Intelligent data acquisition using Bayesian principles

Freddie Bickford Smith

March 2026

OpenAI
@OpenAI

OpenAI is nothing without its people

> **Greg Brockman** @gdb · Nov 22, 2023
> we are so back
>
> 

8:05 AM · Nov 22, 2023 · **3.5M** Views

OpenAI is nothing without its data

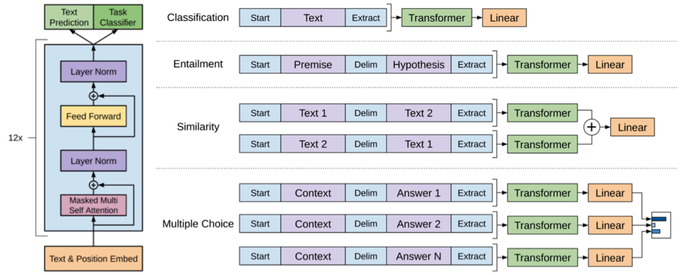


Figure 1: **(left)** Transformer architecture and training objectives used in this work. **(right)** Input transformations for fine-tuning on different tasks. We convert all structured inputs into token sequences to be processed by our pre-trained model, followed by a linear+softmax layer.

3.3 Task-specific input transformations

For some tasks, like text classification, we can directly fine-tune our model as described above. Certain other tasks, like question answering or textual entailment, have structured inputs such as ordered sentence pairs, or triplets of document, question, and answers. Since our pre-trained model was trained on contiguous sequences of text, we require some modifications to apply it to these tasks. Previous work proposed learning task specific architectures on top of transferred representations [44]. Such an approach re-introduces a significant amount of task-specific customization and does not use transfer learning for these additional architectural components. Instead, we use a traversal-style approach [52], where we convert structured inputs into an ordered sequence that our pre-trained model can process. These input transformations allow us to avoid making extensive changes to the architecture across tasks. We provide a brief description of these input transformations below and Figure 1 provides a visual illustration. All transformations include adding randomly initialized start and end tokens ($\langle s \rangle$, $\langle e \rangle$).

Textual entailment For entailment tasks, we concatenate the premise p and hypothesis h token sequences, with a delimiter token ($) in between.

Similarity For similarity tasks, there is no inherent ordering of the two sentences being compared. To reflect this, we modify the input sequence to contain both possible sentence orderings (with a delimiter in between) and process each independently to produce two sequence representations h_l^m which are added element-wise before being fed into the linear output layer.

Question Answering and Commonsense Reasoning For these tasks, we are given a context document z, a question q, and a set of possible answers $\{a_k\}$. We concatenate the document context and question with each possible answer, adding a delimiter token in between to get $[z; q; \$; a_k]$. Each of these sequences are processed independently with our model and then normalized via a softmax layer to produce an output distribution over possible answers.

4 Experiments

Unsupervised pre-training We use the BooksCorpus dataset [71] for training the language model. It contains over 7,000 unique unpublished books from a variety of genres including Adventure, Fantasy, and Romance. Crucially, it contains long stretches of contiguous text, which allows the generative model to learn to condition on long-range information. An alternative dataset, the 1B Word Benchmark, which is used by a similar approach, ELMo [44], is approximately the same size

GPT-1: 1 paragraph

Radford et al (2018)

to infer and perform many different tasks on examples with this type of format.

Language modeling is also able to, in principle, learn the tasks of McCann et al. (2018) without the need for explicit supervision of which symbols are the outputs to be predicted. Since the supervised objective is the the same as the unsupervised objective but only evaluated on a subset of the sequence, the global minimum of the unsupervised objective is also the global minimum of the supervised objective. In this slightly toy setting, the concerns with density estimation as a principled training objective discussed in (Sutskever et al., 2015) are side stepped. The problem instead becomes whether we are able to, in practice, optimize the unsupervised objective to convergence. Preliminary experiments confirmed that sufficiently large language models are able to perform multitask learning in this toy-ish setup but learning is much slower than in explicitly supervised approaches.

While it is a large step from the well-posed setup described above to the messiness of "language in the wild", Weston (2016) argues, in the context of dialog, for the need to develop systems capable of learning from natural language directly and demonstrated a proof of concept – learning a QA task without a reward signal by using forward prediction of a teacher's outputs. While dialog is an attractive approach, we worry it is overly restrictive. The internet contains a vast amount of information that is passively available without the need for interactive communication. Our speculation is that a language model with sufficient capacity will begin to learn to infer and perform the tasks demonstrated in natural language sequences in order to better predict them, regardless of their method of procurement. If a language model is able to do this it will be, in effect, performing unsupervised multitask learning. We test whether this is the case by analyzing the performance of language models in a zero-shot setting on a wide variety of tasks.

"I'm not the cleverest man in the world, but like they say in French: **Je ne suis pas un imbecile [I'm not a fool].**

In a now-deleted post from Aug. 16, Soheil Eid, Tory candidate in the riding of Joliette, wrote in French: "**Mentez mentez, il en restera toujours quelque chose**," which translates as, "**Lie lie and something will always remain.**"

"I hate the word '**perfume**,"' Burr says. 'It's somewhat better in French: '**parfum**.'

If listened carefully at 29:55, a conversation can be heard between two guys in French: "**-Comment on fait pour aller de l'autre coté? -Quel autre coté?**", which means "**- How do you get to the other side? - What side?**".

If this sounds like a bit of a stretch, consider this question in French: **As-tu aller au cinéma?**, or **Did you go to the movies?**, which literally translates as Have-you to go to movies/theater?

"**Brevet Sans Garantie Du Gouvernement**", translated to English: "**Patented without government warranty**".

Table 1. Examples of naturally occurring demonstrations of English to French and French to English translation found throughout the WebText training set.

Most prior work trained language models on a single domain of text, such as news articles (Jozefowicz et al., 2016), Wikipedia (Merity et al., 2016), or fiction books (Kiros et al., 2015). Our approach motivates building as large and diverse a dataset as possible in order to collect natural language demonstrations of tasks in as varied of domains and contexts as possible.

A promising source of diverse and nearly unlimited text is web scrapes such as Common Crawl. While these archives are many orders of magnitude larger than current language modeling datasets, they have significant data quality issues. Trinh & Le (2018) used Common Crawl in their work on commonsense reasoning but noted a large amount of documents "whose content are mostly unintelligible". We observed similar data issues in our initial experiments with Common Crawl. Trinh & Le (2018)'s best results were achieved using a small subsample of Common Crawl which included only documents most similar to their target dataset, the Winograd Schema Challenge. While this is a pragmatic approach to improve performance on a specific task, we want to avoid making assumptions about the tasks to be performed ahead of time.

Instead, we created a new web scrape which emphasizes document quality. To do this we only scraped web pages which have been curated/filtered by humans. Manually filtering a full web scrape would be exceptionally expensive so as a starting point, we scraped all outbound links from Reddit, a social media platform, which received at least 3 karma. This can be thought of as a heuristic indicator for whether other users found the link interesting, educational, or just funny.

The resulting dataset, WebText, contains the text subset of these 45 million links. To extract the text from HTML responses we use a combination of the Dragnet (Peters & Lecocq, 2013) and Newspaper[1] content extractors. All results presented in this paper use a preliminary version of WebText which does not include links created after Dec 2017 and which after de-duplication and some heuristic based cleaning contains slightly over 8 million documents for a total of 40 GB of text. We removed all Wikipedia documents from WebText since it is a common data source for other datasets and could complicate analysis due to over-

[1] https://github.com/codelucas/newspaper

GPT-1: 1 paragraph

GPT-2: 4 paragraphs

Radford et al (2019)

Model Name	n_{params}	n_{layers}	d_{model}	n_{heads}	d_{head}	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	6.0×10^{-4}
GPT-3 Medium	350M	24	1024	16	64	0.5M	3.0×10^{-4}
GPT-3 Large	760M	24	1536	16	96	0.5M	2.5×10^{-4}
GPT-3 XL	1.3B	24	2048	24	128	1M	2.0×10^{-4}
GPT-3 2.7B	2.7B	32	2560	32	80	1M	1.6×10^{-4}
GPT-3 6.7B	6.7B	32	4096	32	128	2M	1.2×10^{-4}
GPT-3 13B	13.0B	40	5140	40	128	2M	1.0×10^{-4}
GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128	3.2M	0.6×10^{-4}

Table 2.1: Sizes, architectures, and learning hyper-parameters (batch size in tokens and learning rate) of the models which we trained. All models were trained for a total of 300 billion tokens.

2.1 Model and Architectures

We use the same model and architecture as GPT-2 [RWC+19], including the modified initialization, pre-normalization, and reversible tokenization described therein, with the exception that we use alternating dense and locally banded sparse attention patterns in the layers of the transformer, similar to the Sparse Transformer [CGRS19]. To study the dependence of ML performance on model size, we train 8 different sizes of model, ranging over three orders of magnitude from 125 million parameters to 175 billion parameters, with the last being the model we call GPT-3. Previous work [KMH+20] suggests that with enough training data, scaling of validation loss should be approximately a smooth power law as a function of size; training models of many different sizes allows us to test this hypothesis both for validation loss and for downstream language tasks.

Table 2.1 shows the sizes and architectures of our 8 models. Here n_{params} is the total number of trainable parameters, n_{layers} is the total number of layers, d_{model} is the number of units in each bottleneck layer (we always have the feedforward layer four times the size of the bottleneck layer, $d_{\text{ff}} = 4 * d_{\text{model}}$), and d_{head} is the dimension of each attention head. All models use a context window of $n_{\text{ctx}} = 2048$ tokens. We partition the model across GPUs along both the depth and width dimension in order to minimize data-transfer between nodes. The precise architectural parameters for each model are chosen based on computational efficiency and load-balancing in the layout of models across GPU's. Previous work [KMH+20] suggests that validation loss is not strongly sensitive to these parameters within a reasonably broad range.

