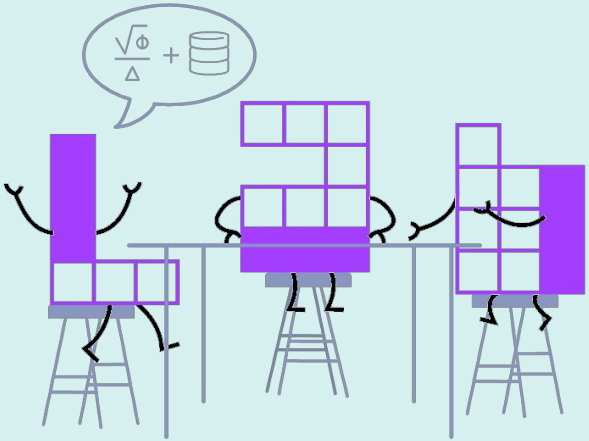
Active Learning in the Real World

AMLD Tutorial, October 26th

Alexandre ABRAHAM & Léo DREYFUS-SCHMIDT

Senior Research Scientist

Research Director



Dataiku is a software editor providing a data science platform to industrial customers

Lead efforts around Active Learning and development of the ML-assisted Labeling plugin

Introduction

Tutorial Outline

Part 1 - Motivation

- When to use active learning?
- Review of the classical approaches

Part 2 - State of the art methods

- Criteria optimized by active learning methods
- Review of recent active learning methods

Part 3 - Active Learning Ops

- Monitoring an active learning experiment

Industrial talk:

Asset Integrity Inspection using (Active) Machine Learning by Nader Salman, Project Manager Data Science Platform at Schlumberger

Introduction

Before we start

Let’s get to know each other !

Tell us your name, company and your experience with labelling & active learning.

Tutorial Resources:

- Github repo with the notebooks hands-on
  https://github.com/dataiku-research/active-learning-tutorial

- Slack channel
  #active-learning-in-the-real-world

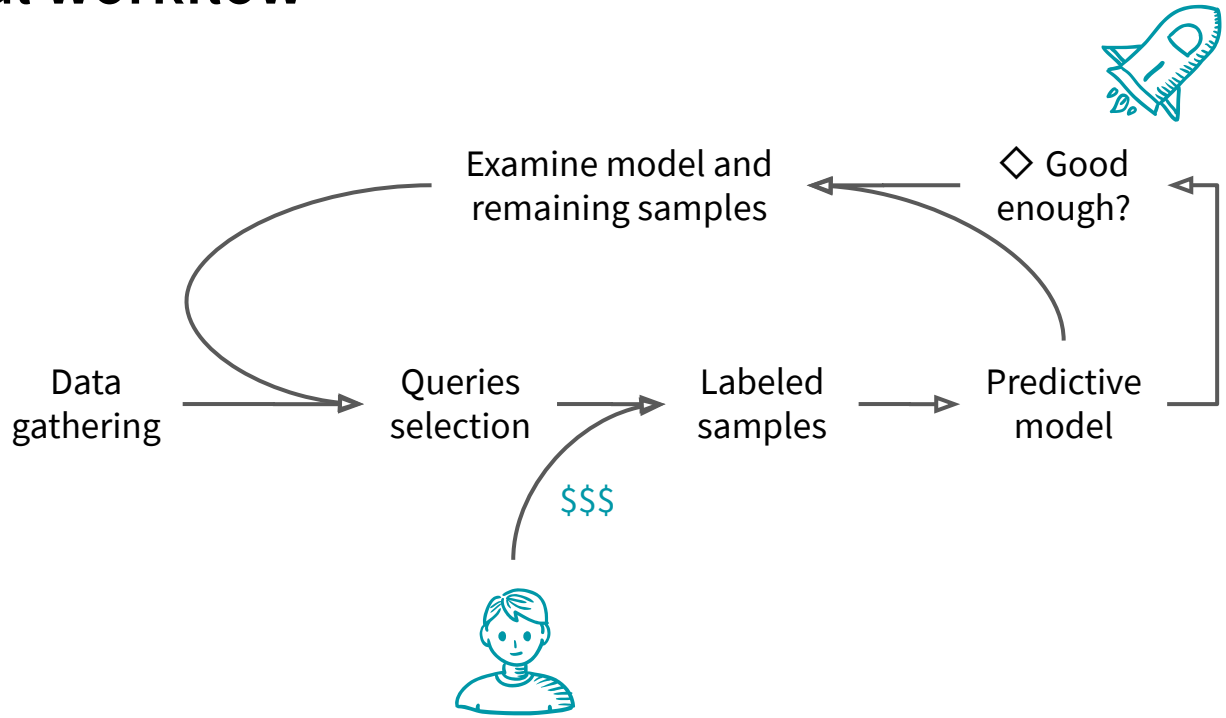
Experiment #1

Life of a labeler

Part 1 - What is Active Learning ?

