Maximum Entropy Density Estimation with Incomplete Presence-Only Data

Bert Huang
Computer Science Department
Columbia University
New York, NY 10027
bert@cs.columbia.edu

Ansaf Salleb-Aouissi
Center for Computational Learning Systems
Columbia University
New York, NY 10115
ansaf@ccls.columbia.edu

Abstract

We demonstrate a generalization of Maximum Entropy Density Estimation that elegantly handles incomplete presence-only data. We provide a formulation that is able to learn from known values of incomplete data without having to learn imputed values, which may be inaccurate. This saves the effort needed to perform accurate imputation while observing the principle of maximum entropy throughout the learning process. We provide analysis and examples of our algorithm under different settings of missing data.

1 INTRODUCTION

In this work we demonstrate a generalization of Maximum Entropy Density Estimation that handles incomplete data without having to perform imputation. Most machine learning work requires rectangular data matrices with fully observable values for all entries. When applying machine learning to the real world, we expect to have incomplete features. Often, data collection machinery, whether human-operated or automated, simply cannot practically fill in every value. It is therefore desirable to find ways to apply machine learning to incomplete data. Furthermore, most existing methods of handling incomplete data attempt to recover the values of the missing features. In our method, we avoid the extra work and possible cascading errors involved in estimating the missing features and directly address the main goal of density estimation.

Appearing in Proceedings of the 12^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2009, Clearwater Beach, Florida, USA. Volume 5 of JMLR: W&CP 5.

1.1 PREVIOUS WORK

There is a large history of analysis of missing data in classical statistics literature. Missing data has been classified as one of three types (Little & Rubin, 1986). Data is either missing completely at random (MCAR), missing at random (MAR) or not missing at random (NMAR). When data is MCAR, whether a data sample is missing any features is truly random, and is independent of the values of any of the sample's other features. When data is MAR, whether a sample is missing a feature is dependent on one or more of the sample's other observable features. When data is NMAR, whether the data is missing a feature is dependent on the value of features that may be missing (possibly including the missing feature itself).

Statistical analysis of incomplete data typically requires one of two options: ignore the data points with any missing values or attempt to estimate the missing values. Both of these approaches can be problematic. Throwing out incomplete data is a drastic measure, because we are unable to make predictions about those data nor are we able to learn from the features that we do know. Estimating the missing values, or imputation, causes the learning algorithm to learn from the estimated values. This is reasonable if we expect to make good estimates of the missing values; the upper bound outcome of imputation is that we perfectly reconstruct the unknown values. The lower bound, however, is unclear, and largely depends on the imputation method and the actual data.

Of the multitude of imputation methods, perhaps the simplest is mean imputation. Mean imputation replaces missing values with the mean of the observed values. One of the more sophisticated methods is the Gaussian Expectation-Maximization (EM) imputation, which iterates between regressing based on a Gaussian model of the data and fitting a new Gaussian to the estimated data (Ghahramani & Jordan, 1994; Schneider, 2001) until convergence to a local optimum.

In practice, data often exhibit Gaussian-like properties, so this method is rather effective. Another effective method stemming from classical statistical analysis is multiple imputation (Little & Rubin, 1986), in which data is imputed multiple times to create multiple versions of the entire data set. From this, better confidence bounds can be obtained than with single imputation.

The algorithm we are extending is one based on the Maximum Entropy Principle, which says to choose the probability distribution with the most uncertainty, or with the maximum entropy subject to what is known. The natural language processing community used the maxent algorithm without regularization (Berger et al., 1996), but the regularized maximum entropy approach for learning with presence-only data was made popular in (Phillips et al., 2004). This work featured the usage of maxent for predicting the habitats of animal species. The regularization functions have since been generalized (Dudík & Schapire, 2006). In (Phillips et al., 2006; Phillips et al., 2004), the authors had to omit many coastal and island areas from their study due to those localities' missing data from one or more features. Specifically, one of the species had twelve (of 128) observed sightings at these missing locations. They lost a significant amount of labeled information due to the limitations induced by incomplete data.

As an aside, the labels in the presence-only framework can also be considered missing data. The training labels are the observable values of the feature "class", and every other example is missing that feature. Considering the problem in this setting it becomes necessary to require that the missing "class feature" is MCAR for maxent's assumptions to hold. This is because maxent requires that the sampled data is drawn i.i.d. to justify that the expected value of the features will be close to the empirical means. If the samples being missing is dependent on the features themselves, the empirical means of the labeled examples will be different from the true expectations. This problem is addressed in (Dudík et al., 2006) where it is described as sample selection bias. Our work does not attempt to solve this problem and assumes the labeled data is i.i.d.