2.2 Training Dataset

Datasets for language models have rapidly expanded, culminating in the Common Crawl dataset[2] [RSR+19] constituting nearly a trillion words. This size of dataset is sufficient to train our largest models without ever updating on the same sequence twice. However, we have found that unfiltered or lightly filtered versions of Common Crawl tend to have lower quality than more curated datasets. Therefore, we took 3 steps to improve the average quality of our datasets: (1) we downloaded and filtered a version of CommonCrawl based on similarity to a range of high-quality reference corpora, (2) we performed fuzzy deduplication at the document level, within and across datasets, to prevent redundancy and preserve the integrity of our held-out validation set as an accurate measure of overfitting, and (3) we also added known high-quality reference corpora to the training mix to augment CommonCrawl and increase its diversity.

Details of the first two points (processing of Common Crawl) are described in Appendix A. For the third, we added several curated high-quality datasets, including an expanded version of the WebText dataset [RWC+19], collected by scraping links over a longer period of time, and first described in [KMH+20], two internet-based books corpora (Books1 and Books2) and English-language Wikipedia.

Table 2.2 shows the final mixture of datasets that we used in training. The CommonCrawl data was downloaded from 41 shards of monthly CommonCrawl covering 2016 to 2019, constituting 45TB of compressed plaintext before filtering and 570GB after filtering, roughly equivalent to 400 billion byte-pair-encoded tokens. Note that during training, datasets are not sampled in proportion to their size, but rather datasets we view as higher-quality are sampled more frequently, such that CommonCrawl and Books2 datasets are sampled less than once during training, but the other datasets are sampled 2-3 times. This essentially accepts a small amount of overfitting in exchange for higher quality training data.

[2] https://commoncrawl.org/the-data/

GPT-1: 1 paragraph

GPT-2: 4 paragraphs

GPT-3: ~10 paragraphs

Brown et al (2020)

from experience. Care should be taken when using the outputs of GPT-4, particularly in contexts where reliability is important.

GPT-4's capabilities and limitations create significant and novel safety challenges, and we believe careful study of these challenges is an important area of research given the potential societal impact. This report includes an extensive system card (after the Appendix) describing some of the risks we foresee around bias, disinformation, over-reliance, privacy, cybersecurity, proliferation, and more. It also describes interventions we made to mitigate potential harms from the deployment of GPT-4, including adversarial testing with domain experts, and a model-assisted safety pipeline.

2 Scope and Limitations of this Technical Report

This report focuses on the capabilities, limitations, and safety properties of GPT-4. GPT-4 is a Transformer-style model [39] pre-trained to predict the next token in a document, using both publicly available data (such as internet data) and data licensed from third-party providers. The model was then fine-tuned using Reinforcement Learning from Human Feedback (RLHF) [40]. Given both the competitive landscape and the safety implications of large-scale models like GPT-4, this report contains no further details about the architecture (including model size), hardware, training compute, dataset construction, training method, or similar.

We are committed to independent auditing of our technologies, and shared some initial steps and ideas in this area in the system card accompanying this release.[2] We plan to make further technical details available to additional third parties who can advise us on how to weigh the competitive and safety considerations above against the scientific value of further transparency.

3 Predictable Scaling

A large focus of the GPT-4 project was building a deep learning stack that scales predictably. The primary reason is that for very large training runs like GPT-4, it is not feasible to do extensive model-specific tuning. To address this, we developed infrastructure and optimization methods that have very predictable behavior across multiple scales. These improvements allowed us to reliably predict some aspects of the performance of GPT-4 from smaller models trained using $1,000\times - 10,000\times$ less compute.

3.1 Loss Prediction

The final loss of properly-trained large language models is thought to be well approximated by power laws in the amount of compute used to train the model [41, 42, 2, 14, 15].

To verify the scalability of our optimization infrastructure, we predicted GPT-4's final loss on our internal codebase (not part of the training set) by fitting a scaling law with an irreducible loss term (as in Henighan et al. [15]): $L(C) = aC^b + c$, from models trained using the same methodology but using at most 10,000x less compute than GPT-4. This prediction was made shortly after the run started, without use of any partial results. The fitted scaling law predicted GPT-4's final loss with high accuracy (Figure 1).

3.2 Scaling of Capabilities on HumanEval

Having a sense of the capabilities of a model before training can improve decisions around alignment, safety, and deployment. In addition to predicting final loss, we developed methodology to predict more interpretable metrics of capability. One such metric is pass rate on the HumanEval dataset [43], which measures the ability to synthesize Python functions of varying complexity. We successfully predicted the pass rate on a subset of the HumanEval dataset by extrapolating from models trained with at most $1,000\times$ less compute (Figure 2).

For an individual problem in HumanEval, performance may occasionally worsen with scale. Despite these challenges, we find an approximate power law relationship $-\mathrm{E}_P[\log(\mathrm{pass\_rate(C)})] = \alpha * \mathrm{C}^{-k}$

[2] In addition to the accompanying system card, OpenAI will soon publish additional thoughts on the social and economic implications of AI systems, including the need for effective regulation.

GPT-1: 1 paragraph

GPT-2: 4 paragraphs

GPT-3: ~10 paragraphs

GPT-4: too shy 🫣

OpenAI (2023)

Scale it up

As pre-training data on the internet dry up, post-training is more important. Labelling companies such as Scale AI and Surge AI earn hundreds of millions of dollars a year collecting post-training data. Scale recently raised $1bn on a $14bn valuation. Things have moved on from the Mechanical Turk days: the best labellers earn up to $100 an hour. But, though post-training helps produce better models and is sufficient for many commercial applications, it is ultimately incremental.

Back to jobs

Data Curator (Chemistry), London

London

Apply

Data Curator (Chemistry), London

Isomorphic Labs is a new Alphabet company that is reimagining drug discovery through a computational- and AI-first approach.

We are on a mission to accelerate the speed, increase the efficacy and lower the cost of drug discovery. You'll be working at the cutting edge of the new era of 'digital biology' to deliver a transformative social impact for the benefit of millions of people.

Come and be part of a multi-disciplinary team driving groundbreaking innovation and play a meaningful role in contributing towards us achieving our ambitious goals, while being a part of an inspiring, collaborative and entrepreneurial culture.

Your impact

This is an exciting opportunity to join the newly established data team at IsoLabs, working closely with world leading AI experts and Drug Discovery scientists to establish machine learning ready datasets that power the discovery of the next generation of medicines. As a data curator specialising in chemistry and biochemical

WANTED

Effective automatic methods
for identifying good data

REWARD: $bns

👉 Good data conveys relevant information

👉 Information means reduced uncertainty

ON A MEASURE OF THE INFORMATION PROVIDED BY AN EXPERIMENT[1, 2]

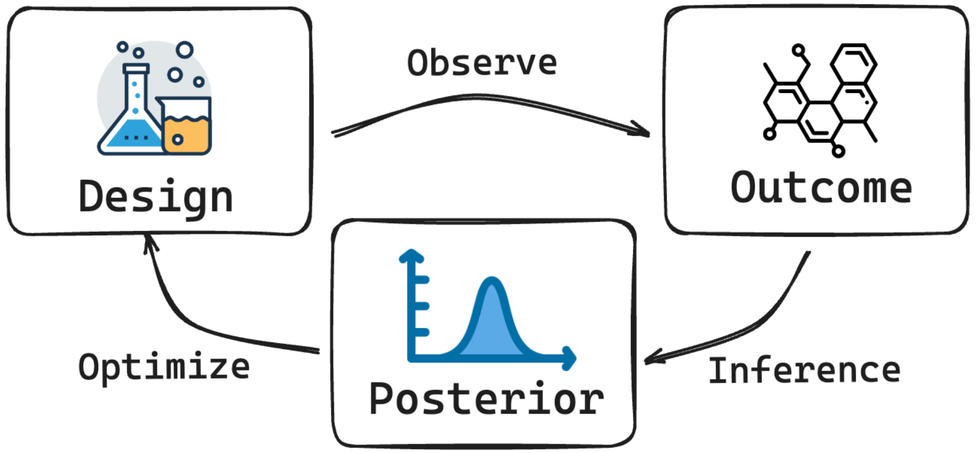
By D. V. Lindley

University of Cambridge and University of Chicago

1. Summary. A measure is introduced of the information provided by an experiment. The measure is derived from the work of Shannon [10] and involves the knowledge prior to performing the experiment, expressed through a prior probability distribution over the parameter space. The measure is used to compare some pairs of experiments without reference to prior distributions; this method of comparison is contrasted with the methods discussed by Blackwell. Finally, the measure is applied to provide a solution to some problems of experimental design, where the object of experimentation is not to reach decisions but rather to gain knowledge about the world.

2. Introduction. Shannon has introduced two important ideas into the theory of information in communications engineering. The first idea is that information is a statistical concept. The statistical frequency distribution of the symbols that make up a message must be considered before the notion can be discussed adequately. The second idea springs from the first and implies that on the basis of the frequency distribution, there is an essentially unique function of the distribution which measures the amount of the information. It is the purpose of the present paper to apply these two ideas to statistical theory by discussing the notion of information in an experiment, rather than in a message. The second of Shannon's ideas has been applied to statistical theory by Kullback and Leibler [6], [7], [8]; but our application is quite distinct from theirs. The interpretation of Shannon's ideas in current statistical theory has been given by McMillan [9]. The discussion in that paper is related to, and partly inspired, that given here. A referee has kindly pointed out that Shannon's theory has been applied in psychometric problems by L. J. Cronbach in an unpublished report [14]. Definition 2, in particular, is used by Cronbach.