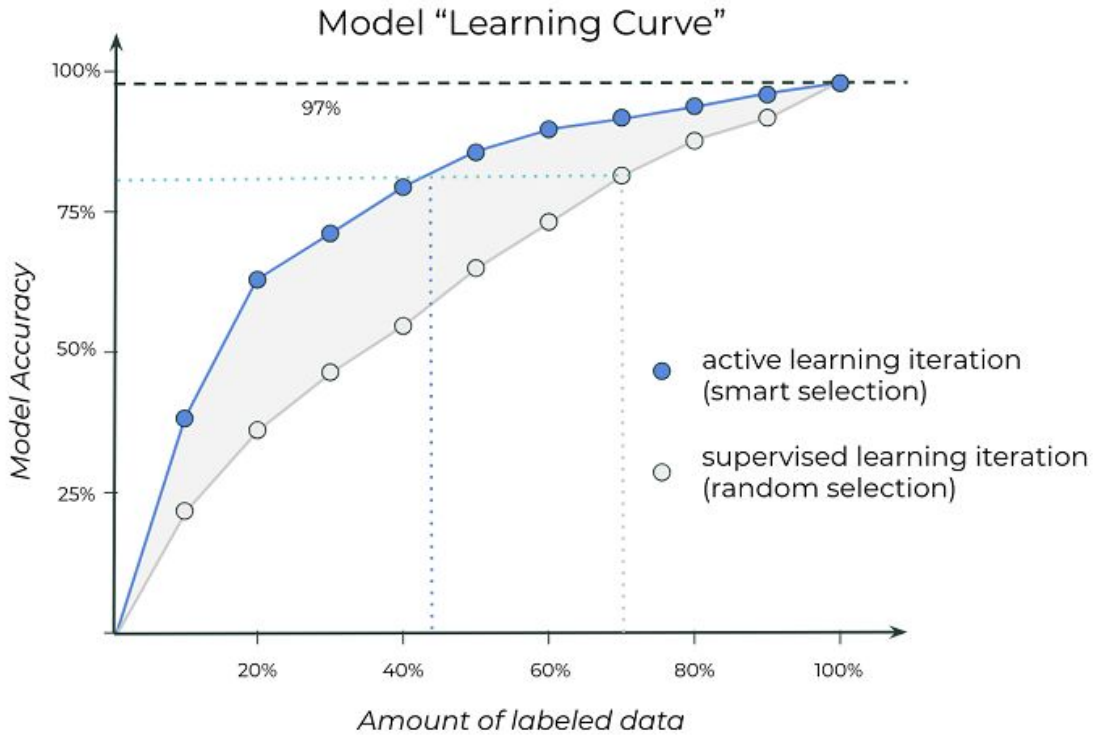
Motivation

Typical workflow



The Theoretical Promise

Learn More With Less

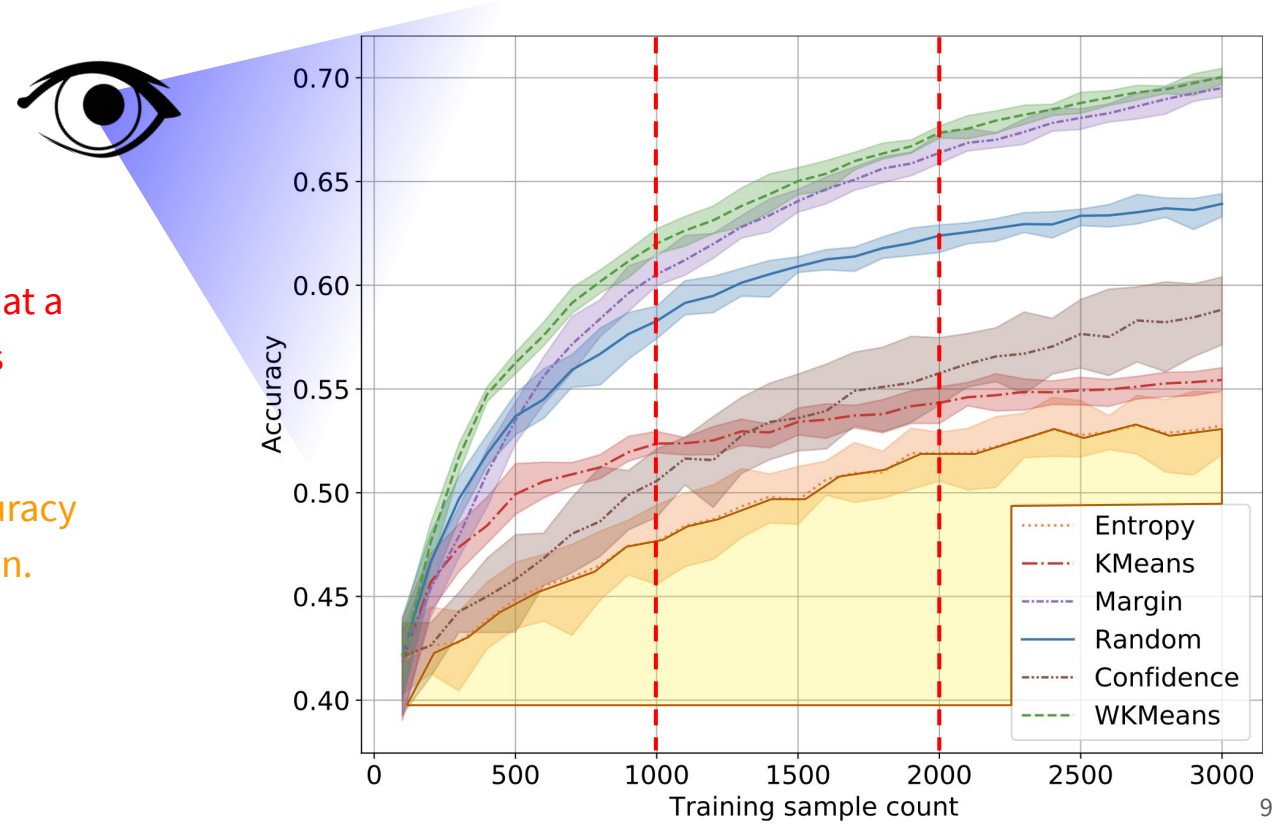


How to analyze an Active Learning experiment?

