Analysis of Discriminative and Generative Graphical Models on Image Classification

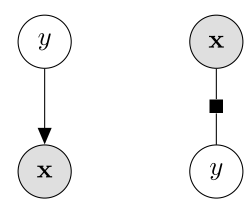
William Watson
Johns Hopkins University
billwatson@jhu.edu

Abstract

In statistical classification, there are two main approaches to learning: *generative* and *discriminative*. I outline a procedure to explore the performance of corresponding probabilistic graphical models on a simple image classification task.

1 Introduction

For a classification task, I are interested in the probability distribution of a class label y conditioned on the sample features $\mathbf{x}$, written as $p(y \mid \mathbf{x})$. However, I can approach modeling this distribution in two ways: *generative* or *discriminative*.



Generative Discriminative

1.1 Generative Approach

A generative model is a statistical process to describe the observed data $\mathbf{x}$. More specifically, generative classifiers, as mentioned by Ng and Jordan (2002) [8], seek to model the joint probability $p(\mathbf{x}, y)$ of the inputs $\mathbf{x}$ and the labels y. I can then make predictions with Bayes rule as follows:

$$\hat{y} = \operatorname*{argmax}_y p(\mathbf{x}, y) = \operatorname*{argmax}_y p(y \mid \mathbf{x})\, p(\mathbf{x}) = \operatorname*{argmax}_y p(\mathbf{x} \mid y)\, p(y) \tag{1}$$

1.2 Discriminative Approach

In contrast, a discriminative model is a statistical process to describe relations between the observed data $\mathbf{x}$ and some latent variables $\mathbf{h}$. For discriminative classifiers, our latent variables are class labels y, and I can model the posterior distribution $p(y \mid \mathbf{x})$ directly [8]. Hence our prediction becomes:

$$\hat{y} = \operatorname*{argmax}_y p(y \mid \mathbf{x}) \tag{2}$$

2 Data

I plan to use the Rijkmuseum Dataset [5] to perform artist classification. However, to make parameter estimation and inference tractable, I will constrain the classification problem to fewer artists and downsample the images.

Machine Learning: Data to Models (CS 676 Spring 2019): Final Project Proposal

3 Models

In order to facilitate comparisons between the generative and discriminative approach to classification, I will explore both the static and general versions as outlined in Figure 2.

3.1 Baseline

Our baseline model will consist of a *Naive Bayes* classifier, a simple generative Bayesian network that assumes all the features $x_i \in \mathbf{x}$ are independent from one another and the sole parent is the class label y. This assumption simplifies the factorization for the joint distribution.

The discriminative analog to Naive Bayes is *Logistic Regression*, which is also a naive classifier, yet models the conditional probability directly.

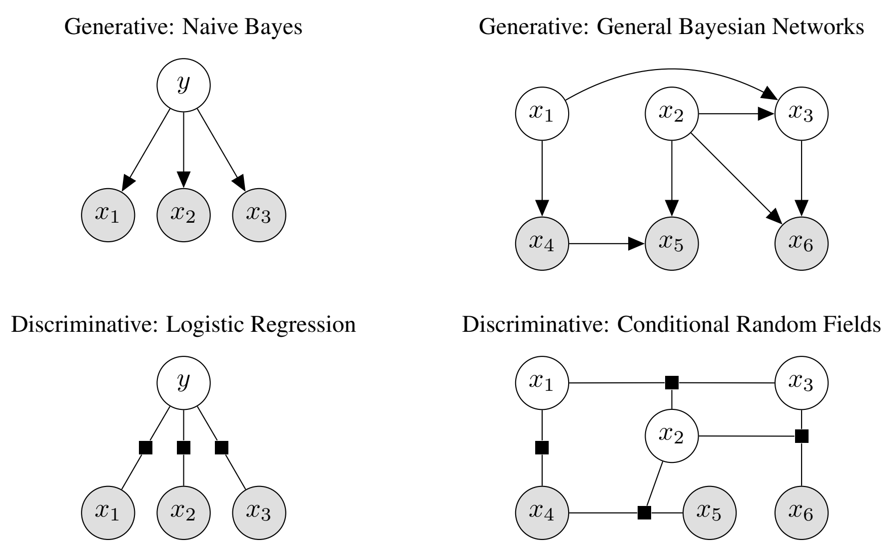
3.2 General

The general counterparts to our baseline models are the generative *General Bayesian Network* and its discriminative analog *Conditional Random Fields (CRFs)*.

General Bayesian Networks are generative probabilistic models that encode the conditional dependencies of its variables via a directed acyclic graph. Each variable is dependent only on its parents, allowing for more complex interactions between our features.

CRFs are a discriminative undirected probabilistic graphical model whose nodes can be partitioned into two disjoint sets: $\mathbf{x}$ and $\mathbf{y}$, and models the conditional distribution $p(\mathbf{y} \mid \mathbf{x})$ directly.

Figure 2: Generative vs. Discriminative Models



4 Parameter Estimation Task

For the generative models, I will use Maximum Likelihood Estimation to learn the parameters of the model.

Logistic Regression will use the L-BFGS optimization algorithm, while General CRFs will use a cutting plane approach to optimize a quadratic programming problem, as outline by Müller et. al (2014) [6] and Memisevic (2006) [4].

5 Inference Task

For inference in predicting the correct class label y with our generative models, I will use belief propagation, if possible. Otherwise I will use Markov Chain Monte Carlo (MCMC) sampling methods with NUTS [3] to approximate our posteriors.

For our discriminative models, CRFs will use the suite of inference algorithms provided by OpenGM [1], including Loopy Belief Propagation or Gibbs sampling. It is important to note that for CRFs, just like Markov Random Fields, the problem of exact inference can be intractable, hence the usage of approximation algorithms. Logistic Regression models $p(y \mid x)$, and hence inference can be seen as a sampling technique.

6 Software

This project will use OpenGM [1], PyStruct [6], pgmpy[1], and scitkit-learn [7] to create, learning, and perform inference tasks with our models. OpenCV [2] will be used for image preprocessing and downsampling.

References

[1] B. Andres, T. Beier, and J. Kappes. OpenGM: A C++ library for discrete graphical models. *CoRR*, abs/1206.0111, 2012.

[2] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[3] M. D. Hoffman and A. Gelman. The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo. *arXiv e-prints*, page arXiv:1111.4246, Nov 2011.

[4] R. Memisevic. An introduction to structured discriminative learning. Technical report, 2006.

[5] T. E. J. Mensink and J. C. van Gemert. The rijksmuseum challenge: Museum-centered visual recognition. In *ACM International Conference on Multimedia Retrieval*, 2014.

[6] A. C. Müller and S. Behnke. pystruct - learning structured prediction in python. *Journal of Machine Learning Research*, 15:2055–2060, 2014.

[7] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[8] A. Y. Ng and M. Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. *Adv. Neural Inf. Process. Sys*, 2, 04 2002.

[1] http://pgmpy.org