The situation in communications engineering is that there is a transmitted message, x, which is received as a message, y. By considerations of the informations in x and y it is possible to discuss the rate at which information has been transmitted along the channel. The analogous description in statistical theory



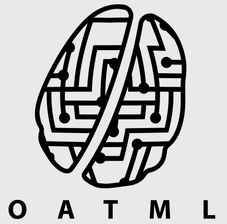
Lindley (1956), Huang et al (2024)

Prediction-oriented Bayesian active learning

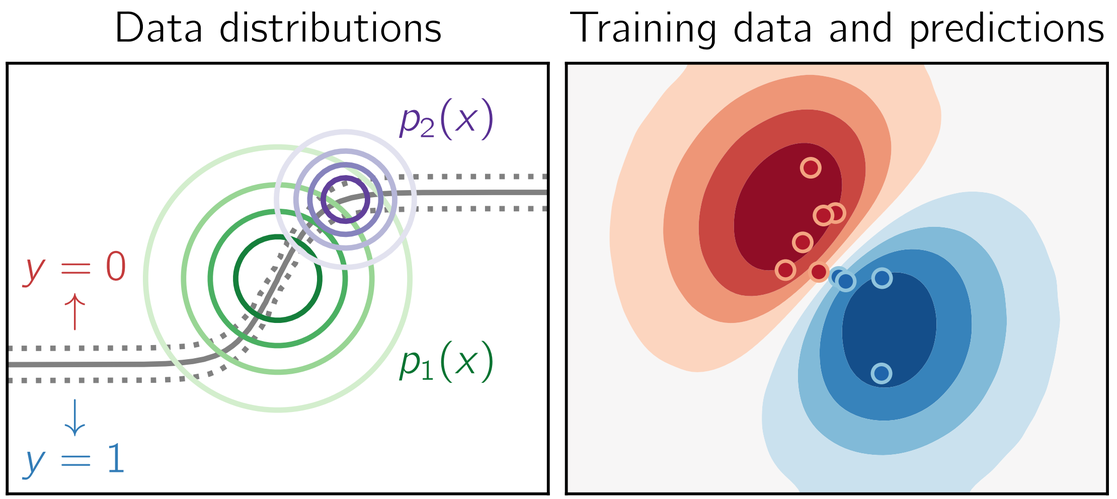
Target info gain in predictions, not parameters, for better data acquisition

Freddie Bickford Smith*, Andreas Kirsch*, Sebastian Farquhar,
Yarin Gal, Adam Foster, Tom Rainforth

AISTATS 2023

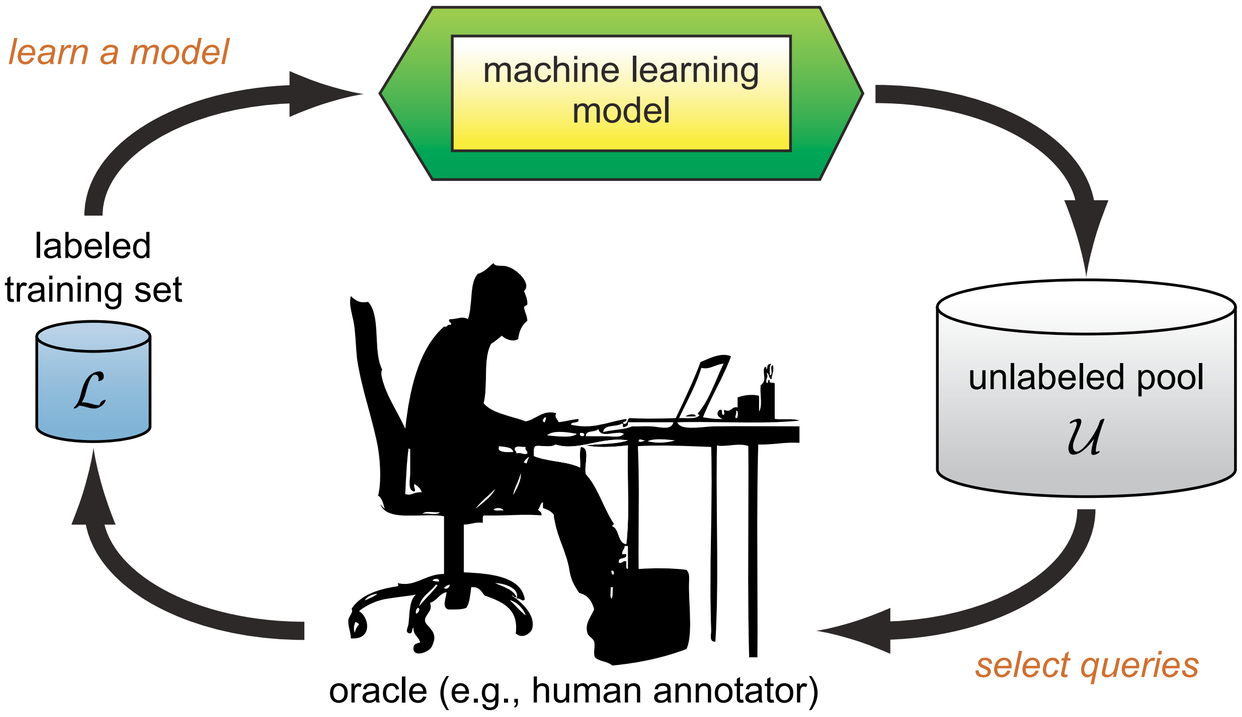


What label should we acquire next?



We need a method for automatically identifying useful data

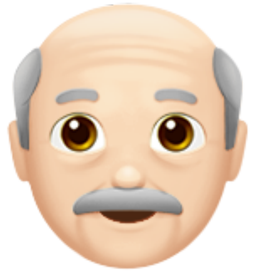
Lots of work focuses on “pool-based” active learning



Often we’re doing

$$x_{\text{acq}} = \arg\max_{x \in \mathcal{U}} a(x)$$

Settles (2009)

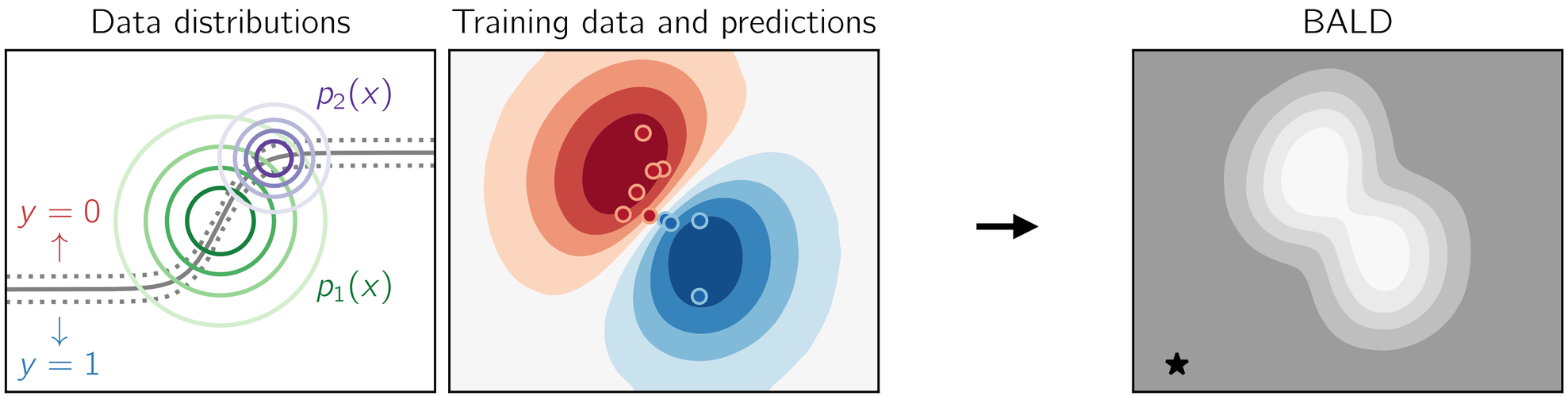


BALD

EPIG

Problem

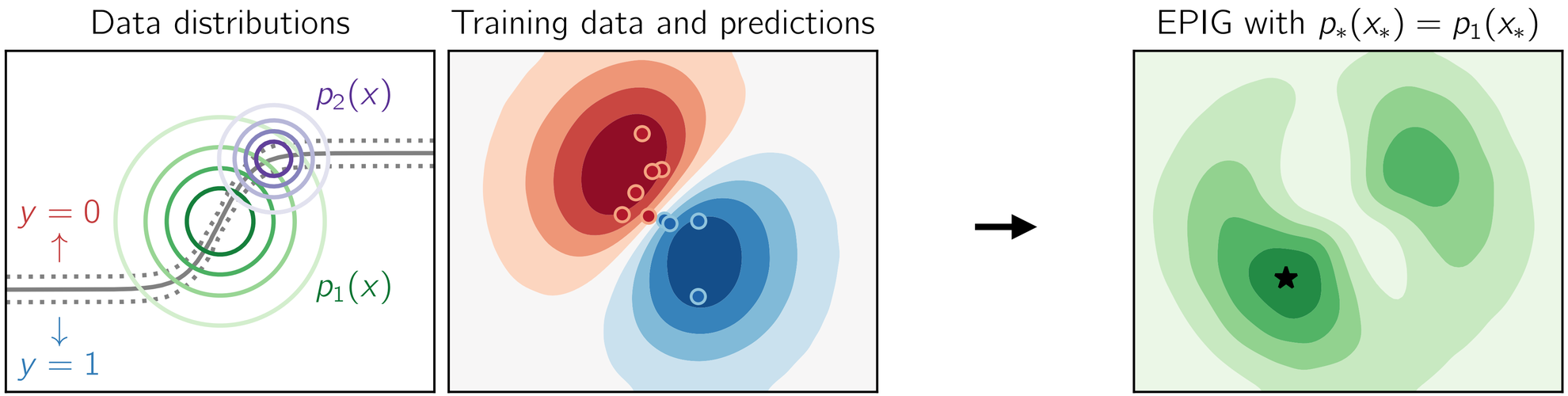
BALD is widely used but often suboptimal



