Robust Logistic Regression using Shift Parameters

Julie Tibshirani and **Christopher D. Manning**
Stanford University
Stanford, CA 94305, USA
{jtibs, manning}@cs.stanford.edu

Abstract

Annotation errors can significantly hurt classifier performance, yet datasets are only growing noisier with the increased use of Amazon Mechanical Turk and techniques like distant supervision that automatically generate labels. In this paper, we present a robust extension of logistic regression that incorporates the possibility of mislabelling directly into the objective. This model can be trained through nearly the same means as logistic regression, and retains its efficiency on high-dimensional datasets. We conduct experiments on named entity recognition data and find that our approach can provide a significant improvement over the standard model when annotation errors are present.

1 Introduction

Almost any large dataset has annotation errors, especially those complex, nuanced datasets commonly used in natural language processing. Low-quality annotations have become even more common in recent years with the rise of Amazon Mechanical Turk, as well as methods like distant supervision and co-training that involve automatically generating training data.

Although small amounts of noise may not be detrimental, in some applications the level can be high: upon manually inspecting a relation extraction corpus commonly used in distant supervision, Riedel et al. (2010) report a 31% false positive rate. In cases like these, annotation errors have frequently been observed to hurt performance. Dingare et al. (2005), for example, conduct error analysis on a system to extract relations from biomedical text, and observe that over half of the system's errors could be attributed to inconsistencies in how the data was annotated. Similarly, in a case study on co-training for natural language tasks, Pierce and Cardie (2001) find that the degradation in data quality from automatic labelling prevents these systems from performing comparably to their fully-supervised counterparts.

In this work we argue that incorrect examples should be explicitly modelled during training, and present a simple extension of logistic regression that incorporates the possibility of mislabelling directly into the objective. Following a technique from robust statistics, our model introduces sparse 'shift parameters' to allow datapoints to slide along the sigmoid, changing class if appropriate. It has a convex objective, is well-suited to high-dimensional data, and can be efficiently trained with minimal changes to the logistic regression pipeline.

In experiments on a large, noisy NER dataset, we find that this method can provide an improvement over standard logistic regression when annotation errors are present. The model also provides a means to identify which examples were mislabelled: through experiments on biological data, we demonstrate how our method can be used to accurately identify annotation errors. This robust extension of logistic regression shows particular promise for NLP applications: it helps account for incorrect labels, while remaining efficient on large, high-dimensional datasets.

2 Related Work

Much of the previous work on dealing with annotation errors centers around filtering the data before training. Brodley and Friedl (1999) introduce what is perhaps the simplest form of supervised filtering: they train various classifiers, then record their predictions on a different part of the train set and eliminate contentious examples. Sculley and Cormack (2008) apply this approach to spam filtering with noisy user feedback.

One obvious issue with these methods is that the noise-detecting classifiers are themselves trained

on noisy labels. Unsupervised filtering tries to avoid this problem by clustering training instances based solely on their features, then using the clusters to detect labelling anomalies (Rebbapragada et al., 2009). Recently, Intxaurrondo et al. (2013) applied this approach to distantly-supervised relation extraction, using heuristics such as the number of mentions per tuple to eliminate suspicious examples.

Unsupervised filtering, however, relies on the perhaps unwarranted assumption that examples with the same label lie close together in feature space. Moreover filtering techniques in general may not be well-justified: if a training example does not fit closely with the current model, it is not necessarily mislabelled. It may represent an important exception that would improve the overall fit, or appear unusual simply because we have made poor modelling assumptions.

Perhaps the most promising approaches are those that directly model annotation errors, handling mislabelled examples as they train. This way, there is an active trade-off between fitting the model and identifying suspected errors. Bootkrajang and Kaban (2012) present an extension of logistic regression that models annotation errors through flipping probabilities. While intuitive, this approach has shortcomings of its own: the objective function is nonconvex and the authors note that local optima are an issue, and the model can be difficult to fit when there are many more features than training examples.

There is a growing body of literature on learning from several annotators, each of whom may be inaccurate (Bachrach et al., 2012; Raykar et al., 2009). It is important to note that we are considering a separate, and perhaps more general, problem: we have only one source of noisy labels, and the errors need not come from the human annotators, but could be introduced through contamination or automatic labelling.

The field of 'robust statistics' seeks to develop estimators that are not unduly affected by deviations from the model assumptions (Huber and Ronchetti, 2009). Since mislabelled points are one type of outlier, this goal is naturally related to our interest in dealing with noisy data, and it seems many of the existing techniques would be relevant. A common strategy is to use a modified loss function that gives less influence to points far from the boundary, and several models along these lines have been proposed (Ding and Vishwanathan., 2010; Masnadi-Shirazi et al., 2010). Unfortunately these approaches require optimizing nonstandard, often nonconvex objectives, and fail to give insight into which datapoints are mislabelled.

