Active Model Selection

Dylan Cashman

1 Introduction

The first step in data analysis, whether it be clustering, regression, or forecasting, is to choose a model that best represents the underlying data. The space of potential models, which encompasses not just different types of models (kNN, SVM, ANNs), but all their potential parameterizations, is intractably large. Tools like the Automatic Statistician, built by researchers at Cambridge, offer some hope by sampling the model space on a training set, and providing some metrics to help the subject matter expert (SME) choose one of the sampled models. Making this choice often requires critical insight into the formulation of those models; a decision tree with no pruning will inevitably overfit and thus underperform on new test data.

Our goal is to allow SMEs to incorporate the same insight that a statistical expert would use to choose a model. Instead of expecting them to understand the intricacies that vary between models and parameterizations, we rely on their domain knowledge. Our method will generate a new data point that has high disagreement among the models that the SME is asked to choose from. The SME then determines a label for that point, and our system will penalize those models that the SME disagreed with. The idea is inspired by the field of Active Learning, in which a human user is asked to label data that the model can learn the most from. However, Active Learning is used to aid training of generally a single model, where we are using it to choose between multiple models after training has occurred. In addition, we are generating new data for the user to label, instead of feeding the user already existing data points. This should help resolve overfitting concerns.

To generate data points, we sample from regions where similar points exist in the training set that are disagreed on. However, we follow the principles of [2] to give the user the most interpretable example possible. In particular, we favor a minimum amount of significant digits, and we minimize the number of variables the user is asked to interpret.

To demonstrate the usefulness of this technique, we will build a system that guides the user through a certain number of these point selections. We will use this system to conduct a between-subject experiment in which the control group will get the interface with no active model selection, and be asked to choose the best model. The experimental group will use the active model selection to close out or rerank models. We'll see which group chooses a better model.

2 One-sentence description

To help Subject Matter Experts choose the optimal model for their data, we generate interpretable data points that different models disagree on, helping them choose the model that best fits their mental model.

3 Project Type

Application

4 Audience

> *Who is the audience for this project? How does it meet their needs? What happens if their needs remain unmet?*

This project would benefit the generally underserved majority of subject matter experts that use data analysis in their day-to-day work. Data analysis is a ubiquitous element of research, and methods are only getting more complicated and further distant from the generalist's statistical education, especially as more universities open data science schools and the audience for data-focused research dramatically expands. Our system will provide evidence for the tactic of active model selection, which would then get incorporated into the regular data analysis pipeline.

If no solution is provided for subject matter experts to choose between models, either of two things will happen. It could be that subject matter experts continue to use the statistical methods they are comfortable with, leaving them with suboptimal models and restricting the audience of the more sophisticated models that are being actively developed in the statistics research communities. The alternative may be worse; subject matter experts could be pressured into using models they neither trust nor understand how to tune or interpret, which would results in incorrect statistical conclusions.

5 Approach

5.1 Details

What is your approach?

Without loss of generality, we assume we are working with a binary classification problem. Our system is given a set of models $\mathcal{M}$, a training set $\mathcal{X} \subset \mathcal{D}$ where $\mathcal{D}$ is the domain of the classification problem. For every example $x_i \in \mathcal{X}$, we have a true label y_i, and a label $\hat{y}_i^j$ for each of the models $m_j \in \mathcal{M}$.

Active Model Selection can be formally described as generating an optimal point, x^*, such that

$$x^* = \arg\max_x H(\{m_i(x)\})$$

where $x \in \mathcal{D} \setminus \mathcal{X}$, $m_i \in \mathcal{M}$, and H is the shannon entropy of the set of labels given by the different models. For the simplified binary classification problem, we can write out H explicitly. Let $p(x)$ be the percent of models in $\mathcal{M}$ that label x as a true label. Then we can define x^* as follows.

$$x^* = \arg\max_x -p(x)\log p(x) - (1-p(x))\log(1-p(x))$$

x^* will be the point in the domain that is most disagreed about. For a single model, this might be a point on the decision boundaries. For an unspecified number of models, this will be a point that there is much disagreement about.

However, we won't be able to sample the entire domain $\mathcal{D}$. Instead, we hope to be able to develop an estimate function, $\tilde{H} \sim H$, that is defined over all of $\mathcal{D}$, that is differentiable so that we can find extrema. This is a core problem for this work. We'd like to use the points that we already have labeled to define this estimate function. One way to do this is with Gaussian Processes.