It tends to prioritise data that has limited relevance to prediction

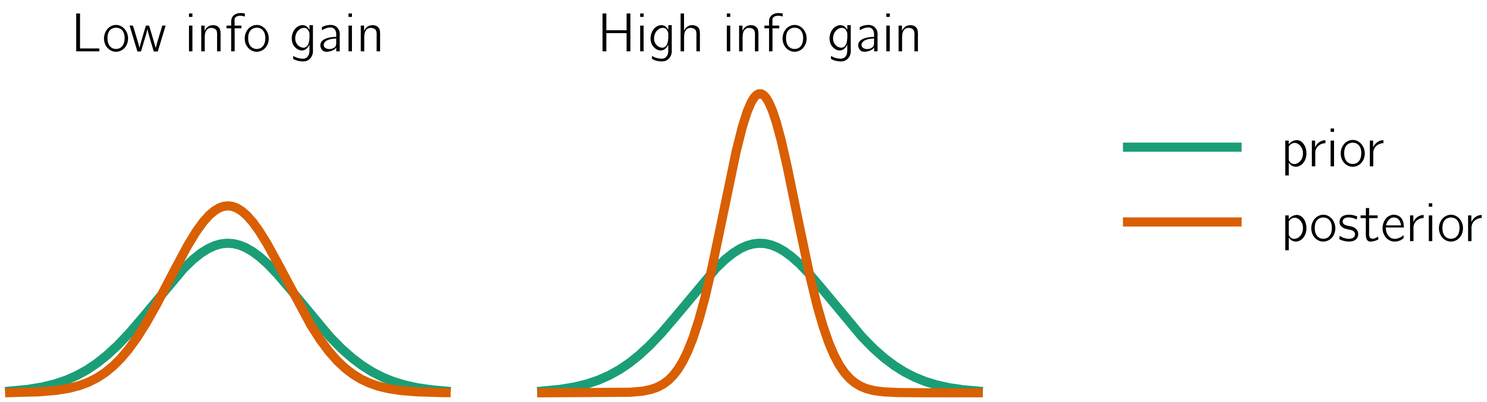
Solution

EPIG targets the predictions of interest



This involves explicitly accounting for the input distribution

We can use Bayesian experimental design to target info gain



The framework:

1. Identify a variable of interest
2. Quantify uncertainty in that variable
3. Seek to maximise info gain (= uncertainty reduction)

BALD = expected info gain in the model parameters

Quantify a reduction in parameter uncertainty:

$$\mathrm{IG}_\theta(x, y) = \mathrm{H}[p_\phi(\theta)] - \mathrm{H}[p_\phi(\theta|x, y)]$$

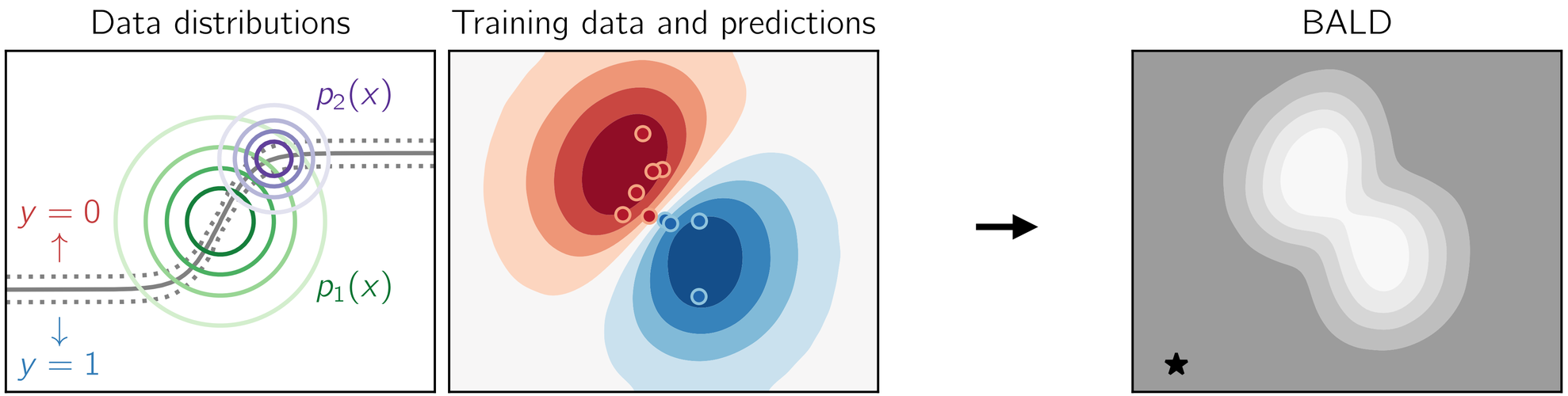
Reason about the unknown label:

$$\begin{aligned}\mathrm{BALD}(x) &= \mathbb{E}_{p_\phi(y|x)}[\mathrm{IG}_\theta(x, y)] \\ &= \mathrm{I}(\theta; y|x)\end{aligned}$$

BALD = Bayesian active learning by disagreement

Failure mode

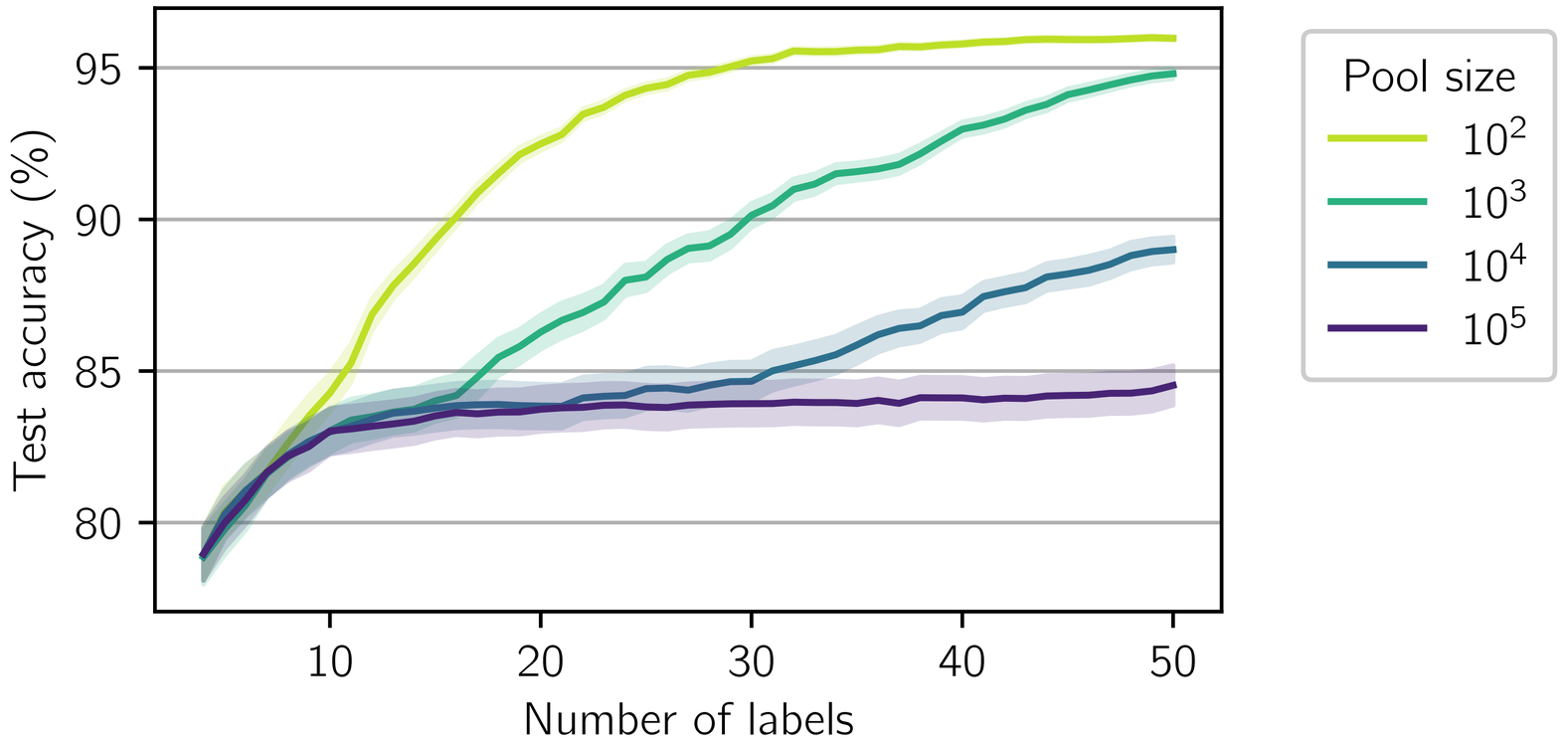
BALD tends to favour obscure data over relevant data