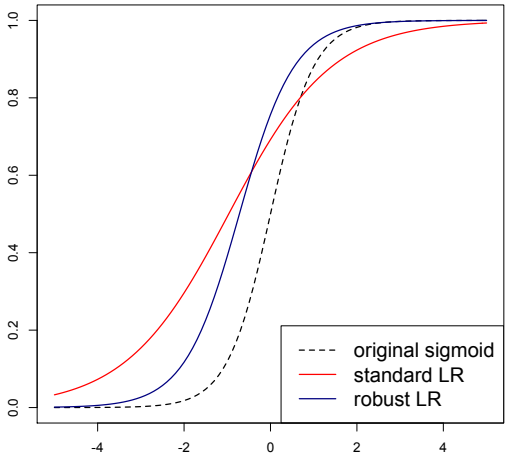

Figure 1: Fit resulting from a standard vs. robust model, where data is generated from the dashed sigmoid and negative labels flipped with probability 0.2.

In a recent advance, She and Owen (2011) demonstrate that introducing a regularized 'shift parameter' per datapoint can help increase the robustness of linear regression. Candes et al. (2009) propose a similar approach for principal component analysis, while Wright and Ma (2009) explore its effectiveness in sparse signal recovery. In this work we adapt the technique to logistic regression. To the best of our knowledge, we are the first to experiment with adding 'shift parameters' to logistic regression and demonstrate that the model is especially well-suited to the type of high-dimensional, noisy datasets commonly used in NLP.

3 Model

Recall that in binary logistic regression, the probability of an example x_i being positive is modeled as

$$g(\theta^T x_i) = \frac{1}{1 + e^{-\theta^T x_i}}.$$

For simplicity, we assume the intercept term has been folded into the weight vector θ, so $\theta \in \mathbb{R}^{m+1}$ where m is the number of features.

Following She and Owen (2011), we propose the following robust extension: for each datapoint $i = 1, \ldots, n$, we introduce a real-valued shift pa-

rameter γ_i so that the sigmoid becomes

$$g(\theta^T x_i + \gamma_i) = \frac{1}{1 + e^{-\theta^T x_i - \gamma_i}}.$$

Since we believe that most examples are correctly labelled, we L_1-regularize the shift parameters to encourage sparsity. Letting $y_i \in \{0, 1\}$ be the label for datapoint i and fixing $\lambda \geq 0$, our objective is now given by

$$l(\theta, \gamma) = \sum_{i=1}^{n} \Big[y_i \log g(\theta^T x_i + \gamma_i) + (1 - y_i) \log \big(1 - g(\theta^T x_i + \gamma_i)\big) \Big] - \lambda \sum_{i=1}^{n} |\gamma_i|. \quad (1)$$

These parameters γ_i let certain datapoints shift along the sigmoid, perhaps switching from one class to the other. If a datapoint i is correctly annotated, then we would expect its corresponding γ_i to be zero. If it actually belongs to the positive class but is labelled negative, then γ_i might be positive, and analogously for the other direction.

One way to interpret the model is that it allows the log-odds of select datapoints to be shifted. Compared to models based on label-flipping, where there is a global set of flipping probabilities, our method has the advantage of targeting each example individually.

It is worth noting that there is no difficulty in regularizing the θ parameters as well. For example, if we choose to use an L_1 penalty then our objective becomes

$$l(\theta, \gamma) = \sum_{i=1}^{n} \Big[y_i \log g(\theta^T x_i + \gamma_i) + (1 - y_i) \log \big(1 - g(\theta^T x_i + \gamma_i)\big) \Big] - \kappa \sum_{j=1}^{m} |\theta_j| - \lambda \sum_{i=1}^{n} |\gamma_i|. \quad (2)$$

Finally, it may seem concerning that we have introduced a new parameter for each datapoint. But in many applications the number of features already exceeds n, so with proper regularization, this increase is actually quite reasonable.

3.1 Training

Notice that adding these shift parameters is equivalent to introducing n features, where the ith new feature is 1 for datapoint i and 0 otherwise. With this observation, we can simply modify the feature matrix and parameter vector and train the logistic model as usual. Specifically, we let $\theta' = (\theta_0, \ldots, \theta_m, \gamma_1, \ldots, \gamma_n)$ and $X' = [X|I_n]$ so that the objective (1) simplifies to

$$l(\theta') = \sum_{i=1}^{n} \Big[y_i \log g(\theta'^T x'_i) + (1 - y_i) \log \big(1 - g(\theta'^T x'_i)\big) \Big] - \lambda \sum_{j=m+1}^{m+n} |\theta'^{(j)}|.$$

Upon writing the objective in this way, we immediately see that it is convex, just as standard L_1-penalized logistic regression is convex.

