extended version of

[2] The other methods work best in cases of extreme class imbalance, a setting we do not consider in this work.



(a) Flu Infection

(b) Flu Vaccination

(c) IMDB Sentiment

(d) Yelp Sentiment

Fig. 3 *Top rows:* the mean squared error of estimating the proportion positive in test data at different classification thresholds, with and without making sampling adjustments, as well as the theoretical error based on the estimated true and false positive rates. *Bottom rows:* the size of 95% confidence intervals (orange) and fraction of true values contained within 95% confidence intervals (blue) at different classification thresholds, when constructing intervals with and without adjusting for error. With error-adjusted bootstrapping, the true value should theoretically be contained in the interval 95% of the time (shown by the dotted gray line)

Algorithm 1 more accurately captures the true values. There are also some instances where the extended Algorithm 1 captures the true value more than 95% of the time, suggesting that in some contexts this method may unnecessarily overcompensate for uncertainty in the predictive values.

Importantly, we see that doing traditional bootstrapping without adjusting for classification error can severely affect the reliability of the confidence intervals. In the flu infection dataset, the unadjusted 95% confidence interval is only correct 85% of the time at best and is as low as 65% at a suboptimal threshold. This is even more striking in the flu vaccination dataset, which is noticeably noisier than other datasets. Here, a 95% confidence interval behaves like a 20% confidence interval in the worst scenario.

In general, our extended algorithm outperforms other methods, including Algorithm 1. The Forman-adjusted count method is consistently more accurate than doing no adjustment, but is generally less accurate than our extended method. Bella's method seems to be the least accurate, except in the flu vaccination dataset where our algorithm performs the worst.

The orange lines show the size of the intervals, to quantify how much wider the intervals must be to correctly adjust for error. In general, traditional bootstrapping produces some of the smallest confidence intervals. The Forman-adjusted and Bella-adjusted methods produce slightly larger confidence intervals, and the confidence intervals produced using our algorithms are the widest.

6.4.1 Sample size

In these experiments, we vary the number of samples per group by randomly sampling with replacement to achieve a predefined number of instances per group ($n = 10, 20, 30$, or 40). This experiment and the class distribution experiment presented in 6.4.2 were not performed on the IMDB dataset because the test dataset is constructed such that all reviews for a given movie are either positive or negative, though the distribution overall is 50%. This makes it incompatible with the sampling scheme that we used for these experiments.

The results of these experiments on individual datasets are shown in supplementary Figs. S2 – S7. We see that the