Most studies rely on visual interpretation of the curve

Some papers use the accuracy at a given point or at fixed intervals

Others use the area under accuracy curve (AUC). We take this option.



When to use Active Learning?

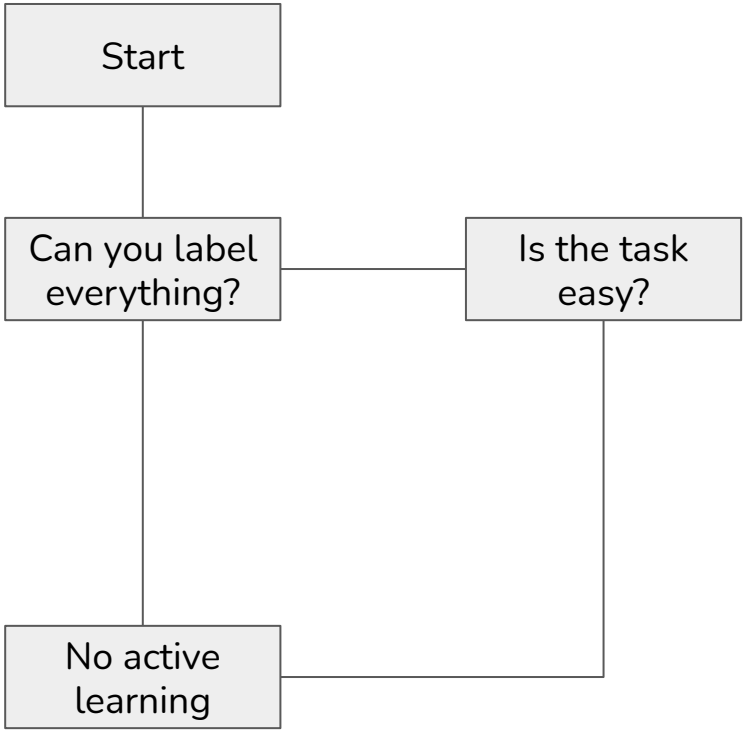
Definition. Active learning aim at minimizing data labeling cost to reach a given model performance. The strategy is defined by a **sampler**, also called a **query strategy**, that select batch of data to be labelled and then fed to the **learner.**

The baseline sampler corresponding to *passive learning* is the **Random sampler**.

Active learning may have hidden costs:

- Setting up the labeling interface
- Need for expert annotation
- Cost of data gathering (Biopsy of tumorous cells)

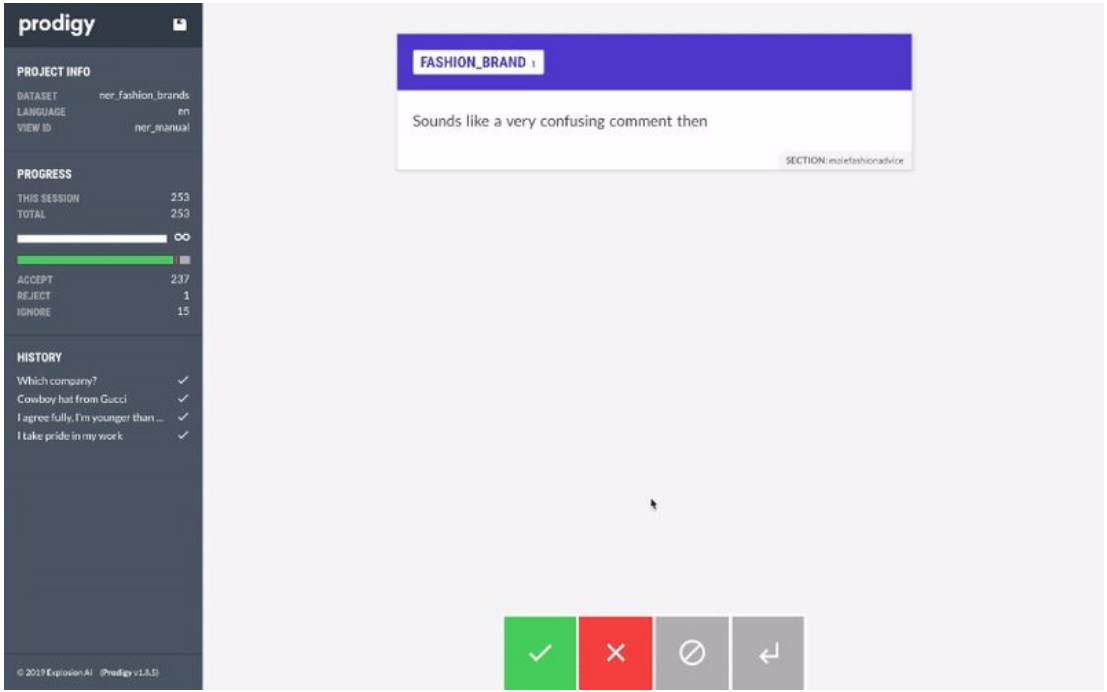
So should I do Active learning at all?



