`activelearning`: An `R` Package for Optimally Labeling Unlabeled Data for Classification with Active Learning

John A. Ramey

Fred Hutchinson Cancer Research Center, Baylor University

November 21, 2012

Abstract

In this document, I provide an overview of **active learning**, the common approaches, and provide a guide to my implementations of these methods using my recently developed `R` package, `activelearning` .

1 Introduction

Perpetual advancements in technologies have enabled researchers and practitioners to acquire complex data sets to better understand data-generating systems. Unfortunately, the size and complexity of these data can be overwhelming to the point that we have large quantities of information at our fingertips, yet

our typical first approaches to analyzing the data may yield anomalous discoveries and poor results. Numerous statistical and machine learning methods, such as supervised and unsupervised classification, have been developed to aid in the initial exploration of the data. Often, supervised classification is an ideal approach in that based on our current data, we are then able to classify future, unseen observations.

The usual approach is to build a supervised classification decision rule (we use the term *classifier* hereafter) with labeled data in-hand and then classify unlabeled observations as we obtain them. The implicit assumption here is that we know the true classification of each training observation and is often too stringent because obtaining the true labels for each training observation can be costly and time-consuming. Hence, in practice, a small subset of our data may be labeled, while the much larger remainder of our data is unlabeled. In the supervised classification paradigm, we use only the labeled data to construct our classifier and ignore the unlabeled observations, but if we only have a handful of labeled observations, our classification performance on future observations may prove ineffective.

To utilize the large quantity of unlabeled observations, semi-supervised learning (SSL) methods have been developed and are often employed to exploit the inherent structure in the training data (here, we mean the union of the labeled and the unlabeled observations). One common approach is to consider the labels of the unlabeled observations missing at random and then to impute the (missing) labels such that a specified loss or risk function, such as deviance or cross-validation error rate, is minimized. With this approach, the

SSL approach can often be an effective alternative to supervised classification if the number of labeled observations is adequate so that a specified algorithm can determine the structure in the training data. However, the SSL paradigm can be ineffective when ...

Ideally, we would like to obtain the true labels for a sufficient number of training observations from which we construct an efficacious classifier, but as we have discussed previously, the cost and time to procure the labels may hinder any attempts to acquire a large number of them. However, project requirements may allow the researcher to obtain the true labels for a feasible number of observations in an effort to improve understanding of the data as much as possible. If the researcher must achieve a set level of efficacy before the classifier is employed in practice, then how many labels are necessary? Similarly, if the researcher's time and budget constraints allow him to hire an expert to provide the true labels for, say 50, training observations, would obtaining these labels be cost-effective? Furthermore, from the entire set of unlabeled observations in hand, which observations should the researcher have the expert consider?

The researcher could choose the unlabeled observations to consider systematically (that is, he sequentially present observations in the order in which we received the data), randomly. However, these approaches are extremely naive in that they ignore the structure of the data that we already know. Moreover, if there is a region in the data space that we have previous knowledge about, but there are other regions that we know little about (in terms of the labels), then we may consider asking the oracle about the regions in the data space

that little is known about.

A subarea of the machine learning literature, known as **active learning**, is an approach to determine which observations should be labeled and how many are necessary to achieve efficacy. In the active learning literature, the researcher **queries** an **oracle** (previously, the expert) with an unlabeled observation to determine its true label. Also, active learning is closely related to both optimal experimental design and sample-size determination in the statistics literature. As mentioned above, we may randomly choose an observation, or we may wish to choose observations to improve the classification of other unlabeled data or unlabeled data that will be collected in the future. Another approach is to query observations that provide the researcher with maximal understanding of the intrinsic data-generating mechanism.

In this paper we begin with a motivating example in Section 2 based on a large data set with only a few labeled observations. In Section 3 we present several of the proposed methods in the active learning literature to determine which unlabeled observations should be queried. We demonstrate our implementations of these methods in the `activelearning` package, which is available on CRAN (`http://cran.r-project.org/`) in Section 4. Then, in Section 5 we revisit our motivating example with the `activelearning` package and provide concluding remarks in Section 6.

2 A Motivating Example

2.1 We Live in a World with Lots of (Unlabeled) Data

2.2 Random Querying: The Naive Approach

3 Active Learning

3.1 Uncertainty Sampling

The **information density** approach to active learning has been proposed by [Dr. Burr Settles][5] in [his Ph.D. thesis][6]. First, he discusses the **uncertainty sampling** method, where an oracle is queried with the unlabeled observation that is most *uncertain*. The uncertainty is typically defined by a loss or risk function, but there are a few ad hoc approaches to determining uncertainty. Dr. Settles notes that there have been several examples discussed in the literature where the most uncertain observations are the outlying observations. Hence, if we do not have much data from the individual groups, which is the nature of the active learning approach, then the outliers can significantly distort our knowledge of the groups. We define the uncertainty function about an observation $\boldsymbol{x}$ as $\phi(\boldsymbol{x})$, so that we will query the observation

$$\boldsymbol{x}^* = \text{argmax}_x \phi(\boldsymbol{x}) \tag{1}$$

[Culotta and McCallum (2005) have proposed the least confidence function][7]