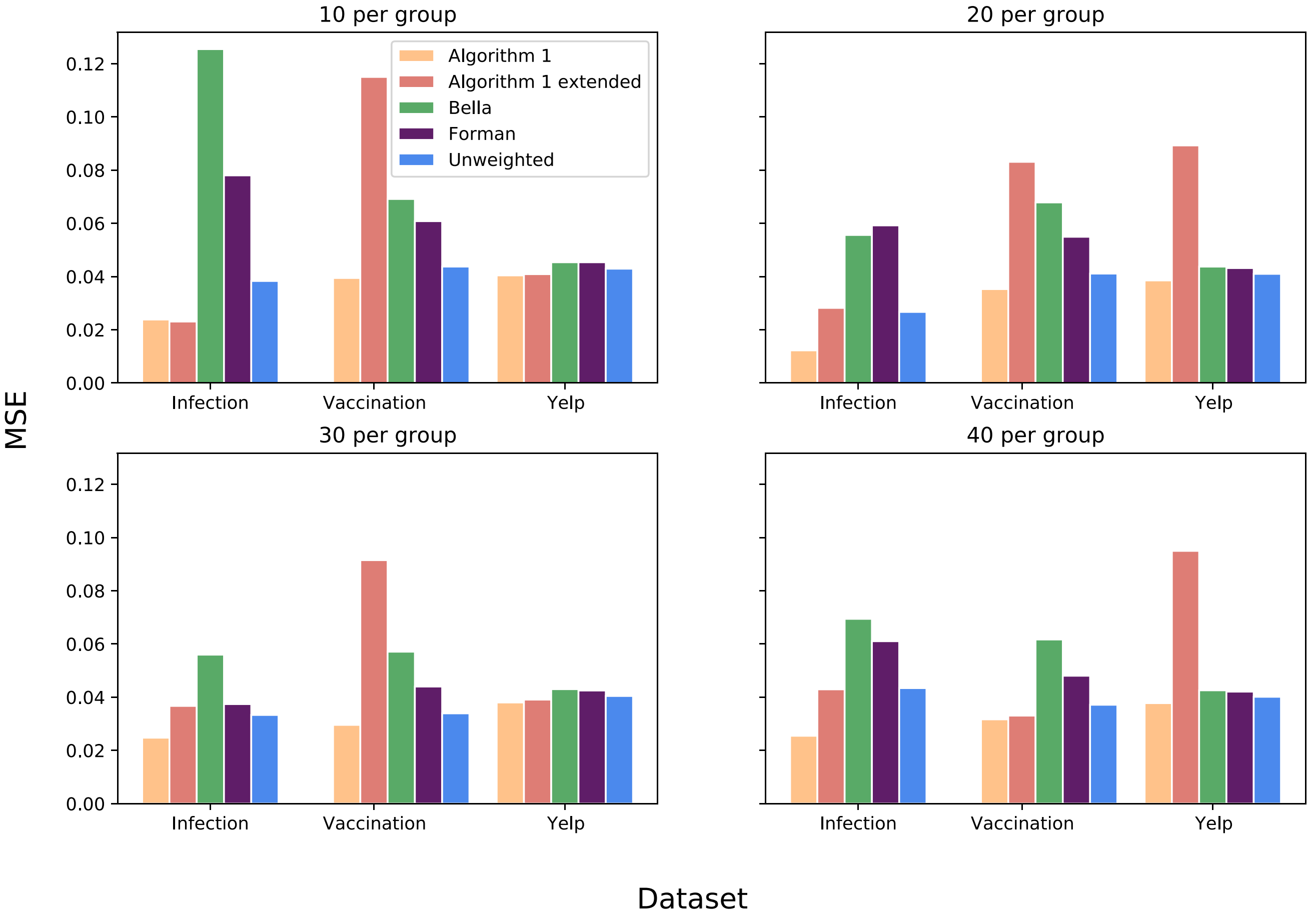


Fig. 4 Mean square error stratified by sample size across all datasets and correction methods. Data are shown at a threshold of 0.5

error for Algorithm 1 (e.g., Fig. S3) is among the smallest, though Bella et al.'s method is often a bit smaller. However, Algorithm 1 also has much closer to the right fraction in the confidence interval (e.g., Fig. S2). Note that the extended algorithm performs better in this capacity even as sample size changes, while the regular Algorithm 1 starts to drop at threshold extremes (though it still tends to outperform Bella and Forman in these contexts).

Data aggregated across datasets at a threshold of 0.5 are shown in Figs. 4 and 5. While error rates vary, patterns are roughly the same regardless of sample sizes. The Algorithm 1 method has the lowest error in almost every dataset while the extended method and the Bella-adjusted methods have the highest. This is unsurprising given that the extended method typically generates a confidence interval that is slightly larger than optimal. In addition, we see that as the number of items per group increases, all methods produce more accurate confidence intervals (Fig. 5), but our method is arguably the most consistent.

6.4.2 Class distribution

Next, we consider the impact of class distribution on each method. To sample to achieve the desired fraction positive, data were categorized into respective groups (e.g., each week in the Twitter data, or each business in the Yelp data). Within each group, we weighted the new sample by the desired fraction positive. For example, in the Yelp data, if the desired fraction positive was 25%, there was a 25% chance that we would pick any positive review and a 75% chance of selecting a negative review. The sampling was only performed if there were at least 10 items in the group. Again, we did not use the IMDB data for this.

Supplementary Figs. S8 – S13 show the results of these experiments across all thresholds on the individual datasets. In general, we again see that Algorithm 1 outperforms all baselines regardless of the class distribution and that the Forman method performs better than the Bella et al. method.