It seeks parameter info without regard for the input distribution

Failure mode

BALD is a liability with big, diverse pools of data



The greater the amount of irrelevant data, the worse BALD performs

Failure mode

We can acquire infinite useless info with BALD

Context: Gaussian-process regression with $x, y \in \mathbb{R}$

Suppose we care about prediction on $x_* \in [0, 1]$

For input locations $M, 2M, \ldots, M^2$, where $M \in \mathbb{N}^+$, we have

$$\lim_{M \to \infty} \mathrm{BALD}((M, 2M, \ldots, M^2)) = \infty$$

$$\lim_{M \to \infty} \mathrm{EIG}_{\theta(x_*)}((M, 2M, \ldots, M^2)) = 0$$



EPIG = expected info gain in the model's predictions

Quantify a reduction in predictive uncertainty:

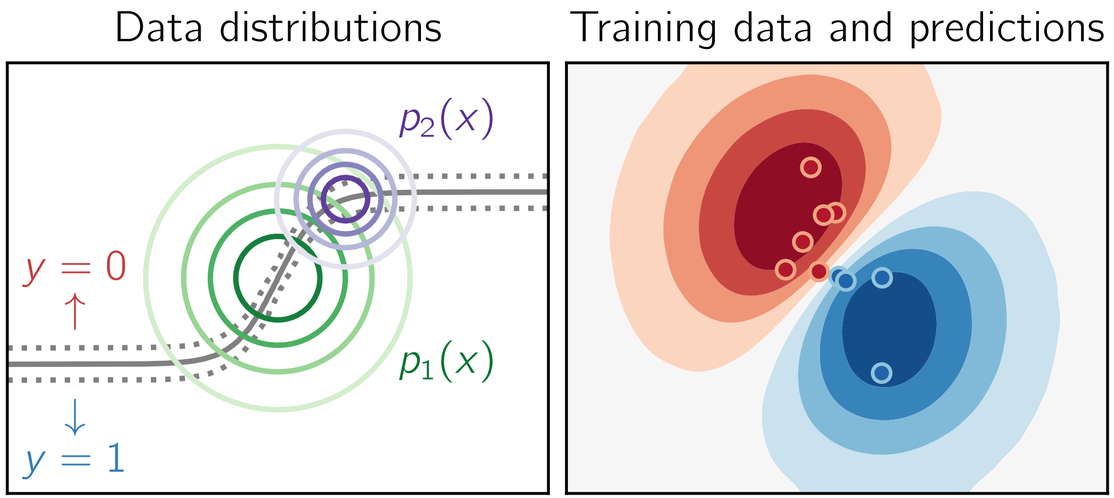
$$\mathrm{IG}_{y_*}(x, y, x_*) = \mathrm{H}[p_\phi(y_*|x_*)] - \mathrm{H}[p_\phi(y_*|x_*, x, y)]$$

Reason about the unknown label and target input:

$$\begin{aligned}\mathrm{EPIG}(x) &= \mathbb{E}_{p_*(x_*)p_\phi(y|x)}[\mathrm{IG}_{y_*}(x, y, x_*)] \\ &= \mathrm{I}(x_*, y_*; y|x)\end{aligned}$$

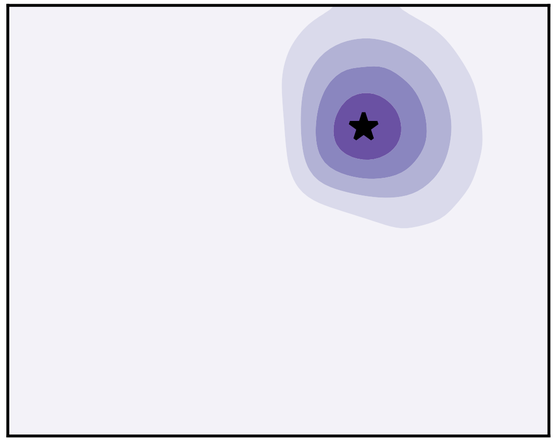
EPIG = expected predictive information gain

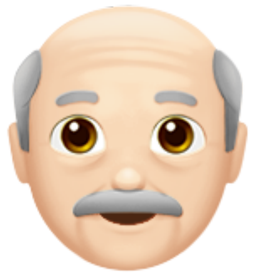
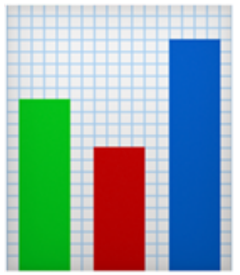
EPIG depends on the target input distribution