Some Active Learning Use Cases

NLP Task

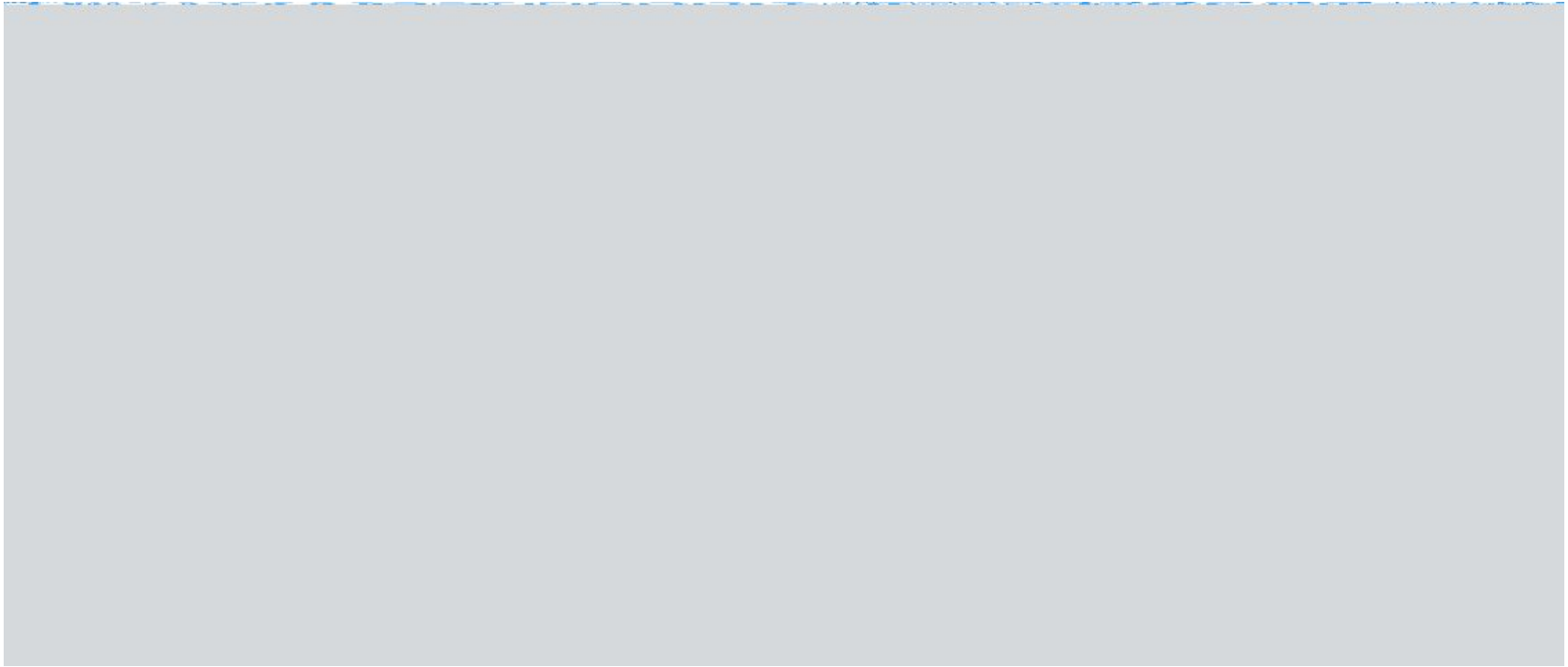
Named Entity Recognition



Prodigy Smart Labeling

Computer Vision Task

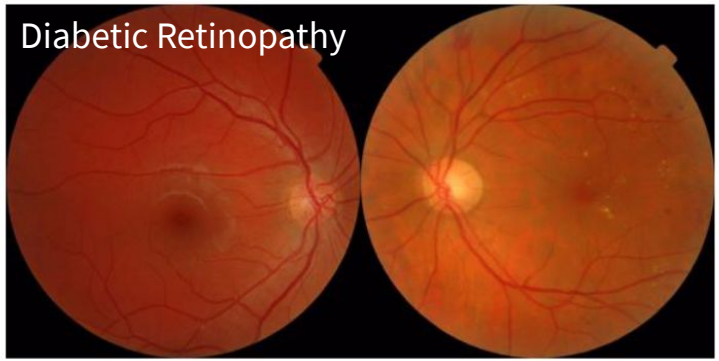
Segmentation / Object detection



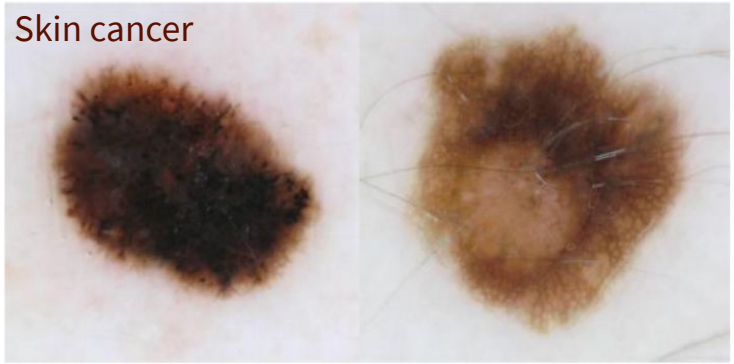
Supervisely Smart Labeling

Complex Labeling Tasks

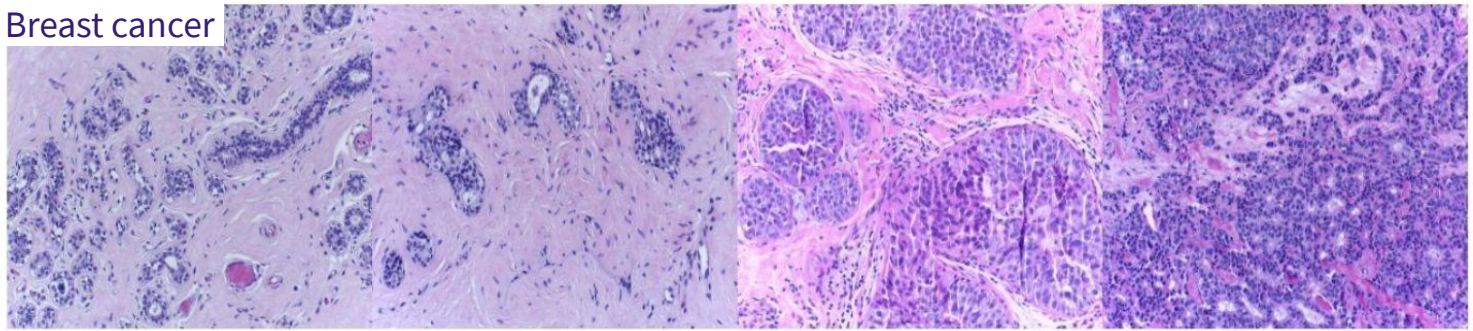
Query sampling for medical imaging [Smailagic 2018]