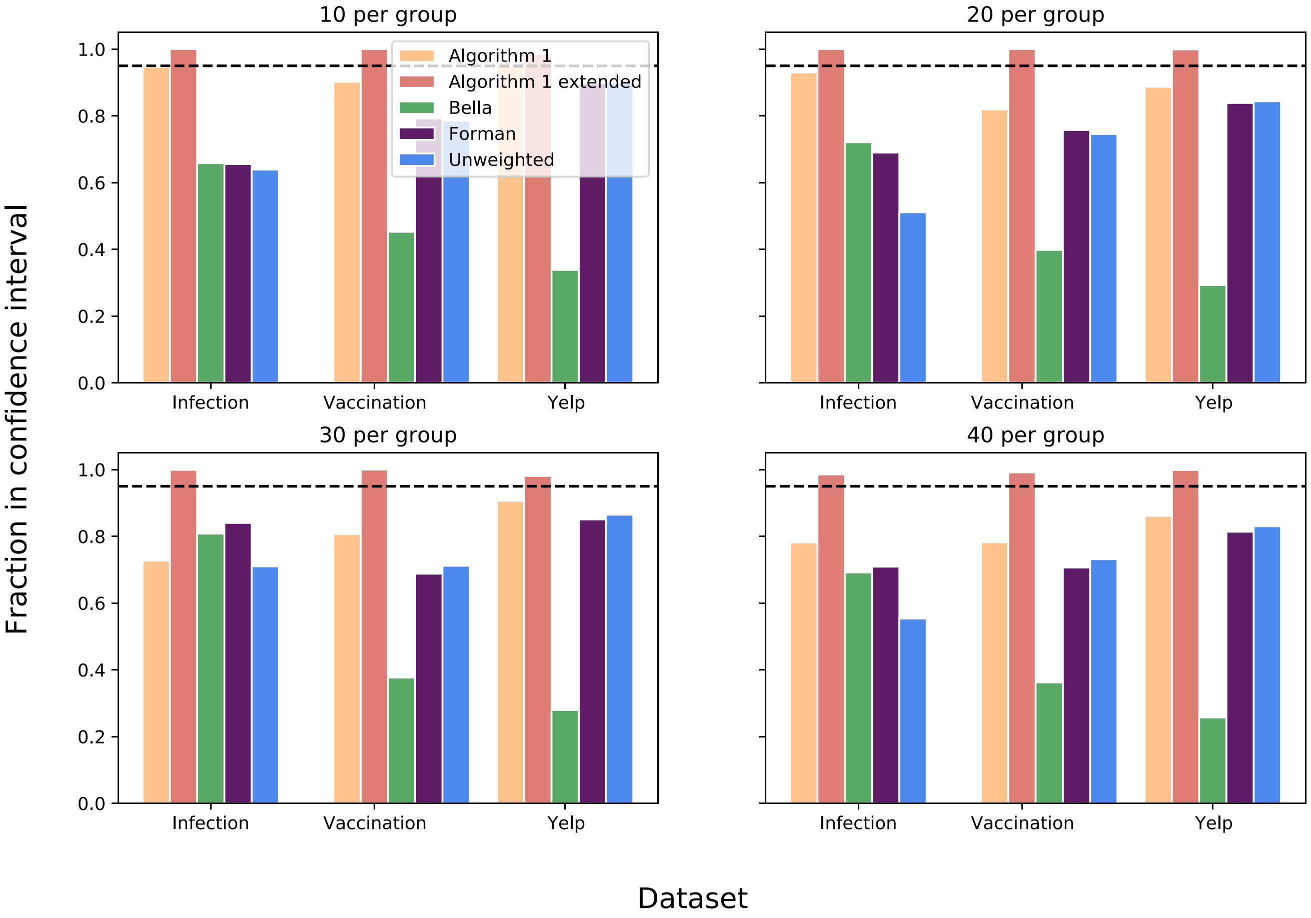


Fig. 5 Fraction of true values included in the confidence interval stratified by sample size across all datasets and correction methods. Data are shown at a threshold of 0.5

Data aggregated across datasets at a threshold of 0.5 are shown in Figs. 6 and 7. Algorithm 1 produces the smallest error, followed generally by the extended Algorithm 1 and Bella et al. Again we also see that the Algorithm 1 and extended Algorithm 1 methods produce the most consistent, and most accurate confidence intervals across all class distributions, and the Bella et al. and Forman methods perform more modestly in cases of greater class imbalance.