Maximum margin classification of incomplete data following similar principles to ours was studied in (Chechik et al., 2008). The authors derive a formulation for separable linear max-margin classification where the unknown dimensions of the data points are dropped from the hyperplane weights. They globally optimize the separable case but the non-separable case becomes non-convex.

1.2 SUMMARY

In Section 2, we first review the presence-only framework and maximum entropy as a viable algorithm to learn from presence-only data. Then we introduce our extension to handle missing data, and derive an optimization algorithm. In Section 3, we discuss the behavior of maxent with our extensions and compare to imputation. In Section 4, we report the results of some synthetic and six real data experiments. Finally we conclude with a brief discussion in Section 5.

2 FRAMEWORK AND ALGORITHM DERIVATION

In the presence-only data framework, we are given a sample space χ, a finite set of data points. From this sample space, we are given a set of positive labeled points $\{x_1, \ldots, x_m\}$. We assume these labeled points are drawn i.i.d. from some unknown target distribution π over χ. To estimate this target distribution we have a set of features $\{f_j : \chi \to \mathbb{R}\}$ for $j \in \{1, \ldots, n\}$ (Phillips et al., 2004)[1].

The maxent solution to the presence only problem entails defining an empirical distribution $\tilde{\pi}$, which puts $\frac{1}{m}$ probability over all labeled points and zero probability on all other points in χ. While this empirical distribution $\tilde{\pi}$ is likely to be different from the true distribution π, the expectations of the features over $\tilde{\pi}$ should be close to those over π. We adopt the following notation for expectations: $\pi[f_j] = E_\pi[f_j] = \sum_{x \in \chi} \pi(x) f_j(x)$, where $f_j(x)$ is the value of feature j for example x.

The empirical means grow exponentially close to the true expectations, so the maxent algorithm only considers distributions whose expectations are close to the empirical means. Of these distributions, the maximum entropy principle suggests that we choose the distribution that fits the constraints with highest entropy: closest to uniform. Given ℓ_1 regularization parameters $\{\beta_1, \ldots, \beta_n\}$, (Dudík et al., 2004; Phillips et al., 2004)

$$\max_p \quad -\sum_{x \in \chi} p(x) \ln p(x) \tag{1}$$

$$s.t. \quad \begin{cases} \sum_{x \in \chi} p(x) = 1 \\ |p[f_j] - \tilde{\pi}[f_j]| \leq \beta_j, \ \forall j \in \{1, \ldots, n\} \end{cases}$$

So far, we have reviewed previous work on maxent. In the next section, we describe our extension.

[1] The labeled samples can also be considered points in the positive class of a classification problem in which the learning algorithm is not given the labels of the negative class.

2.1 INCOMPLETE DATA

In addition to the standard information given for maxent in the presence-only framework, we are given certain features for which we do not know the values. We indicate missing values with

$$o_j(x) = \begin{cases} 1 & \text{if } f_j(x) \text{ is known} \\ 0 & \text{if } f_j(x) \text{ is missing.} \end{cases} \quad (2)$$

Without loss of generality, we set all unknown values to $f_j(x) = 0$. Then we can compute empirical means and expectations with incomplete data with

$$\tilde{\pi}[f_j]' = \left(\sum_{i=1}^{m} \tilde{\pi}(x_i) f_j(x_i)\right) / \left(\sum_{i=1}^{m} \tilde{\pi}(x_i) o_j(x_i)\right) \quad \text{(empirical mean)} \quad (3)$$

$$\pi[f_j]' = \left(\sum_{x\in\chi} \pi(x) f_j(x)\right) / \left(\sum_{x\in\chi} \pi(x) o_j(x)\right) \quad \text{(true expectation)} \quad (4)$$

$$p[f_j]' = \left(\sum_{x\in\chi} p(x) f_j(x)\right) / \left(\sum_{x\in\chi} p(x) o_j(x)\right) \quad \text{(estimated expectation)} \quad (5)$$

Note that if all feature values are known, the formulas above are equivalent to the actual expectations. We can rewrite the constraints of the objective using the incomplete data formulation.