5.1.1 Gaussian Processes

A Gaussian Process is a random variable defined over an entire domain. It is an abstraction of a one-dimensional Gaussian distribution to possibly infinite dimensions of a domain, where each element of a domain is a separate dimension. While it sounds intractable, it is actually fairly simple to work with. Just as, in a one-dimensional Gaussian distribution, drawing a certain point can help you determine what your mean and variance are (or how weighted a coin is), drawing the value of a random variable at a certain point in a gaussian process helps you determine the mean and variance of that specific point but also all other points in the domain. The rate at which learning one point updates all other points in a domain is dictated by the prior, which is a parameter of the model.

It may be illustrative to consider the application to our problem. Suppose we had the Wine dataset, and we had N models all trying to classify as white vs. red. Before we have the labelings of any data points, we have no idea whether a new data point should be classified as white or red. Suppose we draw a point from the training set, and it is classified as 75% white, and 25% red. That tells us that the points near that will probably have similar entropy. Depending on the covariance matrix we choose for our Gaussian Process, we should result in a convex estimate function $\tilde{H}$. Choosing the right prior will be important and potentially difficult. It may be problem specific. Exploring prior selection will be very interesting.

Ideally, we would be able to enforce constraints - we may not want to maximize $\tilde{H}$, but rather $\tilde{H}$ constrained to some subset of $\mathcal{D}$, so that we are only choosing practical and interpretable new points in $\mathcal{D}$. This should be possible with basic convex optimizers, though.

5.2 Evidence for Success

Why do you think it will work?

The basic idea of prototyping data points is already used in statistics for understanding models. The theory is sound in that we should be able to generate interesting data points.

6 Best-case Impact Statement

In the best-case scenario, what would be the impact statement (conclusion statement) for this project?

I think this is a really big idea. Choosing between models is very important and a widespread problem. This is the first real foray into it that I've been able to find in the literature.

7 Major Milestones

- Coding out a prototype in Python, without a GUI
- Coding up a GUI (browser-based)
- Running a pilot experiment with Wine dataset
- Choosing experimental variables - different priors, whether we use Explainers-type constraints

8 Obstacles

- Choosing a Gaussian Process prior may prove difficult - for this to be useful it should be automated
- This may be computationally intensive - Gaussian Process inference involves taking a matrix inverse, where the matrix is the size of the number of points considered. We should be able to just sample the space intelligently, though. Maybe through a k-d tree?
- Our function may end up pointing us to points that are extremely close to points we already know about - maybe we can force it to be more interesting via a prior or with constraints?

8.1 Major obstacles

It might just not generate interesting points.

8.2 Minor obstacles

The coding might turn out to be a bit heavy.

9 Resources Needed

What additional resources do you need to complete this project?

I'll need someone to help with the GUI. I think I need to find the right dataset to do a user test for this. It may be hard deciding which types of models go into the pilot.

10 5 Related Publications

- Mike Gleicher's Explainers paper[2] suggests that we will get value out of picking points that the users can interpret.
- This work is inspired by the Automatic Statistician[1], but I believe it continues on that work in helping the user choose their own model.
- I'm not sure which publications specifically we'll use, but I'll have to look into Gaussian Process literature to determine appropriate priors to use.
- Settles published a great survey on active learning 2010[4] that was instrumental in this idea and has a section on data generation. It should be explored more.
- There was a paper from Vis 2016 which aided users in selecting parameters[3] that had a really slick GUI. I don't know if we'll be using their methods, but I really liked their system.

11 Define Success

What is the minimum amount of work necessary for this work be publishable?

The algorithm needs to be implemented in python.

This could be publishable without a GUI and a user study. We could just look at some reference datasets, and simulate a domain expert with some outside knowledge, and show how the model chosen at the end by the automatic process performs better across all folds of cross validation.

References

[1] Z. Ghahramani. Probabilistic machine learning and artificial intelligence. *Nature*, 521(7553):452–459, 2015.

[2] M. Gleicher. Explainers: Expert explorations with crafted projections. *IEEE transactions on visualization and computer graphics*, 19(12):2042–2051, 2013.

[3] T. Löwe, E.-C. Förster, G. Albuquerque, J.-P. Kreiss, and M. Magnor. Visual analytics for development and evaluation of order selection criteria for autoregressive processes. *IEEE transactions on visualization and computer graphics*, 22(1):151–159, 2016.

[4] B. Settles. Active learning literature survey. *University of Wisconsin, Madison*, 52(55-66):11, 2010.