6.5 Use case: Vaccination surveillance

Lastly, we consider how this type of analysis relates to one of the motivating applications, which is using the proportion of vaccine-related tweets to measure vaccination rates in a population. To do this, we applied the classifier trained on the Twitter dataset to a larger set of approximately 1 million tweets, from 2017. At different classification thresholds, we estimate the proportion of positive tweets in each month, and we compare these proportions to official flu vaccination data from the US Centers for Disease Control and Prevention (CDC), to evaluate how well monthly variations in vaccine tweets track true vaccination behavior (Huang et al. 2017). We measure this with Pearson correlation, calculating the proportions using adjusted bootstrapping from Algorithm 1 versus no adjustment.

Figure 8 shows the correlations between Twitter proportions and CDC data. With the exception of a large and unexplained drop in correlation when the threshold is 0.6, the correlation closely follows the mean squared error in Fig. 3, with an optimal correlation at a threshold of 0.5, and with low thresholds resulting in generally worse correlations than high thresholds. We note that minimizing error on tweet proportions is different from maximizing similarity to the CDC data, so it is not *a priori* obvious that the correlations would follow a similar pattern as error. The fact that they do have a similar pattern suggests that this type of analysis of classification error can be of additional use to downstream tasks that use class proportions in an indirect way.

While error-adjusted bootstrapping reduced the error by substantial amounts in the class proportion estimate (Fig. 3),