EPIG with $p_*(x_*) = p_1(x_*)$

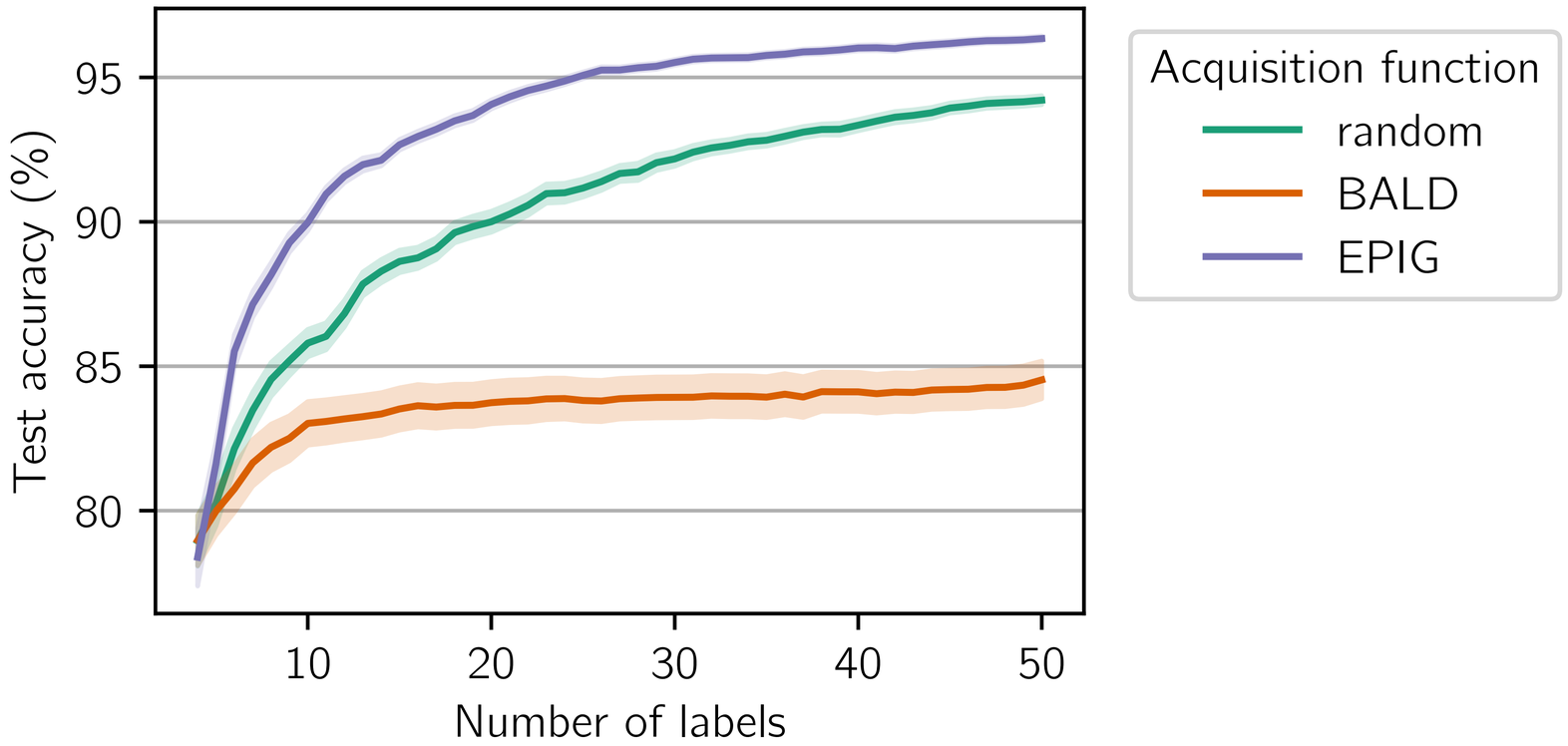
EPIG with $p_*(x_*) = p_2(x_*)$




Results on synthetic data

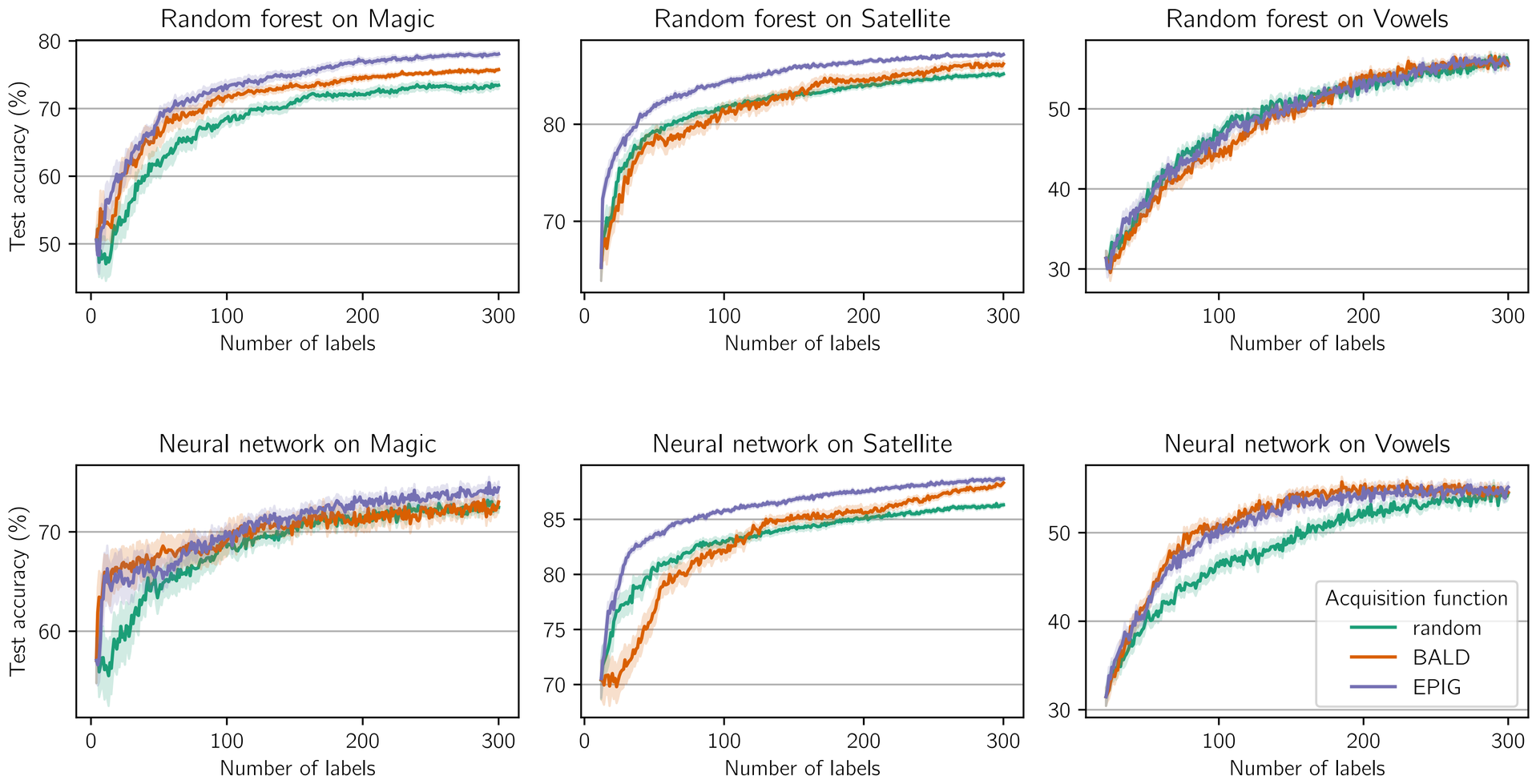
EPIG works where BALD fails



The pool here contains 100,000 unlabelled inputs

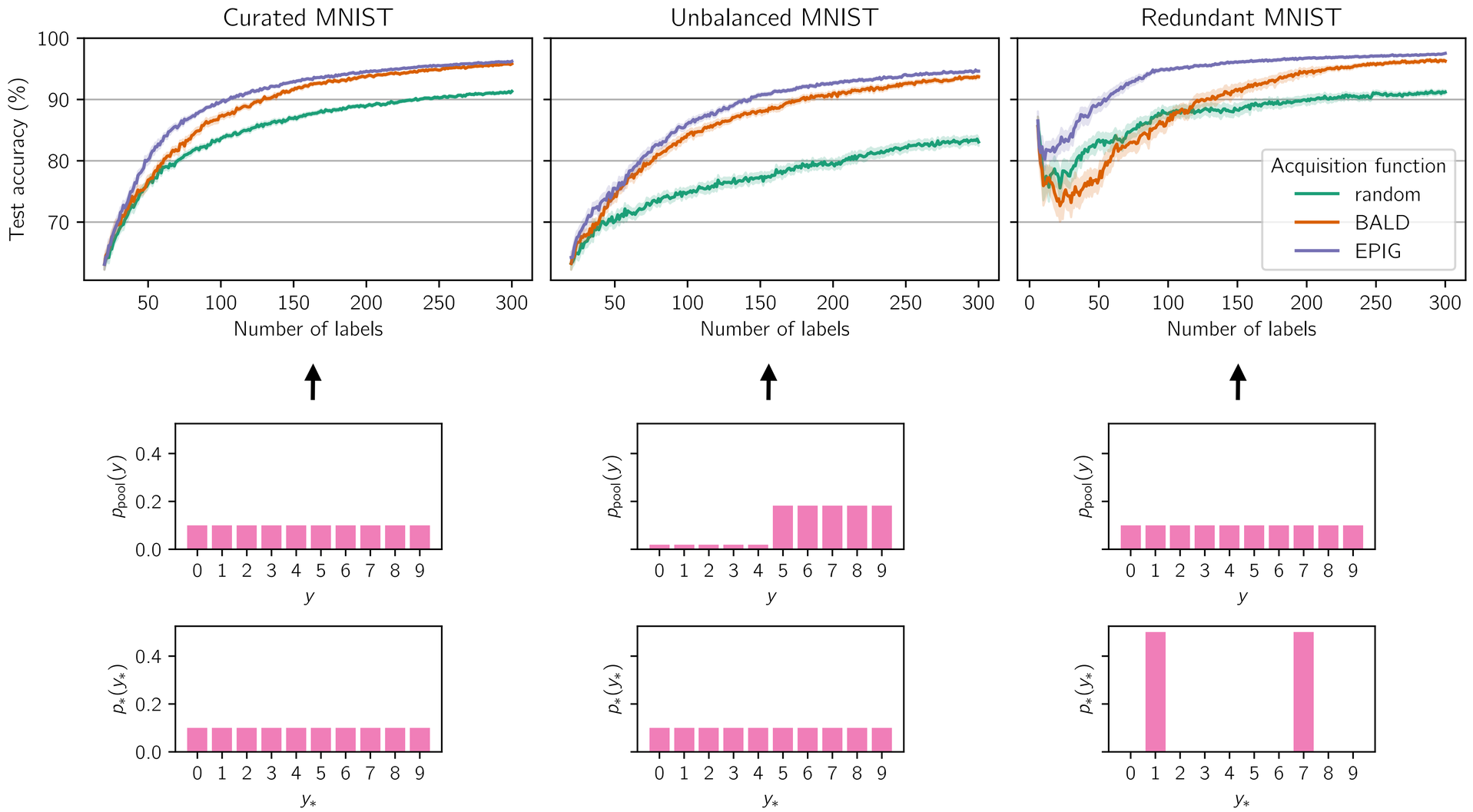
Results on UCI data

EPIG beats or matches BALD on standard datasets



Results on MNIST data

EPIG beats or matches BALD on standard datasets



Since the 1990s, Bayesian active learning ≈ maximising BALD

But BALD is suboptimal in prediction-oriented settings

We revisit the 1950s framework that BALD is derived from

EPIG, our proposal, serves as a drop-in replacement for BALD

Twitter thread

Paper

Code

Making better use of unlabelled data in Bayesian active learning

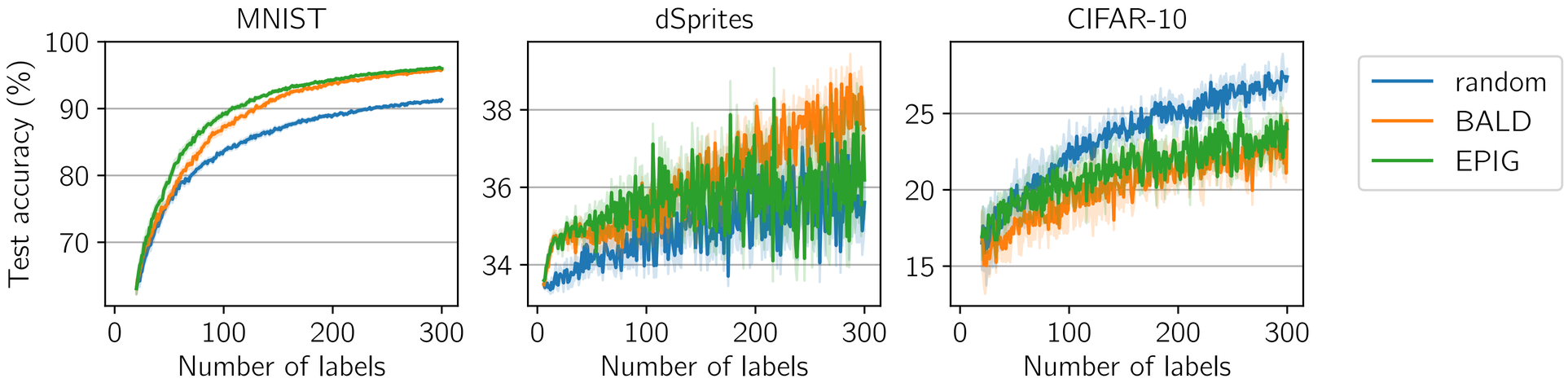
Semi-supervised models support better data acquisition

Freddie Bickford Smith, Adam Foster, Tom Rainforth

AISTATS 2024

Problem

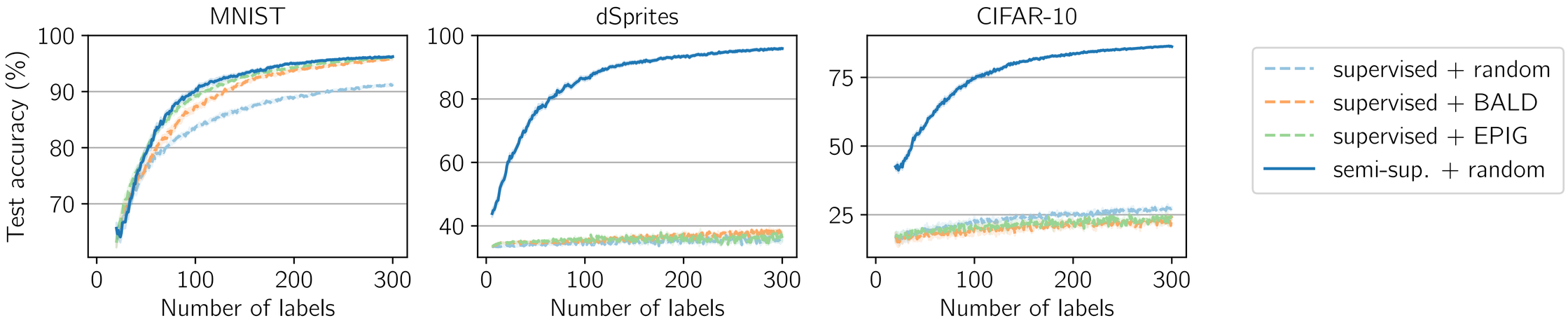
How do we scale Bayesian active learning beyond MNIST?