No DR DR



Benign Malignant



Normal Benign *In situ* Invasive

Wrap Up

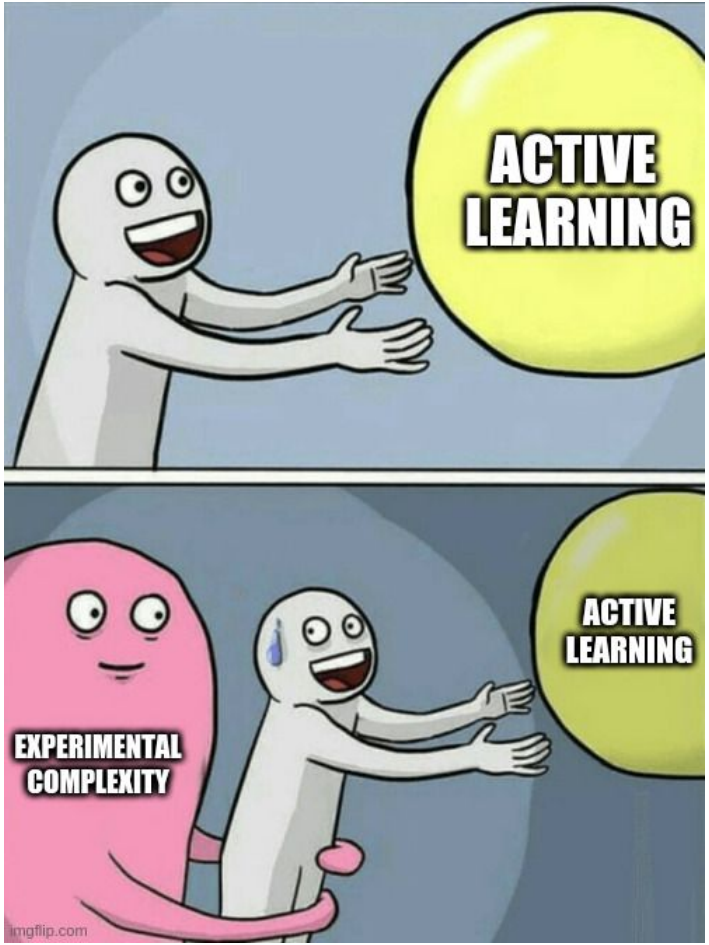
In conclusion

Just do

```
pip install active-learning
```

… but it’s not that simple. A wide variety of methods exist and after a quick look at the literature, you’ll find there is no consensus.

Even worst, there sometimes is no scientific rigor !