$$\max_p \quad -\sum_{x\in\chi} p(x) \ln p(x) \quad (6)$$

$$s.t. \quad \begin{cases} \sum_{x\in\chi} p(x) = 1 \\ |p[f_j]' - \tilde{\pi}[f_j]'| \le \beta_j, \ \forall j \in \{1, \ldots, n\} \end{cases}$$

2.2 DUAL FORM

The expectation constraints can be written to remove the fractions. The original constraint is as follows:

$$\begin{aligned} |p[f_j]' - \tilde{\pi}[f_j]'| &\le \beta_j \\ \left| \frac{\sum_{x\in\chi} p(x) f_j(x)}{\sum_{x\in\chi} p(x) o_j(x)} - \tilde{\pi}[f_j]' \right| &\le \beta_j \end{aligned}$$

We multiply both sides by the denominator.

$$\left| \sum_{x\in\chi} p(x) f_j(x) - \tilde{\pi}[f_j]' \sum_{x\in\chi} p(x) o_j(x) \right| \le \beta_j \sum_{x\in\chi} p(x) o_j(x) \quad (7)$$

Since entropy is convex and our proposed constraints are still linear, we solve the convex program using Lagrange multipliers as in (Phillips et al., 2004). Equation (6) corresponds to the following Lagrangian:

$$\begin{aligned} L(\lambda^+, \lambda^-, \lambda_0, p) = & \\ & -\textstyle\sum_{x\in\chi} p(x) \ln p(x) - \lambda_0 \left(\sum_{x\in\chi} p(x) - 1\right) \\ & + \textstyle\sum_j \left(\lambda_j^+ - \lambda_j^-\right) \left(\sum_{x\in\chi} p(x) f_j(x) - \tilde{\pi}[f_j]' \sum_{x\in\chi} p(x) o_j(x)\right) \\ & + \textstyle\sum_j \left(\lambda_j^+ + \lambda_j^-\right) \beta_j \sum_{x\in\chi} p(x) o_j(x). \end{aligned} \quad (8)$$

Only one of the constraints corresponding to λ_j^+ and λ_j^- will be active in the solution because the estimated expectation cannot be both at the upper bound *and* the lower bound, we can assume only λ_j^+ or λ_j^-, but not both, must be nonzero. We introduce a change of variable $\lambda_j = \lambda_j^+ - \lambda_j^-$, then $\lambda_j^+ + \lambda_j^- = |\lambda_j|$. We can then simplify the above Lagrangian (Dudík et al., 2004).

$$\begin{aligned} L(p, \lambda) = & -\textstyle\sum_{x\in\chi} p(x) \ln p(x) - \lambda_0 \left(\sum_{x\in\chi} p(x) - 1\right) + \\ & \textstyle\sum_j \lambda_j \left(\sum_{x\in\chi} p(x) f_j(x) - \tilde{\pi}[f_j]' \sum_{x\in\chi} p(x) o_j(x)\right) \\ & + \textstyle\sum_j |\lambda_j| \beta_j \sum_{x\in\chi} p(x) o_j(x) \end{aligned} \quad (9)$$

This Lagrangian is concave with respect to p. We can simply take the derivative and solve for $p(x)$ analytically. Then the optimal $p(x)$ is

$$\hat{p}(x) = e^{\sum_j \left[\lambda_j \left(f_j(x) - \tilde{\pi}[f_j]' o_j(x)\right) + |\lambda_j| \beta_j o_j(x)\right] - \lambda_0 - 1}. \quad (10)$$

Plugging this solution back into L, we get

$$\begin{aligned} L(p, \lambda) = & \quad (11) \\ & \lambda_0 + e^{-\lambda_0 - 1} \sum_{x\in\chi} e^{\sum_j \lambda_j \left(f_j(x) - \tilde{\pi}[f_j]' o_j(x)\right) + |\lambda_j| \beta_j o_j(x)} \end{aligned}$$

Solving the derivative with respect to λ_0 gives the optimal setting of λ_0:

$$\hat{\lambda}_0 = \ln \left(\sum_{x\in\chi} e^{\sum_j \lambda_j \left(f_j(x) - \tilde{\pi}[f_j]' o_j(x)\right) + |\lambda_j| \beta_j o_j(x)} \right) - 1.$$