The standard recipe does not work on higher-dimensional inputs

Solution

Semi-supervised models are a strong basis for active learning

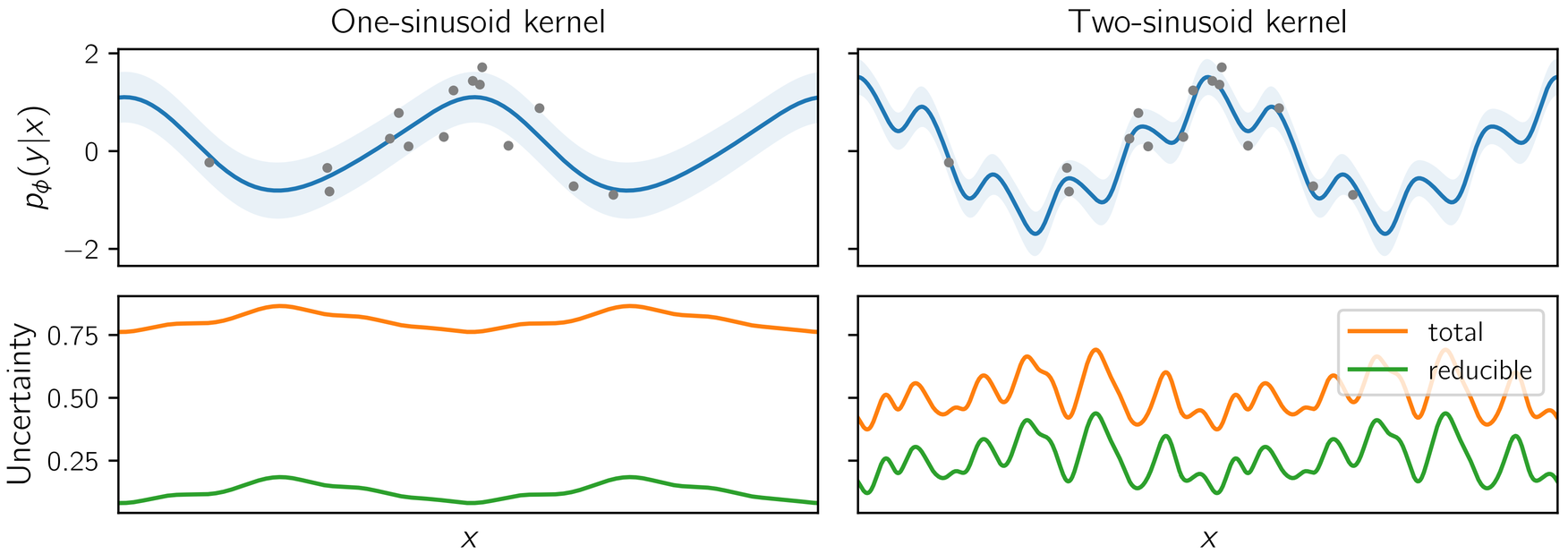


We need a good model for prediction but also for data acquisition

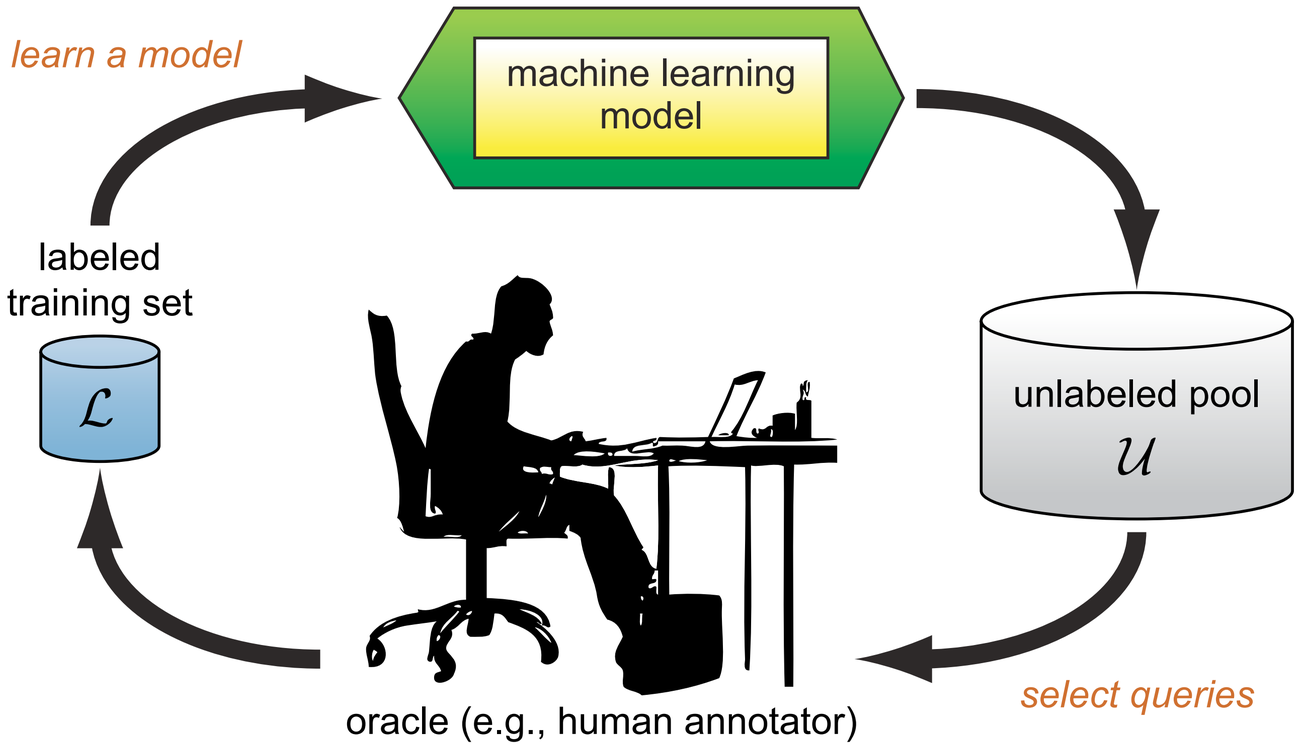
Bayesian active learning relies on decomposing uncertainty

$$\underbrace{\mathrm{BALD}(x)}_{\text{reducible}} = \underbrace{\mathrm{H}[p_\phi(y|x)]}_{\text{total}} - \underbrace{\mathbb{E}_{p_\phi(\theta)}[\mathrm{H}[p_\phi(y|x,\theta)]]}_{\text{irreducible}}$$

The decomposition into reducible vs irreducible depends on the model:



Supervised models fail to capture info in unlabelled data

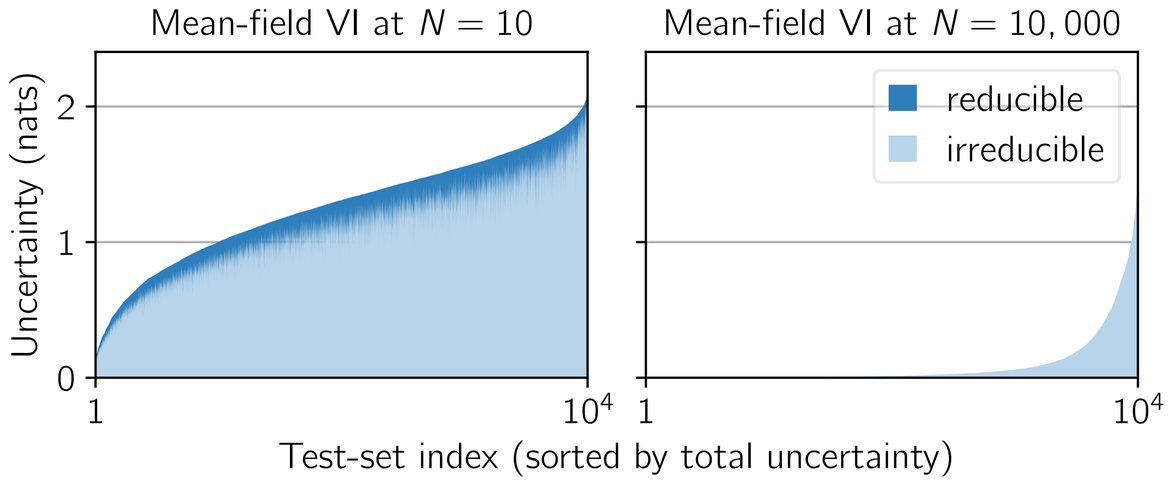


Manually encoding insights from unlabelled data doesn't scale

Settles (2009)