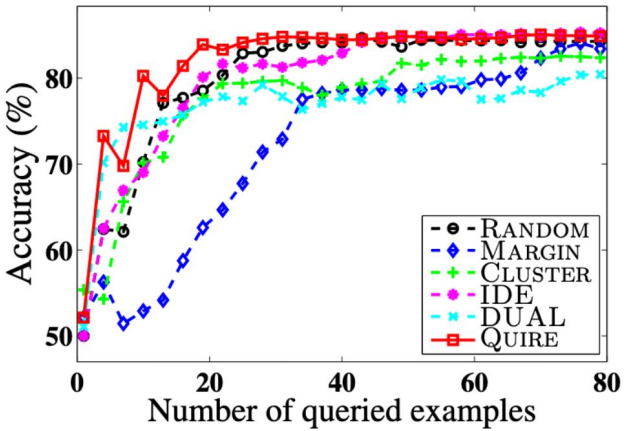
Research Challenges of Active Learning

Research Challenges

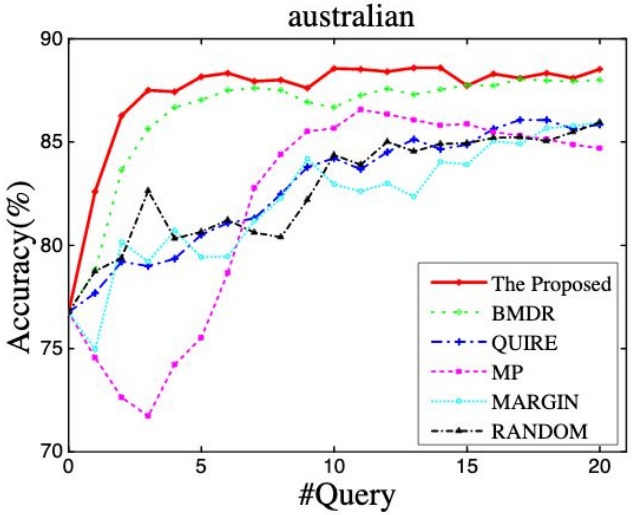
Inconsistent results across studies

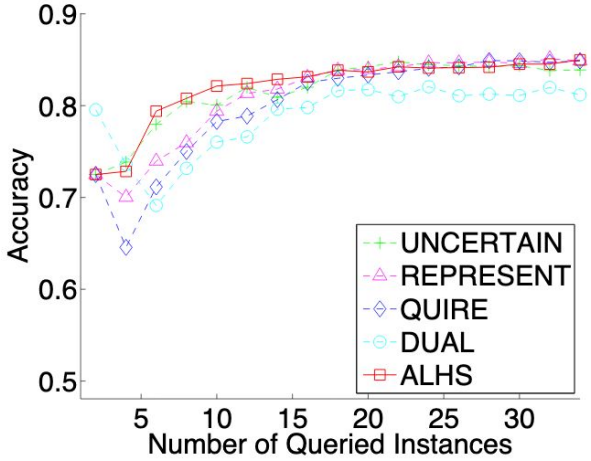
The best option depends on protocol parameters: model, batch size, baseline samplers...
What is the best option?

Top right is [Du 2015], bottom right is [Li 2012], below is [Huang 2010]



(a) *austra*





(a) *australian*

Research Challenges

High Results Sensibility

Experimental results depend a lot on the seed used to perform the experiments.
Showing the noise is paramount! [Kottke 2017]

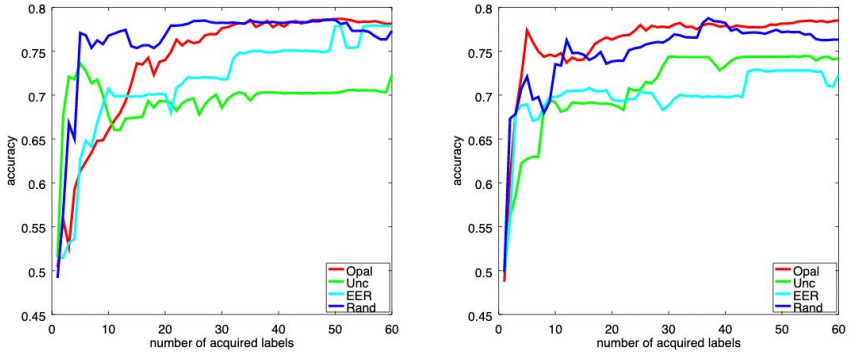
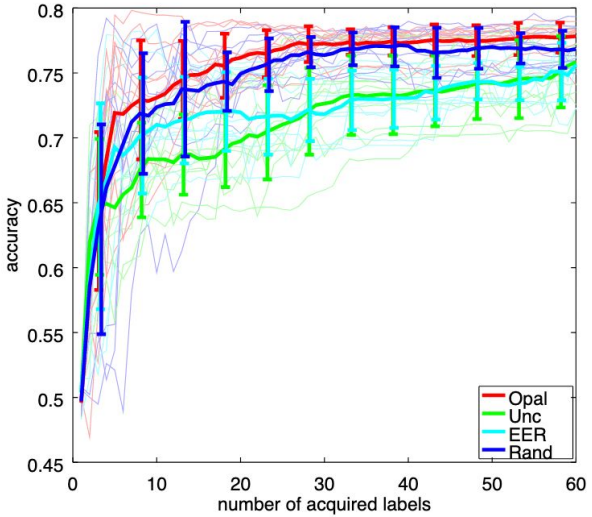


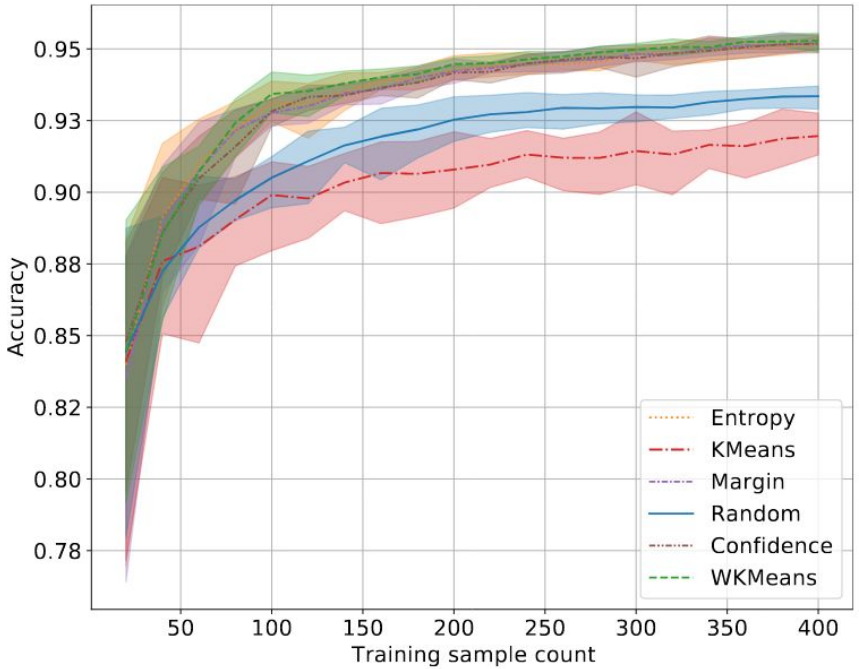
Fig. 3. Results of a 5-fold cross validation: two executions with different seeds of a complete 5-fold cross validation.

