# AIStats?

# Story

Preference learning is useful for problems such as finding convincing arguments, where it is hard to give a score to a single document in isolation. Another example is finding a solution to question x; or choosing the best summary of a piece of text; or the best graph fit to some data.

In a large dataset, we cannot get enough pairs to label all items. Instead, we learn to generalise from the items’ features.

There are several ways to do preference learning that generalises beyond seen pairs and uses the structure in the features. GPPL has a number of advantages of using state-of-the-art nonparametric Bayesian model. Neural networks are an alternative but not yet tested on this problem – a likely drawback is the need for lots of training data before they can work well.

The problem with GPPL to date is that it was proposed with an inefficient MAP approximation, which may also give poorer results. Recent work on SVI shows how we can overcome this to scale to much larger datasets than were previously possible.

We present our SVI method.

The alternative to using our method is to alter the likelihood model of the Hensman approach:

* It’s unclear why this would be better. See the difference between the factorised and other models in Hensman’s paper.
* Does my method correspond to a factored approximation?

Alternative continuations of the story:

* Make the method more scalable by using active learning – compare BALD, max entropy, and max combined function variance. Need:
  + Results for active learning (implement and rerun with different sampling functions)
  + SVI vs. VB small data test
  + Clarify our method vs. Hensman, FITC, other popular sparse methods?
* Show how to apply SVI to the collaborative model, which contains multiple GPs and is therefore more complex. Need:
  + Results for collaborative model (bugfixes, simplify implementation and rerun). Detailed error analysis and tests with Lukin dataset & personality could be left for separate paper.
  + Can include variations on the collaborative model with common mean etc.
  + Can be extended to show common mean for finding ground truth.
  + Write-up the collaborative SVI equations
  + SVI vs. VB small data test
  + Clarify our method vs. Hensman, FITC, other popular sparse methods?

Experiments should show how well SVI works with GPPL.

*Novelty compared to TACL paper:*

* *we explain the methodology in detail showing how it was developed and why*
* *evaluate how SVI parameters affect convergence*
* *advantage of submitting now is that the two papers can be different enough, yet don’t require explicit comparison of novelty – make sure that key results from TACL paper are not spoilt?*

*In comparison to Hensman 2015:*

* *They do not show how to do this for preference learning, nor do they test it for this task.*
* *They do not use an approximation to the Beta distribution to handle the binary preference label probabilities. This means that the output noise is not easily interpretable?*

*Novelty compared to Houlsby:*

* *Theirs is a more complex model for multiple users. We suppose there is just one ground-truth preference function to learn.*
* *CP inference has complexity of: O(D(U^3 + P^3)) where P is number of pairs. There is a speedup from using FITC.*

Experiments:

* Experiments in Steinberg & Bonilla 2014:
  + Toy data, NLPD, table + 2 figures showing learning rate and example of function
* Real data with argumentation.
* Active learning with uncertainty sampling
  + why are we including this, isn’t it basically covered by Houlsby et al.? We can show whether their version works at scale. We can show a proper comparison of methods.
  + Compare current method (Maximum entropy sampling) with using variance in the latent function. This could be done as follows: sample from posterior of f\_1 – f\_2 to sample Phi(z). Variance of samples \approx variance of the beta over Phi(z). Possible advantage: item pairs that are very similar but well know are sampled less.
  + Houlsby et al. test BALD: expected change in entropy over Phi(z) given the latent function values. This is a reversal of expected change in entropy over the latent function given class labels. This method has an advantage over uncertainty sampling: it considers whether the uncertainty would be changed by the update.
* Small subset to compare SVI / VB:
  + Show how M affects quality of final solution. Plot with single line/points only. X-axis shows M, y-axis shows log marginal likelihood or AUC?
  + Show how P\_n affects time needed to converge. Plot with different lines for different P\_n. x-axis = no. iterations, y-axis = log marginal likelihood or AUC?
* Another large PL dataset, ideally from a different domain and with metrics for other rival methods.
  + Is this necessary? What have related papers done?
  + Houlsby:
    - 1.5 pages intro/background
    - 1.5 pages model
    - 1 page active learning method
    - 1 page inference
    - 1.2 pages experiments on core methodology
      * 5 standard datasets
      * table of test error with 100 users
      * table of training times with 100 users
      * table of test error with 1000 users when using active learning.
    - 0.8 pages on active learning experiments
      * 5 standard datasets
      * 5 plots showing error versus no. samples
    - 0.1 pages conclusion
  + Chu and Ghahramani:
    - 1.2 pages intro
    - 1.2 pages model
    - 1.2 pages inference
    - 1 page second method
    - 2.2 experiments
      * error rates against no. preferences
      * cpu time against no. preferences
      * classification performance table
      * ranking performance plots (error rate and correlation)

# Inferring the Ground Truth from Multiple Workers

Adding in a common mean to infer ground truth:

* Advantage: if we were to take a latent component as true instead, it is unclear how to set priors so that the common gold pattern is chosen for that feature.
* Is there a switch to turn off the spammers? In MACE, IBCC, we learn that spammers are random/always choose one label. Here we can learn either very high noise s\_k for worker k, or a factor that is (1 – mean) + f\_spammer.

One form of noise that is not well covered is in the labels themselves: the worker could click the left-hand button all the time, which would be detected using IBCC/MACE. In GPPL it would manifest eventually as high s\_k noise. Coupling the models together might help catch this noise earlier.

# To-Do list for Paper D:

1. Change style to IJCAI? IJCAI 2017 had a page limit of 6 pages for the main text + 1 for references. Deadline is end of January. ICML likely to be similar. AISTATS is mid October. AAAI is 8th September abstract and 11th paper; 7 pages main text + 1 references.
2. Create new ML paper \*[Paper D] Scalable GP Preference Learning\* outline. This can combine with technical bits from the personalised argumentation paper; also take on the crowdsourcing angle --> using the collaborative model to de-noise the crowdsourced data. May also need some scalability plots to show convergence.