Failure mode

Supervised Bayesian NNs can have pathological uncertainty



A flexible model can make correct, confident predictions on unseen data

In other words, its actual irreducible uncertainty is low (see N = 10,000)

Yet at N = 10 it is pessimistic about how much uncertainty can be reduced



Simple combo: unsupervised pretraining + Bayesian AL

$$x \xrightarrow[\text{encoder}]{g} z \xrightarrow[\text{pred. head}]{h} p_\phi(y|x)$$

Encoder captures info from unlabelled data, and is fixed and deterministic

Prediction head captures label info, and is trainable and stochastic, giving

$$p_\phi(y|x) = \mathbb{E}_{p_\phi(\theta_h)}[p_\phi(y|g(x), \theta_h)]$$

Overall the model is semi-supervised, incorporating both forms of data

Prediction-oriented data acquisition with EPIG is key

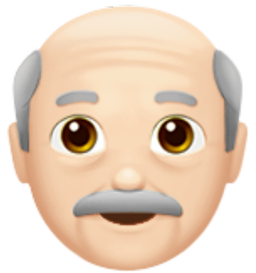
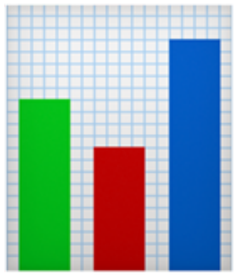
Adapting a result from prior work, we prove that

$$\mathrm{EPIG}(x) \leq \mathrm{BALD}(x)$$

So EPIG is a “filtered” version of BALD that retains relevant info

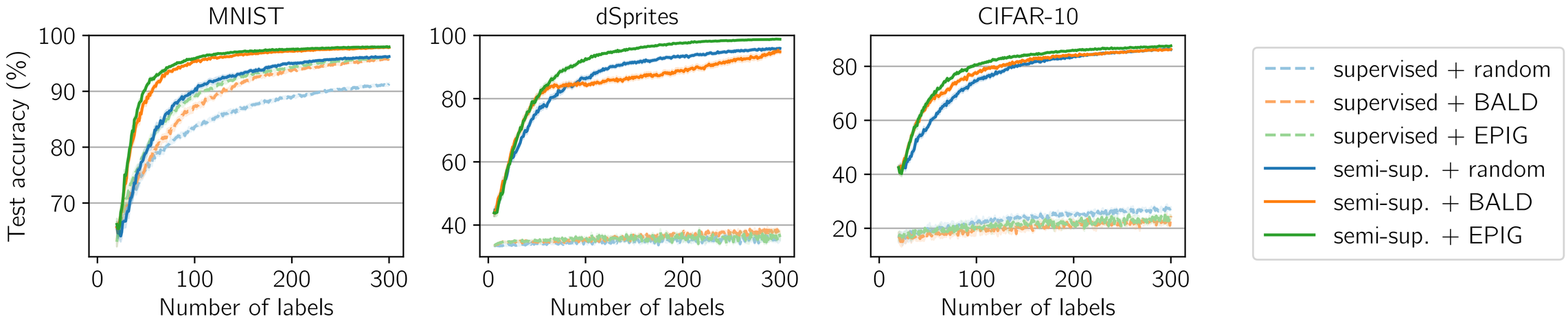
Complementary roles:

1. Unsupervised pretraining provides breadth
2. EPIG supports focused learning

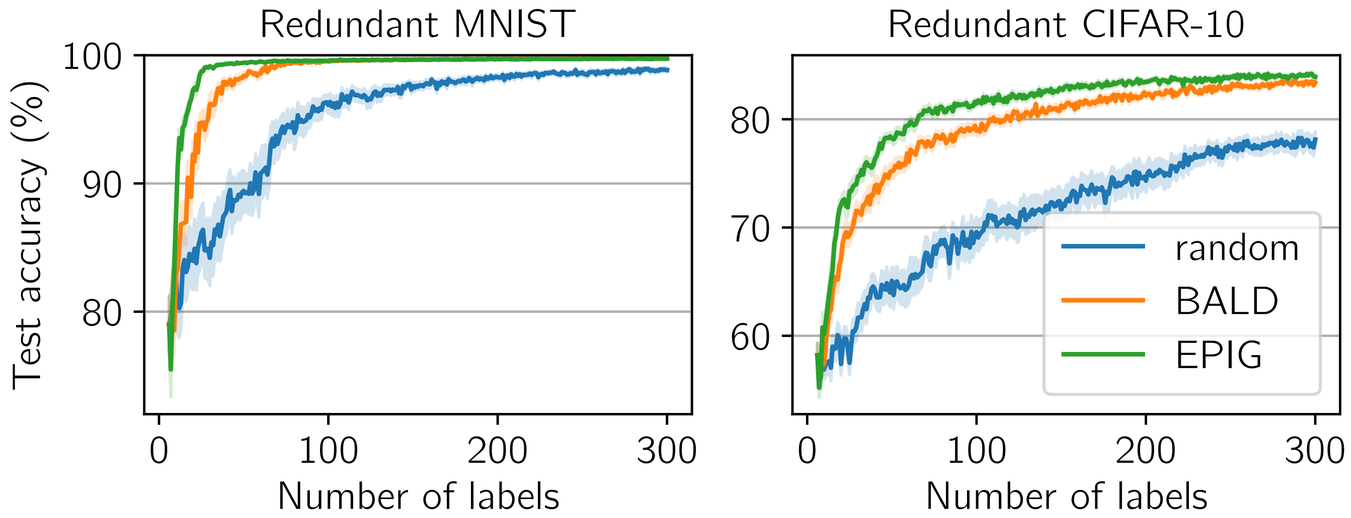

Results on standard data

EPIG acquisition provides reliable gains while BALD does not



Focusing on predictions is critical for beating random acquisition

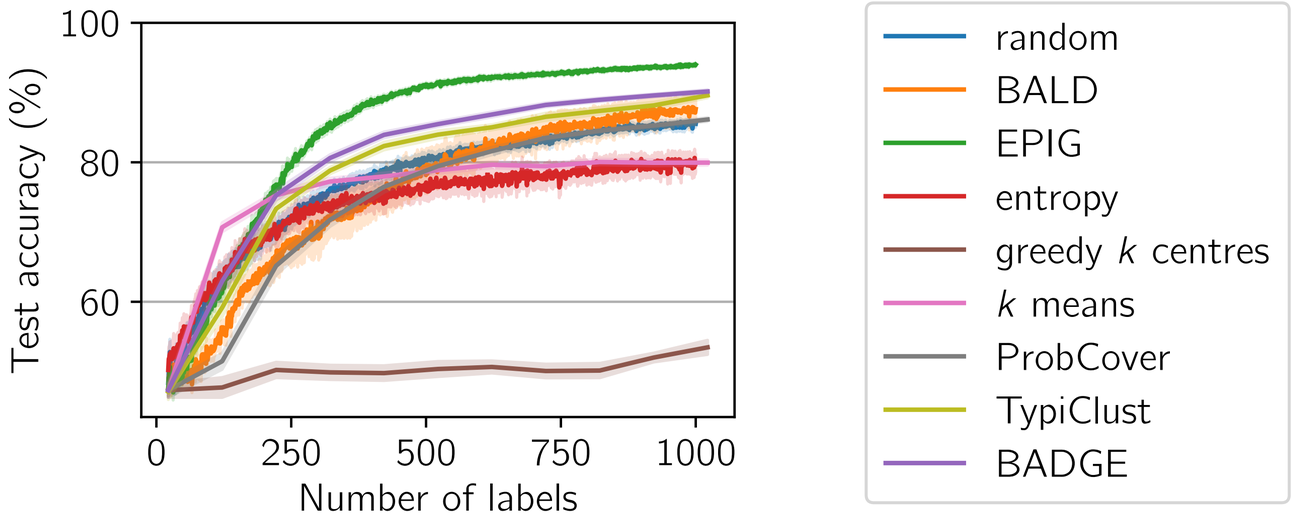
Bayesian AL deals well with messy pools of unlabelled data



Here the pools contain inputs from lots of irrelevant classes

Results on ImageNet data

Bayesian AL scales to high-dimensional ImageNet inputs



Here we classify inputs into eleven coarse superclasses

Practical benefit

Our proposed approach allows faster active learning

Dataset	Time (sup.)	Time (semi)	Speedup	Enc. time
MNIST	32 sec	17 sec	2×	2 msec
dSprites	4 min 17 sec	16 sec	16×	6 msec
CIFAR-10	42 min 14 sec	29 sec	89×	75 msec

The fixed, deterministic encoder allows us to cache embeddings

The lightweight prediction head speeds up updating and estimation

The speedup could enable new practical use-cases for Bayesian AL

Bayesian active learning has historically focused on supervised models

These fail to capture the abundant info in unlabelled data

We propose a simple framework for incorporating unlabelled data

This allows us to scale up Bayesian active learning effectively

---

Twitter thread

Paper

Code

Focusing on predictions and incorporating unlabelled data are key

Models and acquisition methods need to be studied in conjunction

Bayesian active learning has a bright future ahead of it

There’s scope for work on more reliable uncertainty estimation, moving beyond pools, asynchronous/batch/non-myopic acquisition, …

---

rainml.uk