In other words, λ_0 corresponds to the log partition function, $\lambda_0 = \ln Z - 1$, and the optimal $\hat{p}$ given λ is the following Gibbs-like distribution:

$$\hat{p}(x) = \frac{1}{Z} e^{\sum_j \left[\lambda_j \left(f_j(x) - \tilde{\pi}[f_k]' o_j(x)\right) + |\lambda_j| \beta_j o_j(x)\right]}. \quad (12)$$

Plugging the new λ_0 back into the objective gives us

$$L(\lambda) = \lambda_0 + \frac{Z}{Z} = \ln Z$$

Finally, observing the monotonicity of the natural log, the unconstrained optimization is

$$\min_\lambda Z = \sum_{x\in\chi} \prod_j e^{\lambda_j \left(f_j(x) - \tilde{\pi}[f_j]' o_j(x)\right) + |\lambda_j| \beta_j o_j(x)}.$$

This unconstrained objective is convex; it is a sum of convex functions when the β_j's are non-negative (which is true in all sensible settings). To see that the inner function is convex, we can write the absolute value as a maximum of two inner functions who touch at their minima, $\lambda_j = 0$:

$$\prod_j e^{\lambda_j\left(f_j(x)-\tilde{\pi}[f_j]'o_j(x)\right)+\max(\lambda_j\beta_j o_j(x),-\lambda_j\beta_j o_j(x))} \quad (13)$$

Algorithm 1 Dual Maxent for Incomplete Data. The inputs to the procedure are sample means $\tilde{\pi}[f_j]$, regularization parameters β_j, features $f_j(x)$ and missingness indicators $o_j(x)$, for $j \in \{1,\ldots,n\}$, $x \in \chi$.

$\lambda_j \leftarrow 0, \forall j$
repeat
  for $j = 0$ to D **do**
    $\lambda_j^+ \leftarrow \lambda_j^+ - \frac{\delta Z^+}{\delta\lambda_j}/\frac{\delta^2 Z^+}{\delta\lambda_j^2}$
    $\lambda_j^- \leftarrow \lambda_j^- - \frac{\delta Z^-}{\delta\lambda_j}/\frac{\delta^2 Z^-}{\delta\lambda_j^2}$ {See Eq. (14), (15)}
    if $\lambda_j^+ > 0$ **then**
      $\lambda_j \leftarrow \lambda_j^+$
    else if $\lambda_j^- < 0$ **then**
      $\lambda_j \leftarrow \lambda_j^-$
    else
      $\lambda_j \leftarrow 0$
    end if
  end for
until Convergence
$\hat{p}(x) = \frac{1}{Z}e^{\sum_j\left[\lambda_j\left(f_j(x)-\tilde{\pi}[f_k]'o_j(x)\right)+|\lambda_j|\beta_j o_j(x)\right]}, \forall i$

2.3 OPTIMIZATION

Since we have an unconstrained convex program, there are a multitude of methods to optimize the λ's. We choose to greedily minimize along one dimension at a time using a Newton optimization. The first and second derivatives with respect to one dimension at a time are

$$\begin{aligned}\frac{\delta Z^\pm}{\delta\lambda_j} &= \sum_{x\in\chi}\left(f_j(x)-\tilde{\pi}[f_j]'o_j(x)\pm\beta_j o_j(x)\right) \quad (14)\\ &\quad\times\prod_k e^{\left(\lambda_k\left(f_k(x)-\tilde{\pi}[f_k]'o_k(x)\right)+|\lambda_k|\beta_k o_k(x)\right)}\\ \frac{\delta^2 Z^\pm}{\delta\lambda_j^2} &= \sum_{x\in\chi}\left(f_j(x)-\tilde{\pi}[f_j]'o_j(x)\pm\beta_j o_j(x)\right)^2 \quad (15)\\ &\quad\times\prod_k e^{\left(\lambda_k\left(f_k(x)-\tilde{\pi}[f_k]'o_k(x)\right)+|\lambda_k|\beta_k o_k(x)\right)}\end{aligned}$$

To account for the absolute value, we try both signs of the above derivatives. The correct optimal λ_j will be the same sign as the $\pm$ in the derivatives. If both optima are the wrong sign, the function is optimal at 0 and we set $\lambda_j = 0$. The algorithm is summarized in Algorithm 1.

$$\lambda_j = \lambda_j - \frac{\delta Z^\pm}{\delta\lambda_j}/\frac{\delta^2 Z^\pm}{\delta\lambda_j^2} \quad (16)$$

3 EFFECT OF MISSINGNESS ON MEANS AND EXPECTATIONS

In the NMAR missingness setting, the true expectation of the data can be quite distant from the observable expectation. However, since the labels are i.i.d., the missingness affects the observable training and testing expectations the same way. In this section we explore this idea more formally.