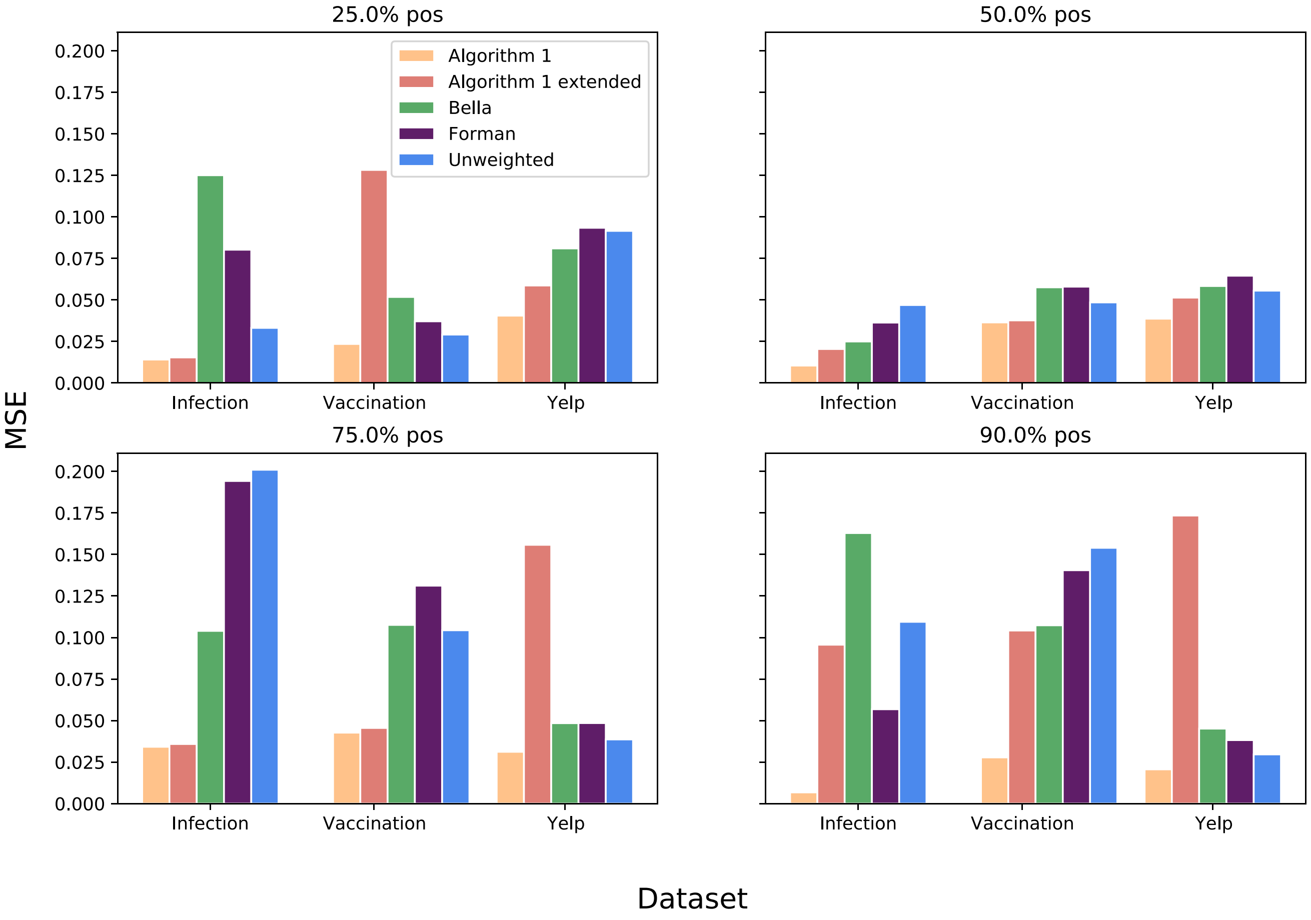


Fig. 6 Class distribution comparison. MSE is shown across all datasets and correction methods. Data are shown at a threshold of 0.5

we do not see comparably large gains in correlations in this task when comparing the adjusted estimates to unadjusted estimates. However, error-adjusted bootstrapping seems to provide a small benefit at some thresholds.

7 Conclusion

We have analyzed, both theoretically and empirically, how classification error propagates to estimates of class proportions, which is often measured incorrectly in practice despite being a common application of classifiers to user-generated data. We found that confidence intervals constructed without accounting for classification error could be surprisingly inaccurate in our experiments (e.g., a 95% interval behaves like a 65% interval), highlighting the need to be careful about using classifiers in a multi-stage pipeline. We showed that a simple-to-implement adjustment to bootstrap sampling can correct for this, and this adjustment can reduce mean squared error when estimating proportions. While we show the adjustment using text-based data, it is trivial to apply it to any classification dataset where classifier metrics can be reasonably estimated from training data. We suggest that the type of analysis presented here can help practitioners trade off error types when tuning classifiers, and that error adjustments should be made when calculating statistics from classifier output.