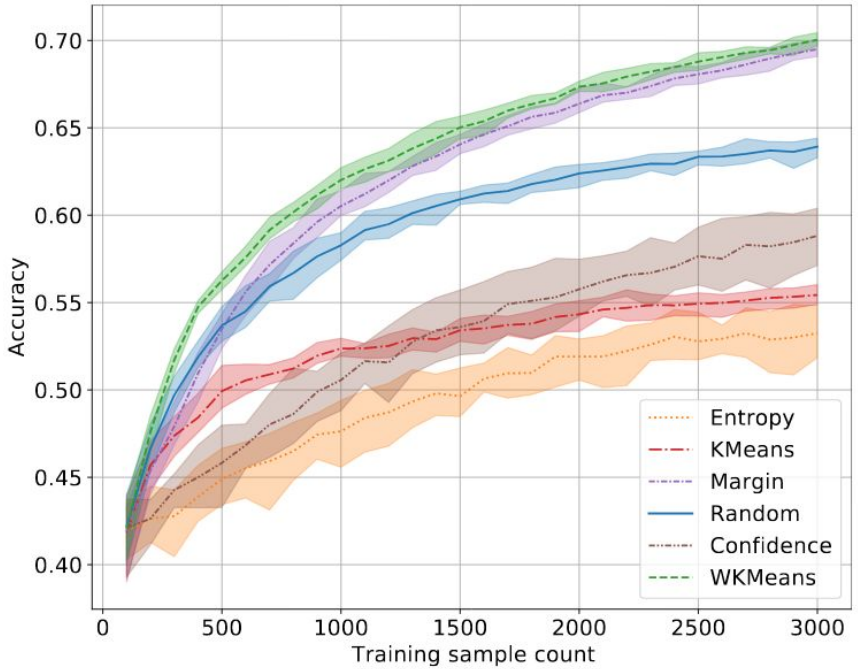


Experiment with uncertainty bounds

Research Challenges

High Results Variability of Samplers



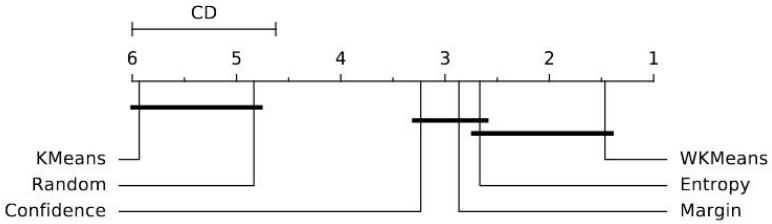


Accuracy comparison between NOMAO, easy (left) and LDPA, hard (right)

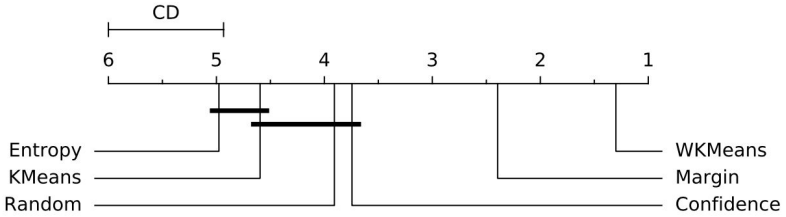
Research Challenges

High Results Variability of Samplers

Easy tasks



Hard tasks



Good accuracy achieved with few samples

Model selection is not critical

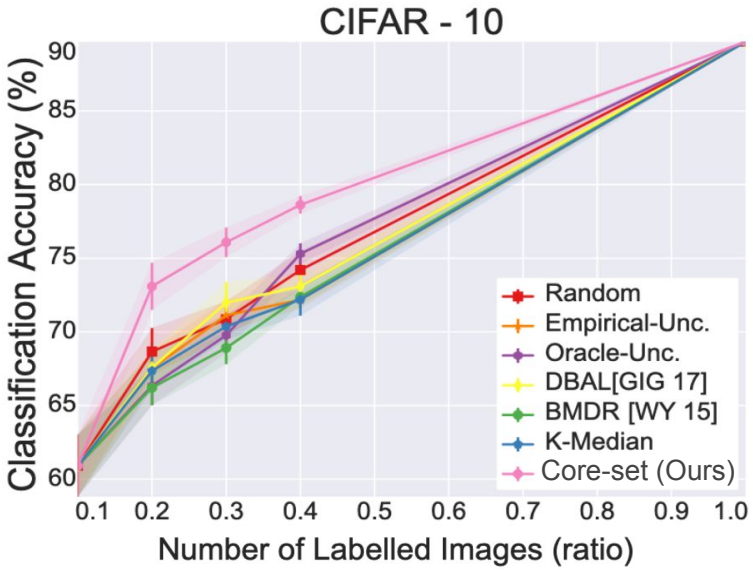
- NOMAO
- Phishing Websites
- Wall Robot Navigation