Using the shorthand $f(x) = \{f_1(x),\ldots,f_n(x)\}$, we have a prior $p(o_j(x))$ and a conditional distribution of missingness $p(o_j(x)|f(x))$ (which can be MCAR, MAR or NMAR) then by Bayes Rule

$$p(f(x)|o_j(x)) = \frac{p(o_j(x)|f(x))p(f(x))}{p(o_j(x))}. \quad (17)$$

When considering any single feature we can absorb the probability of observing the feature into the probability of that sample being selected[2].

$$\begin{aligned}p[f_j]' &= \left(\sum_{x\in\chi}p(x)f_j(x)\right)/\left(\sum_{x\in\chi}p(x)o_j(x)\right)\\ &= \left(\sum_{x\in\chi}p(x|o_j(x))f_j(x)\right)/\left(\sum_{x\in\chi}p(x|o_j(x))\right)\\ &= \sum_{x\in\chi}p(x|o_j(x)=1)f_j(x) = E_{p(x|o_j(x)=1)}[f_j]\end{aligned}$$

In other words, the expectations on which we impose constraints are no longer over $p(x)$, but instead over the conditional probability $p(x|o_j(x))$. Compare to what happens with imputation. Let $g_j(x|\chi,(f_1,o_1),\ldots,(f_n,o_n))$, which we write as $g_j(x)$ for brevity, represent any imputation method invoked when $f_j(x)$ is missing. That is, if $f_j(x)$ is missing, it is replaced by $g_j(x)$.

$$\begin{aligned}p[f_j]'' &= \sum_{\substack{x\in\chi\\ o_j(x)=1}}p(x)f_j(x) + \sum_{\substack{x\in\chi\\ o_j(x)=0}}p(x)g_j(x)\\ &= \frac{E_{p(x|o_j(x)=1)}[f_j]}{\sum_x o_j(x)} + \frac{E_{p(x|o_j(x)=0)}[g_j]}{\sum_x(1-o_j(x))} \quad (18)\end{aligned}$$

Expectations after imputation can therefore be written as the weighted sum of two expectations, one of which

[2]Note that because we assume our label distribution to be dependent on the features, $p(f(x))$ and $p(x)$ are practically equivalent and therefore interchangeable.

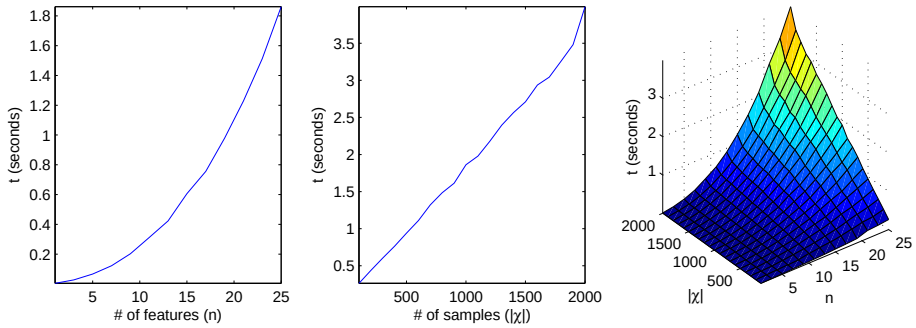

Figure 1: Mean running time over 100 random trials. Left: Running time with respect to number of features (n). Middle: Running time w.r.t. size of sample space $|\chi|$. Right: Running time w.r.t. both n and $|\chi|$.

	Missing	Mean (all)	Mean (positive)	EM
MCAR	-261.96 ± 3.04	-262.02 ± 2.99	-262.07 ± 3.05	-262.04 ± 3.05
MAR	-258.58 ± 3.90	-258.70 ± 3.88	-258.75 ± 4.01	-258.63 ± 3.86
NMAR	-258.79 ± 4.02	-259.05 ± 4.01	-258.88 ± 4.22	-259.04 ± 4.00
Full	-254.99 ± 3.30			

Table 1: Average log-likelihoods of estimated density over synthetic data. The average log-likelihood of maxent on the full data was -254.99 ± 3.30. "Missing", which is our method, scores the highest average likelihood.

is over the conditional distribution as in our formulation. The second, however, is the expectation over another conditional distribution of some new feature g_j, which, may perfectly reconstruct f_j, thus restoring the original expectation, or may be completely arbitrary.