3.2 Testing

To obtain our final logistic model, we keep only the θ parameters. Predictions are then made as usual:

$$\mathbf{I}\{g(\hat{\theta}^T x) > 0.5\}.$$

3.3 Selecting Regularization Parameters

The parameter λ from equation (1) would normally be chosen through cross-validation, but our set-up is unusual in that the training set may contain errors, and even if we have a designated development set it is unlikely to be error-free. We found in simulations that the errors largely do not interfere in selecting λ, so in the experiments below we cross-validate as normal.

Notice that λ has a direct effect on the number of nonzero shifts γ and hence the suspected number of errors in the training set. So if we have information about the noise level, we can directly incorporate it into the selection procedure. For example, we may believe the training set has no more than 15% noise, and so would restrict the choice of λ during cross-validation to only those values where 15% or fewer of the estimated shift parameters are nonzero.

We now consider situations in which the θ parameters are regularized as well. Assume, for example, that we use L_1-regularization as in equation (2), so that we now need to optimize over both κ and λ. We perform the following simple procedure:

1. Cross-validate using standard logistic regression to select κ.
2. Fix this value for κ, and cross-validate using the robust model to find the best choice of λ.

method	suspects identified									false positives
	T2	T30	T33	T36	T37	N8	N12	N34	N36	
Alon et al. (1999)		•	•	•		•		•	•	
Furey et al. (2000)	•				•	•		•	•	T6, N2
Kadota et al. (2003)	•	•	•	•			•	•	•	T8, N2, N28, N29
Malossini et al. (2006)	•	•	•	•			•	•	•	
Bootkrajang et al. (2012)		•	•	•		•	•	•	•	
Robust LR										

Table 1: Results of various error-identification methods on the colon cancer dataset. The first row lists the samples that are biologically confirmed to be suspicious, and each other row gives the output from an automatic detection method. Bootkrajang et al. report confidences, so we threshold at 0.5 to obtain these results.

4 Experiments

We conduct two sets of experiments to assess the effectiveness of the approach, in terms of both identifying mislabelled examples and producing accurate predictions.

4.1 Contaminated Data

Our first experiment is centered around a biological dataset with suspected labelling errors. Called the colon cancer dataset, it contains the expression levels of 2000 genes from 40 tumor and 22 normal tissues (Alon et al., 1999). There is evidence in the literature that certain tissue samples may have been cross-contaminated. In particular, 5 tumor and 4 normal samples should have their labels flipped.

In this experiment, we examine the model's ability to identify mislabelled training examples. Because there are many more features than datapoints and it is likely that not all genes are relevant, we choose to place an L_1 penalty on θ.

Using `glmnet`, an R package for training regularized models (Friedman et al., 2009), we select κ and λ using cross-validation. Looking at the resulting values for γ, we find that only 7 of the shift parameters are nonzero and that each one corresponds to a suspicious datapoint. As further confirmation, the signs of the gammas correctly match the direction of the mislabelling. Compared to previous attempts to automatically detect errors in this dataset, our approach identifies at least as many suspicious examples but with no false positives. A detailed comparison is given in Table 1. Although Bootkrajang and Kaban (2012) are quite accurate, it is worth noting that due to its nonconvexity, their model needed to be trained 20 times to achieve these results.

4.2 Manually Annotated Data

We now consider the problem of *named entity recognition* (NER) to evaluate how our model performs in a large-scale prediction task. In traditional NER, the goal is to determine whether each word is a person, organization, location, or not a named entity ('other'). Since our model is binary, we concentrate on the task of deciding whether a word is a person or not. (This task does not trivially reduce to finding the capitalized words, as the model must distinguish between people and other named entities like organizations).

For training, we use a large, noisy NER dataset collected by Jenny Finkel. The data was created by taking various Wikipedia articles and giving them to five Amazon Mechanical Turkers to annotate. Few to no quality controls were put in place, so that certain annotators produced very noisy labels. To construct the train set we chose a Turker who was about average in how much he disagreed with the majority vote, and used only his annotations. Negative examples are subsampled to bring the class ratio to a reasonable level, for a total of 200,000 negative and 24,002 positive examples. We find that in 0.4% of examples, the majority agreed they were negative but the chosen annotator marked them positive, and 7.5% were labelled positive by the majority but negative by the annotator. Note that we still include examples for which there was no majority consensus, so these noise estimates are quite conservative.