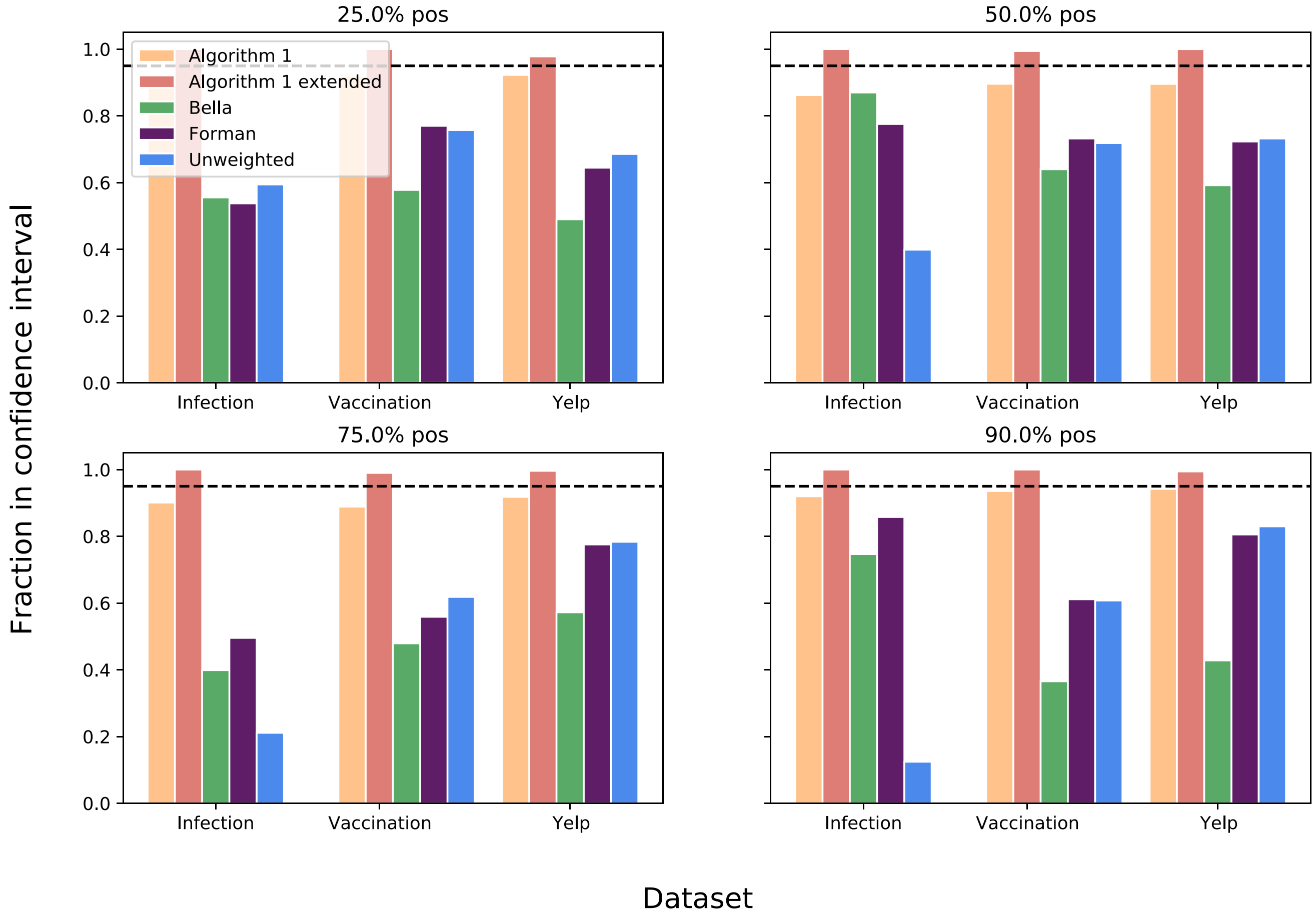


Fig. 7 Class distribution comparison. Fraction included in the confidence interval are shown across all datasets and correction methods. Data are shown at a threshold of 0.5

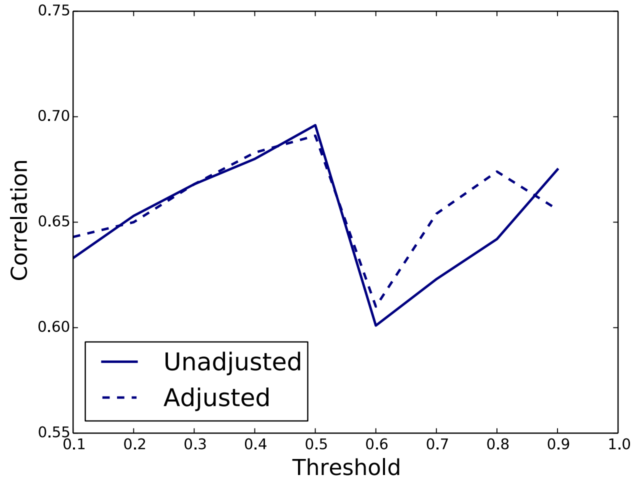


Fig. 8 Correlations between Twitter classifier output and official vaccination data (higher is better)

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s13278-021-00760-0.

Acknowledgements Support from the US Department of Energy through the Los Alamos National Laboratory. Los Alamos National Laboratory is operated by Triad National Security, LLC, for the National Nuclear Security Administration of the US Department of Energy (Contract No. 89233218CNA000001).