$$\phi^{LC}(\boldsymbol{x}) = 1 - P_\theta(\widehat{y}|\boldsymbol{x}) \tag{2}$$

where $\widehat{y} = \arg\max_y P_\theta(\widehat{y}|\boldsymbol{x})$ is the observation that maximizes the *a posteriori* probability under model θ. Hence, the least confidence observation that is queried is

$$\boldsymbol{x}^*_{LC} = \text{argmax}_x 1 - P_\theta(\widehat{y}|\boldsymbol{x}) = \text{argmin}_x P_\theta(\widehat{y}|\boldsymbol{x}) \tag{3}$$

which is the observation that has minimizes the *a posteriori* probability under the model θ. Although the least confidence function is simple to interpret, it often leads to the selection of outlying feature vectors, which can degrade future classification performance for classifiers that are not robust to outliers. Even if a classifier is robust outliers, if the classifier is trained on mostly outlying observations, then there will a significant training bias.

Consider, for example, the binary classification problem with two univariate normal distributions. Also, assume that each population has a population variance of 1 and that the means of classes 1 and 2 are 1 and 2 respectively. Now, suppose that we have a training data set with 10 observations, where the majority of the observations are between 0 and 3. For our thought experiment, consider that only two of the observations are labeled: the first observation is 1 and is labeled as class 1, and the second observation is labeled as class 2.

Finally, we assume that we have the unlabeled, outlying observation 7. Clearly, this observation will be queried as it is the least likely of the observations to have occurred from either population or from the marginal distribution of both populations. If the oracle determines that the observation 7 is from the second class, then the sample mean of the second population then will shift from 2 to 4.5. If the oracle determines that the observation 7 is from the first class, the bias induced is even worse. Hence, our knowledge about the underlying distributions can be dramatically shifted with the least confidence function approach to active learning.

Other uncertainty sampling methods include **margin sampling**, which is a correction for the **least confidence** model in the multi-class setting. Perhaps, the most popular proposal for the **uncertainty sampling** function is the Shannon entropy function, so that we query the observation

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}} -\sum_{i=1}^{K} P_\theta(y_i|\boldsymbol{x}) \log P_\theta(y_i|\boldsymbol{x}), \tag{4}$$

where K is the number of populations from which the data was generated. Notice that for binary classification, the Shannon entropy function approach is equivalent to the **least confidence** approach and is also equivalent to margin sampling.

We note that the uncertainty sampling function can be considered as a loss function from a decision theoretic perspective. Hence, we query observations that maximize the specified loss function. This will be important from a Bayesian perspective as we will rely on this feature to extend the information density approach to active learning.

Finally, a more detailed discussion of uncertainty sampling is given in the [active learning overview from Settles][1].

3.2 Query by Committee

3.3 Query by Bagging

3.4 Expected Model Change

3.5 Expected Error Reduction

3.6 Variance Reduction

3.7 Density-Weighted Methods

3.8 Others

4 Using `activelearning`

5 Revisiting the Motivating Example with `activelearning`

6 Concluding Remarks

We have demonstrated the `R` package `activelearning` and how it may be used in the active learning framework.

We have considered many effective approaches to query an oracle with unlabeled data and have discussed the future classification performance that active learning may yield. However, as we have also discussed, active learning should not be used as a black-box approach to labeling data because no currently proposed method is able to account for all idiosyncrasies in the data.

Practically speaking, we may present an oracle with numerous features in the form of an image in the context of facial recognition, but in other domains, a graphical summary of an individual observation is often difficult to construct and to interpret (e.g. a graph of a linear combination of an observation's features). Also, the presentation of *every* feature of an observation may be unintelligible to the eye of a human oracle. An approach is to present a subset of an observation's features, but caution must be used if the queried label is applied to the entire observation.

Additionally, the methods that we have considered and implemented have been largely developed for the purpose of text classification, where typically we assume that the feature vectors are probabilistically independent. Although this assumption is widely used in all areas of statistics and machine learning, the approach does not take into account if the data are correlated, especially when the data consists of sequential observations. The active learning methods that we have discussed do not explicitly account for or attempt to model autocorrelations in the data, although some active learning approaches may do this implicitly with ad-hoc reasoning.

The application of active learning to time series models should not be ignored because it is plausible that there are events occurring over time and that

an expert may wish to determine if that event is interesting. For example, consider city-wide or nation-wide blackouts. If we query the oracle (ask the expert) regarding a set of features collected a specific point in time, the oracle may suggest that the moment in time is not interesting but that another moment in time is atypical or is behaving strangely. The collection of categories for time-dependent events may further help a user in understanding their system as we begin to determine automatically events that are unique or peculiar.

Additionally, it may not be advisable to query an instance in time to categorize it because if a user is presented with a set of events that occurred at, say 9:47:23 PM PST, the oracle may have no grasp of the exact event at the specific time. But if the oracle was presented instead with a range of times, say 9:30 PM to 10:00 PM and a summary of the events over that time period, then the user may be able to provide detailed information. This also may be useful if a couple of observations within the time window are atypical; the user may be able to label specific observations within the window, especially because they may differ greatly with the other observations. This approach may be feasible with [multiple instance active learning][3], which essentially approaches the active learning problem hierarchically so that a number of observations may be labeled (the labeling might be considered soft in that it is subject to change) if they are similar enough to the specific instance that was queried or a summary of a set of similar observations.