We evaluate on the English development test set from the CoNLL shared task (Tjong Kim Sang and Meulder, 2003). This data consists of news articles from the Reuters corpus, hand-annotated by researchers at the University of Antwerp.

We extract a set of features using Stanford's NER pipeline (Finkel et al., 2005). This set was

model	precision	recall	F1
standard	76.99	85.87	81.19
flipping	76.62	86.28	81.17
robust	**77.04**	**90.47**	**83.22**

Table 2: Performance of standard vs. robust logistic regression in the Wikipedia NER experiment. The flipping model refers to the approach from Bootkrajang and Kaban (2012).

chosen for simplicity and is not highly engineered – it largely consists of lexical features such as the current word, the previous and next words in the sentence, as well as character n-grams and various word shape features. With a total of 393,633 features in the train set, we choose to use L_2-regularization, so that our penalty now becomes

$$\frac{1}{2\sigma^2}\sum_{j=0}^{m}|\theta_j|^2 + \lambda\sum_{i=1}^{n}|\gamma_i|.$$

This choice is natural as L_2 is the most common form of regularization in NLP, and we wish to verify that our approach works for penalties besides L_1.

The robust model is fit using Orthant-Wise Limited-Memory Quasi Newton (OWL-QN), a technique for optimizing an L_1-penalized objective (Andrew and Gao, 2007). We tune both models through 5-fold cross-validation to obtain $\sigma^2 = 1.0$ and $\lambda = 0.1$. Note that from the way we cross-validate (first tuning σ using standard logistic regression, fixing this choice, then tuning λ) our procedure may give an unfair advantage to the baseline.

We also compare against the algorithm proposed in Bootkrajang and Kaban (2012), an extension of logistic regression mentioned in the section on prior work. This approach assumes that each example's true label is flipped with a certain probability before being observed, and fits the resulting latent-variable model using EM.

The results of these experiments are shown in Table 2 as well as Figure 2. Robust logistic regression offers a noticeable improvement over the baseline, and this improvement holds at essentially all levels of precision and recall. Interestingly, because of the large dimension, the flipping model consistently learns that no labels have been flipped and thus does not show a substantial difference with standard logistic regression.

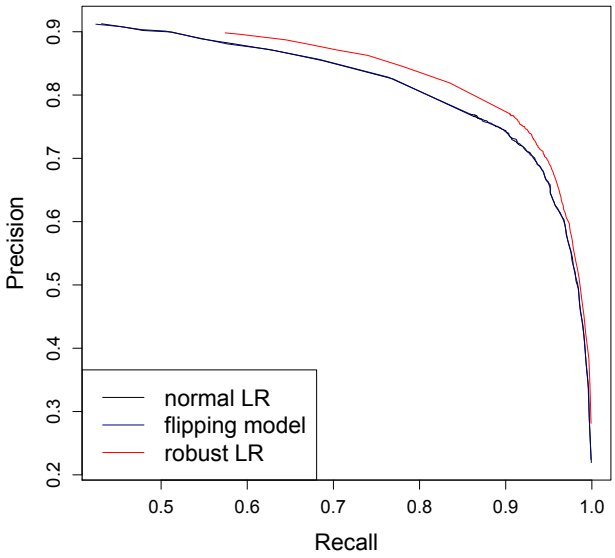

Figure 2: Precision-recall curve obtained from training on noisy Wikipedia data and testing on CoNLL. The flipping model refers to the approach from Bootkrajang and Kaban (2012).

5 Future Work

A natural direction for future work is to extend the model to a multi-class setting. One option is to introduce a γ for every class except the negative one, so that there are $n(c-1)$ shift parameters in all. We could then apply a group lasso, with each group consisting of the γ for a particular datapoint (Meier et al., 2008). This way all of a datapoint's shift parameters drop out together, which corresponds to the example being correctly labelled.

CRFs and other sequence models could also benefit from the addition of shift parameters. Since the extra variables can be neatly folded into the linear term, convexity is preserved and the model could essentially be trained as usual.

Acknowledgments

Stanford University gratefully acknowledges the support of the Defense Advanced Research Projects Agency (DARPA) Deep Exploration and Filtering of Text (DEFT) Program under Air Force Research Laboratory (AFRL) contract no. FA8750-13-2-0040. Any opinions, findings, and conclusion or recommendations expressed in this material are those of the authors and do not necessarily reflect the view of the DARPA, AFRL, or the US government. We are especially grateful to Rob Tibshirani and Stefan Wager for their invaluable advice and encouragement.