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

Ardehaly EM, Culotta A (2017) Learning from noisy label proportions for classifying online social data. Soc Netw Anal Mining 8(1):2. https://doi.org/10.1007/s13278-017-0478-6

Barbiero A, Manzi F, Mecatti G (2015) Bootstrapping probability-proportional-to-size samples via calibrated empirical population. J Stat Comput Simul 85(3):608–620

Barranquero J, González P, Díez J, del Coz JJ (2013) On the study of nearest neighbor algorithms for prevalence estimation in binary problems. Patt Recogn 46(2):472–482. https://doi.org/10.1016/j.patcog.2012.07.022

Bella A, Ferri C, Hernandez-Orallo J, Ramirez-Quintana MJ (2010) Quantification via probability estimators. In *ICDM*. ISBN 978-1-4244-9131-5. https://doi.org/10.1109/ICDM.2010.75

Benton A, Paul MJ, Hancock B, Dredze M (2016) Collective supervision of topic models for predicting surveys with social media. In *AAAI*

Bollen J, Mao H, Pepe A (2011) Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena. In *ICWSM*, URL https://www.aaai.org/ocs/index.php/ICWSM/ICWSM11/paper/view/2826

Buonaccorsi J, Romeo G, Thoresen M (2018) Model-based bootstrapping when correcting for measurement error with application to logistic regression. Biometrics 74(1):135–144

Culotta A (2010) Towards detecting influenza epidemics by analyzing twitter messages. In *KDD Workshop on Social Media Analytics*

Da San Martino G, Gao W, Sebastiani F (2016) Ordinal text quantification. SIGIR. https://doi.org/10.1145/2911451.2914749

Daughton AR, Paul MJ (2019) Constructing accurate confidence intervals when aggregating social media data for public health monitoring. In *AAAI International Workshop on Health Intelligence (W3PHIAI)*, Honolulu, HI

Diakopoulos NA, Shamma DA (2010) Characterizing debate performance via aggregated twitter sentiment. CHI. https://doi.org/10.1145/1753326.1753504

Efron B (1979) Bootstrap methods: Another look at the jackknife. Annal Stat 7(1):1–26. https://doi.org/10.2307/2958830

Efron B, Tibshirani RJ (1993) An Introduction to the Bootstrap. Chapman & Hall, London

Forman G (2005) Counting positives accurately despite inaccurate classification. In *ECML*

Forman G (2006) Tackling concept drift by temporal inductive transfer. In *SIGIR*. ISBN 978-1-59593-369-0. https://doi.org/10.1145/1148170.1148216

Forman G (2008) Quantifying counts and costs via classification. Data Min. Knowl. Discov. 17(2):164–206. ISSN 1384-5810, 1573-756X. https://doi.org/10.1007/s10618-008-0097-y

Gao W, Sebastiani F (2015) Tweet sentiment: From classification to quantification. In *Advances in Social Networks Analysis and Mining (ASONAM)*, ISBN 978-1-4503-3854-7. https://doi.org/10.1145/2808797.2809327

Gao W, Sebastiani F (2016) From classification to quantification in tweet sentiment analysis. *Social Network Analysis and Mining*, 6 (1): 19. ISSN 1869-5469. https://doi.org/10.1007/s13278-016-0327-z

González P, Castaño A, Chawla NV, Coz JJD (2017) A review on quantification learning. *ACM Comput. Surv.*, 50 (5): 74:1–74:40. ISSN 0360-0300. https://doi.org/10.1145/3117807

González P, Castaño A, Chawla NV, Coz JJD (2017) A review on quantification learning. *ACM Computing Surveys*, 50 (5): 1–40. ISSN 03600300. https://doi.org/10.1145/3117807

Huang X, Michael MJP, Smith C, Ryzhkov D, Quinn SC, Broniatowski DA, Dredze M (2017) Examining patterns of influenza vaccination in social media. In *AAAI Joint Workshop on Health Intelligence*