In both cases, the missingness of the data forces us to learn using the expectations of the actual features over some altered distribution, but with imputation, we may also be perturbing our view of the data with the imputed features.

4 EXPERIMENTS

In this section, we present the procedures and results from a few experiments designed to help understand the behavior of our method. In Section 4.1, we test the empirical running time of our optimization routine. In Section 4.2, we compare the out-of-sample likelihood performance of our estimate to popular imputation methods on synthetic data sampled according to each of the three missingness regimes. In Section 4.3, we compare performance between our method and the cases of accurate imputation versus inaccurate imputation. Finally, in Section 4.4, we compare performance of our method to imputation on real data with real missingess patterns.

4.1 RUNNING TIME

We sampled random test data by drawing data points from two random Gaussian distributions and labeling only a subset of one Gaussian as present. To test the running time of our algorithm, we sampled such data sets of various sizes $|\chi|$ and feature dimensionalities n. We ran our optimization until the dual variables changed less than 10^{-12} total. Running the optimization beyond this produced almost no change in the solution. We ran these tests using MATLAB on a 2.4 Ghz Intel Core 2 Duo Apple Macintosh running Mac OS X 10.5. The results are plotted in Figure 1.

There are many black-box non-linear optimization methods, including standard packages that implement conjugate gradient descent or Quasi-Newton optimization, to solve our objective. However, extra care would be necessary to properly implement the absolute values. We report our running time to demonstrate that our optimization method is effective for practical usage.

4.2 MISSINGNESS TESTS

In this experiment we created synthetic presence-only data following the assumptions from the framework. We created 5000 synthetic data sets by drawing 200 data points uniformly from the $[0,1]^{10}$. For each data set, we randomly drew ten λ values from a 0-mean normal distribution of variance 1 and a random mean $\mu \in [0,1]^{10}$ computed a Gibbs distribution over the data using $p(x) = \frac{1}{Z}\exp(\sum_j \lambda_j(f_j - \mu_j))$. We then drew 100 samples from $p(x)$, labeled half of those positive and left the remaining 50 as testing points.

Next we created missing versions of these data sets for

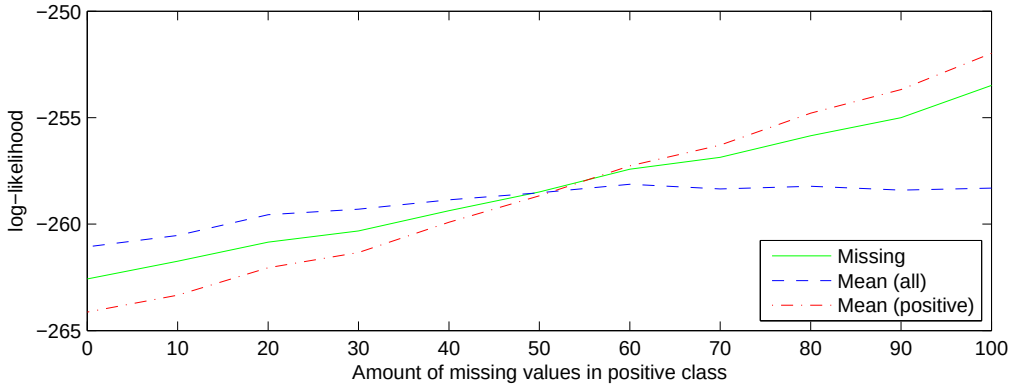

Figure 2: Average log-likelihood of our proposed method versus maxent with mean imputation as we vary the number of points with missing values in the positive class. "Missing" is our method.

each of the three types of missingness. To synthesize MCAR data, we randomly hid half of the entries in the feature matrix. To synthesize MAR data, we selected a random number of features to have missing values, then sampled yet another Gibbs distribution over the remaining features. We then drew points from that distribution and hid features from those points. Finally, to synthesize NMAR data, we sampled our Gibbs parameters for the features we were going to hide. Note that for the more interesting settings, MAR and NMAR, the fraction of missing features varies between our sampled data sets.