References

U. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, A. J. Levine. 1999. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *National Academy of Sciences of the USA.*

Galen Andrew and Jianfeng Gao. 2007. Scalable Training of L_1-Regularized Log-Linear Models. *ICML.*

Yoram Bachrach, Thore Graepel, Tom Minka, and John Guiver. 2012. How To Grade a Test Without Knowing the Answers: A Bayesian Graphical Model for Adaptive Crowdsourcing and Aptitude Testing. *arXiv preprint arXiv:1206.6386 (2012).*

Jakramate Bootkrajang and Ata Kaban. 2012. Label-noise Robust Logistic Regression and Its Applications. *ECML PKDD.*

Carla E. Brodley and Mark A. Friedl. 1999. Identifying mislabeled Training Data. *JAIR,* 11, 131-167.

Emmanuel J. Candes, Xiaodong Li, Yi Ma, John Wright. 2009. Robust Principal Component Analysis? *arXiv preprint arXiv:0912.3599, 2009.*

Nan Ding and S. V. N. Vishwanathan. 2010. t-Logistic regression. *NIPS.*

Shipra Dingare, Malvina Nissim, Jenny Finkel, Christopher Manning, and Claire Grover. 2005. A system for identifying named entities in biomedical text: How results from two evaluations reflect on both the system and the evaluations. *Comparative and Functional Genomics.* 6(1–2), 77-85.

Jenny Rose Finkel, Trond Grenager, Christopher Manning. 2005. Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. *ACL.*

Jerome Friedman, Trevor Hastie, Rob Tibshirani 2009. Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of statistical software,* 33(1), 1.

Terrence S. Furey, Nello Cristianini, Nigel Duffy, David W. Bednarski, Michel Schummer, David Haussler. 2000. Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics,* 16(10), 906-914.

Peter J. Huber and Elvezio M. Ronchetti. 2000. *Robust Statistics.* John Wiley & Sons, Inc., Hoboken, NJ.

Ander Intxaurrondo, Mihai Surdeanu, Oier Lopez de Lacalle, and Eneko Agirre. 2013. Removing Noisy Mentions for Distant Supervision. *Congreso de la Sociedad Espaola para el Procesamiento del Lenguaje Natural.*

Koji Kadota, Daisuke Tominaga, Yutaka Akiyama, Katsutoshi Takahashi. 2003. Detecting outlying samples in microarray data: A critical assessment of the effect of outliers on sample. *ChemBio Informatics Journal,* 3(1), 30-45.

Andrea Malossini, Enrico Blanzieri, Raymond T. Ng. 2006. Detecting potential labeling errors in microarrays by data perturbation. *Bioinformatics,* 22(17), 2114-2121.

Hamed Masnadi-Shirazi, Vijay Mahadevan, and Nuno Vasconcelos. 2010. On the design of robust classifiers for computer vision. *IEEE International Conference Computer Vision and Pattern Recognition.*

Lukas Meier, Sara van de Geer, Peter Buhlmann. 2008. The group lasso for logistic regression. *Journal of the Royal Statistical Society,* 70(1), 53-71.

David Pierce and Claire Cardie. 2001. Limitations of co-training for natural language learning from large datasets. *EMNLP.*

Vikas Raykar, Shipeng Yu, Linda H. Zhao, Anna Jerebko, Charles Florin, Gerardo Hermosillo Valadez, Luca Bogoni, and Linda Moy. 2009. Supervised learning from multiple experts: whom to trust when everyone lies a bit. *ICML.*

Umaa Rebbapragada, Lukas Mandrake, Kiri L. Wagstaff, Damhnait Gleeson, Rebecca Castano, Steve Chien, Carla E. Brodley 2009. Improving Onboard Analysis of Hyperion Images by Filtering mislabelled Training Data Examples. *IEEE Aerospace Conference.*

Sebastian Riedel, Limin Yao, Andrew McCallum. 2010. Modeling Relations and Their Mentions without Labelled Text. *ECML PKDD.*

D. Sculley and Gordon V. Cormack 2008. Filtering Email Spam in the Presence of Noisy User Feedback. *CEAS.*

Yiyuan She and Art Owen. 2011. Outlier Detection Using Nonconvex Penalized Regression. *Journal of the American Statistical Association,* 106(494).

Erik F. Tjong Kim Sang, Fien De Meulder. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. *CoNLL.*

John Wright and Yi Ma. 2009. Dense Error Correction via l_1-Minimization *IEEE Transactions on Information Theory.*