Kück H, de Freitas N (2005) Learning about individuals from group statistics. In *UAI*. URL http://dl.acm.org/citation.cfm?id=3020336.3020378

Lamb A, Paul MJ, Dredze M (2013) Separating fact from fear: Tracking flu infections on twitter. In *NAACL*

Lehmann E, Casella G (1998) Theory of Point Estimation. Springer Verlag. ISBN 0-387-98502-6

Maas AL, Daly RE, Pham PT, Huang D, Ng AY, Potts C (2011) Learning word vectors for sentiment analysis. In *ACL*, URL http://dl.acm.org/citation.cfm?id=2002472.2002491

Milli L, Monreale A, Rossetti G, Giannotti F, Pedreschi D, Sebastiani F (Dec. 2013) Quantification Trees. In *2013 IEEE 13th International Conference on Data Mining*, pages 528–536, Dallas, TX, USA, IEEE. ISBN 978-0-7695-5108-1. https://doi.org/10.1109/ICDM.2013.122

Mitra T, Counts S, Pennebaker J (2016) Understanding anti-vaccination attitudes in social media. In *ICWSM*, URL https://www.aaai.org/ocs/index.php/ICWSM/ICWSM16/paper/view/13073/12747

Musicant D, Christensen J, Olson J (2007) Supervised learning by training on aggregate outputs. In *ICDM*

Nakov P, Ritter A, Rosenthal S, Sebastiani F, Stoyanov V (2016) Semeval-2016 task4: Sentiment analysis in twitter. In *Proceedings of SemEval-2016*, pages 1–18. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/S16-1001

O'Connor B, Balasubramanyan R, Routledge B, Smith N (2010) From tweets to polls: Linking text sentiment to public opinion time series. In *ICWSM*. URL https://www.aaai.org/ocs/index.php/ICWSM/ICWSM10/paper/view/1536/1842

Machine learning in python (2011) F. Pedregosa and others, Scikit-learn. JMLR 12:2825–2830

Pérez-Gállego P, Quevedo JR, del Coz JJ (2017) Using ensembles for problems with characterizable changes in data distribution: A case study on quantification. *Information Fusion*, 34: 87 – 100. ISSN 1566-2535. https://doi.org/10.1016/j.inffus.2016.07.001. URL http://www.sciencedirect.com/science/article/pii/S1566253516300628

Quadrianto N, Smola A, Caetano T, Le Q (2009) Estimating labels from label proportions. JMLR 10:2349–2374

Sadilek A, Kautz H, Silenzio V (2012) Modeling spread of disease from social interactions. In *ICWSM*. URL https://www.aaai.org/ocs/index.php/ICWSM/ICWSM12/paper/view/4493/4999

Sebastiani F (2018) Sentiment quantification of user-generated content. In *ESNAM*

Stram D, Langholz B, Huberman M, Thomas D (1999) Correcting for exposure measurement error in a reanalysis of lung cancer mortality for the Colorado Plateau Uranium Miners cohort. Health Phys 77(3):265–275

Szpiro AA, Paciorek CJ (2014) Measurement error in two-stage analyses, with application to air pollution epidemiology. Environmetrics 24(8):501–517

Wang H, Can D, Kazemzadeh A, Bar F, Narayanan S (2012) A system for real-time twitter sentiment analysis of 2012 U.S. presidential election cycle. In *ACL System Demonstrations*. URL https://doi.org/10.1145/1753326.17535040

West R, Pfeffer J (2017) Armed conflicts in online news: A multilingual study. In *ICWSM*, URL https://doi.org/10.1145/1753326.17535041

Xue JC, Weiss GM (2009) Quantification and semi-supervised classification methods for handling changes in class distribution. In *KDD*

Yin Z, Malin B, Warner J, Hsueh P-Y, Chen C-H (2017) The power of the patient voice: Learning indicators of treatment adherence from an online breast cancer forum. In *ICWSM*. URL https://doi.org/10.1145/1753326.17535042

Yu FX, Liu D, Kumar S, Jebara T, Chang S-F (2013) ∝SVM for learning with label proportions. In *ICML*

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.