We compare our method ("Missing") to three standard imputation strategies: two variants of mean imputation and Gaussian EM imputation. For mean imputation, we fill in the missing values with either the means of the whole data set ("Mean (all)") or the means of only the labeled points ("Mean (positive)"). For EM imputation we use the author's code from (Schneider, 2001). On each data set, we run four-fold cross-validation to find the regularization term and evaluate the highest scoring model on the out-of-sample test points. We choose the regularization terms by sweeping through a single normalized parameter β and setting the individual tolerances for each feature according to

$$\beta_j = \beta \frac{\text{std}(\{F_{ij}; O_{ij} > 0\})}{\sqrt{\sum y_i O_{ij}}}.$$

The results are listed in Table 1.

Even with such a large number of random data sets, the differences in performance are small. In general, it is difficult to predict which method will be the most accurate since one of the imputation methods can always be fortunate enough to estimate fairly accurate feature values or the imputation could be completely wrong, allowing our method's agnosticism about the missing features to be the dominant strategy.

4.3 QUALITY OF IMPUTATION

To demonstrate the effect of quality of imputation on performance, we compare the two mean imputation methods to ours. If we impute to the mean of the whole sample space, we treat every data point equally during the imputation step; we ignore the labels. If we impute to the mean of the labeled samples, we give the data points with missing features the benefit of the doubt. This means the points with missing values will pull the expected value toward the empirical mean. In addition, $\tilde{\pi}[f_j]$ using labeled-mean imputation is equal to $\tilde{\pi}[f_j]$ using our method. Therefore, it is tempting to expect labeled-mean imputation to behave much like our method.

In fact, the difference between our method and imputing to the labeled mean is subtle but important. Equations (3) and (12) using labeled-mean imputed features are exactly equal to our method, but the expectation over the full sample space, Equation (5) and subsequent formulas derived using Equation (5) are different. Points missing values will add constraints that prefer high probability over those points with missing values. One example scenario in which this is undesirable is when few missing values occur in true high probability points.

We created 500 data sets using the same sampling method as in Section 4.2. For each data set, we hid values randomly on 100 of the 200 samples, varying the ratio of the number of positive samples that have missing values. Since we always had 100 data points with missing values, if k points were true positives and contained missing values, $100 - k$ non-positive points had missing features.

The log-likelihoods resulting from this experiment are shown in Figure 2. When the positive class has more missing values, mean imputation over the whole sam-

	Missing	Mean (all)	Mean (positive)	EM	$p(o)$	$p(y\|o)$
bands	**-711.1±2.9**	-711.5 ± 2.8	**-710.8±2.7**	-711.5 ± 2.8	0.92	0.87
crx	**-991.0±3.2**	**-990.9±3.2**	**-991.0±3.2**	**-990.8±3.3**	0.99	0.42
echo	**-57.2±1.1**	-57.3 ± 1.1	**-57.1±1.0**	-57.4 ± 1.0	0.92	0.32
hep	**-74.8±2.6**	-75.4 ± 2.5	-75.2 ± 2.3	**-75.1±2.5**	0.90	0.27
horse-colic	**-299.3±2.9**	-300.1 ± 2.9	-304.7 ± 2.2	**-299.6±2.7**	0.66	0.38
house-votes	**-555.1±2.8**	**-555.1±2.8**	**-555.1±2.5**	**-555.0±2.8**	0.91	0.42

Table 2: Log-likelihoods of out-of-sample positive data. Best performance and those not significantly worse via a two-sample t-test with a rejection threshold 5% in bold. Far right columns give a simple picture of missingness in data sets. $p(o)$ represents the proportion of data not missing, while $p(\text{pos.}|o)$ represents the proportion of points with missing values in the positive class. "Missing" is our method.

ple space scores the lowest likelihood while imputation to the labeled mean scores the highest likelihood. Conversely, when the missing values are mostly in the non-positive points, labeled-mean imputation scores the lowest likelihood while sample-space-mean imputation scores the highest. Our method, as any proper agnostic algorithm should, scores safely in the middle of these two high-energy imputation methods.

4.4 INCOMPLETE REAL DATA

To test the behavior of our method on real missing data, we downloaded some of the popular data sets from the UCI Machine Learning Repository (D. Newman & Merz, 1998) that contain missing values. These are classification data sets that we cast as presence-only by hiding the class with smaller cardinality and treating the larger class as the only "present" class. Unfortunately most popular data sets are mostly complete due to the fact that missing data is difficult to handle. To exacerbate the natural missingness of these data sets, we ignore the most complete half of the feature columns. The missingness in these experiments is therefore real and occurs in varying proportions.

On 500 random splits of 50% training and 50% testing, we evaluated the same four methods as the synthetic case above, measuring the log-likelihood on the out-of-sample test points given the model that scored the highest on four-fold cross-validation. In general, our method was competitive with the imputation methods. More importantly, comparing to the best performing method, ours is never significantly worse according to a two sample t-test with a rejection threshold of 5%.The average log-likelihoods and some basic statistics on the missingness of each set are listed in Table 2.

5 DISCUSSION

We have proposed a natural generalization of maxent for incomplete presence-only data. Our formulation learns from all the features that are observable without having to learn from imputed features. We keep track of the missing values and use constraints based only on the observable features. The resulting algorithm follows the principle of maximum entropy throughout the learning process, while standard imputation methods deviate from the principle when the imputed means are assumed to be true.

Many problems cast as classification make sense in the presence-only framework: for example, the many data sets using "death" as a class label (such as the hepatitis data set). Since everything eventually dies, it is reasonable to consider samples labeled "death" to be samples from an underlying density instead of considering all samples, dead or alive, to be drawn from two distinct classes.

We derived this paper's extension to maxent out of necessity. We work with proprietary data where our goal is to predict faults in system components (analogous to "death") and our data features are very incomplete. Much of our missing data is due to limitations in sensor technology and convenience of the human effort necessary to take measurements. This implies a strong change of NMAR missingness. It was therefore unlikely that a simple MCAR method such as mean imputation could produce accurate results. Conversely, sophisticated methods that may perform better in the NMAR setting were impractical because our data set was so large. We expect other machine learning practitioners have similar experiences, and these data sets never become popular public benchmarks because their incompleteness (or their presence-only nature) make them seem infeasible for machine learning. New methods for elegantly handling incomplete data may attract attention to these data sets.

One possible future direction can exploit the fact that typical maxent applications employ various expanded features. For features generated by a function over individual original features, such as threshold features, our method translates directly. However, for features generated by combining two dimensions, such as quadratic features, it may be possible to precisely

model the range of possible values of the induced features when a subset of the original features are known.

Acknowledgments

This work has been partly supported by a research contract from Consolidated Edison New York.

References

Berger, A., Pietra, S., & Pietra, V. (1996). A maximum entropy approach to natural language processing. *Computational Linguistics*, *22*, 39–71.

Chechik, G., Heitz, G., Elidan, G., Abbeel, P., & Koller, D. (2008). Max-margin classification of incomplete data. *Journal of Machine Learning Research*, *9*, 1–21.

D. Newman, S. Hettich, C. B., & Merz, C. (1998). UCI repository of machine learning databases.

Dudík, M., Phillips, S., & Schapire, R. (2004). Performance guarantees for regularized maximum entropy density estimation. *Learning Theory, 17th Annual Conference on Learning Theory, COLT 2004, Banff, Canada, July 1-4, 2004, Proceedings* (pp. 472–486).

Dudík, M., Phillips, S., & Schapire, R. (2006). Correcting sample selection bias in maximum entropy density estimation. In Y. Weiss, B. Schölkopf and J. Platt (Eds.), *Advances in neural information processing systems 18*, 323–330. Cambridge, MA: MIT Press.

Dudík, M., & Schapire, R. (2006). Maximum entropy distribution estimation with generalized regularization. *Learning Theory, 19th Annual Conference on Learning Theory, COLT 2006, Pittsburgh, PA, USA, June 22-25, 2006, Proceedings* (pp. 123–138).

Ghahramani, Z., & Jordan, M. (1994). Supervised learning from incomplete data via an EM approach. *Advances in Neural Information Processing Systems* (pp. 120–127). Morgan Kaufmann Publishers, Inc.

Little, R., & Rubin, D. (1986). *Statistical analysis with missing data*. New York, NY, USA: John Wiley & Sons, Inc.

Phillips, S., Anderson, R., & Schapire, R. (2006). Maximum entropy modeling of species geographic distributions. *Ecological Modelling*, *190*, 231–259.

Phillips, S., Dudík, M., & Schapire, R. (2004). A maximum entropy approach to species distribution modeling. *Machine Learning, Proceedings of the Twenty-first International Conference (ICML 2004), Banff, Alberta, Canada, July 4-8, 2004*.

Schneider, T. (2001). Analysis of incomplete climate data: Estimation of mean values and covariance matrices and imputation of missing values. *Journal of Climate*, *14*, 853871.