Best accuracy is hard to achieve

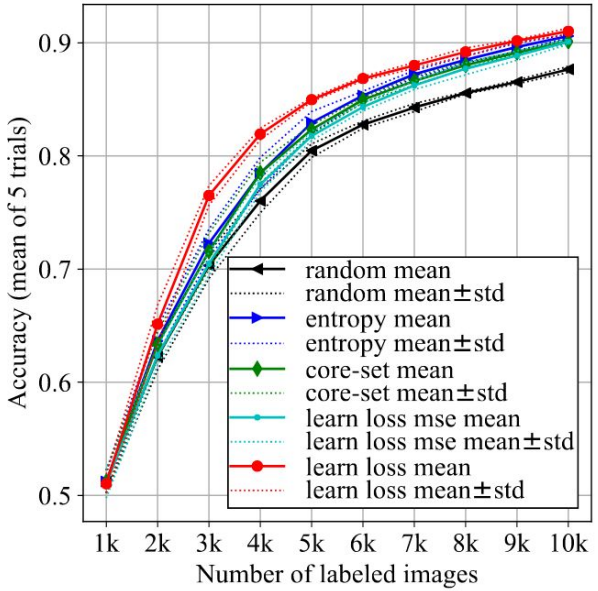
Model selection is critical

- MNIST
- Fashion MNIST
- CIFAR-10
- CIFAR-100
- LDPA
- 20 Newsgroup

High Variability Among Studies [Munjal 2020]



CIFAR 10 - VGG16 - [Sener 2017]



CIFAR 10 - ResNet-18 - [Yoo 2019]

Research Challenges

No Consensus

Active learning adds a layer of complexity over model training.

We cannot really trust the literature: experimental settings vary from one paper to the other and query strategy rankings are inconsistent.

We need need to go back to the fundamentals of active learning to build our own intuition.

Classical Active Learning Samplers

Basic Uncertainty Methods [Settles 2009]

Let us take an example with 4 classes and 3 samples.

A	B	C	D
0.1	0.2	0.3	0.4
0.0	0.05	0.45	0.5
0.15	0.2	0.2	0.45

U	M	H
0.60	0.9	0.92
0.50	**0.95**	0.62
0.55	0.75	**0.93**

Let C_1, C_2, C_3, C_4 be the probability to belong to a class given by a predictor sorted decreasingly.

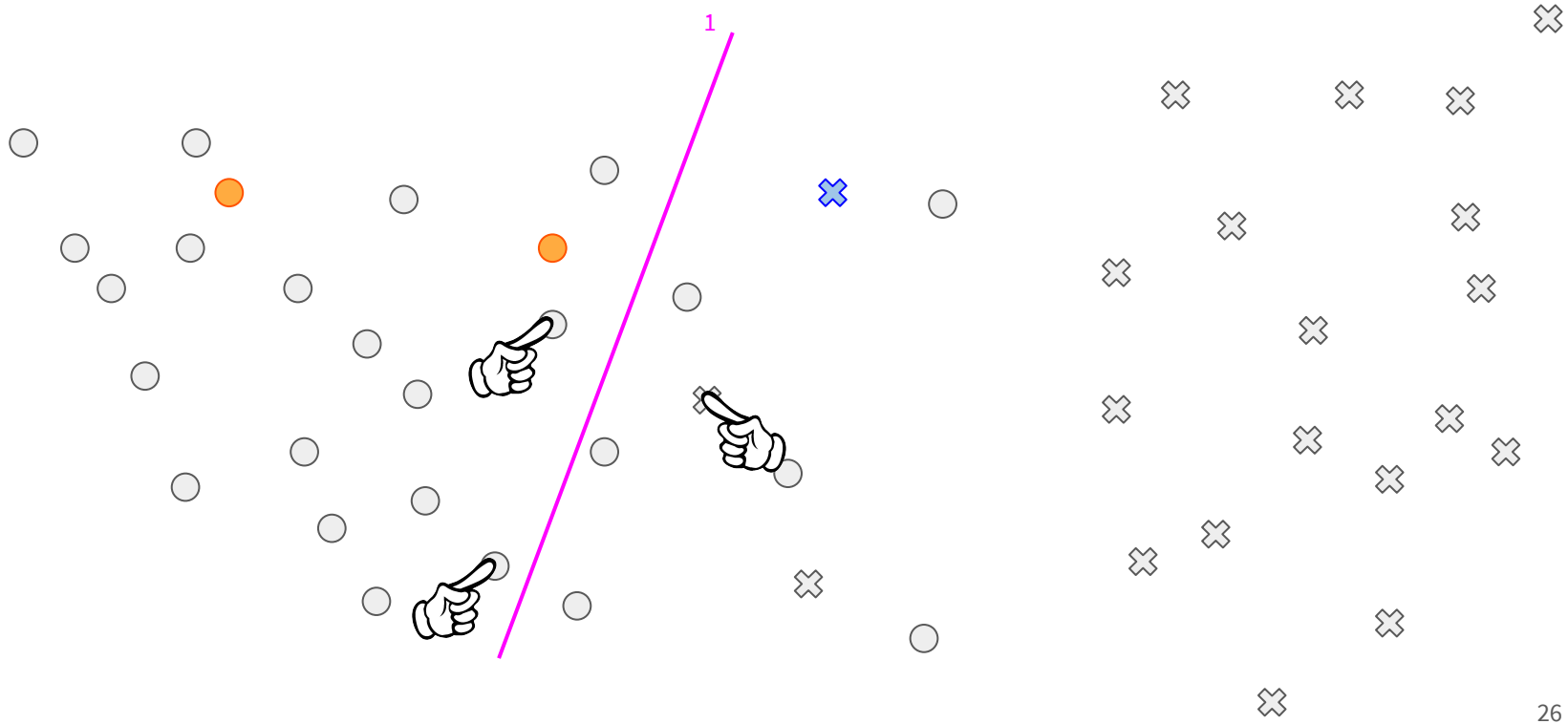
Lowest confidence: $U(x) = 1 - C_1$

Classification margin: $M(x) = 1 - (C_1 - C_2)$

Classification entropy: $H(x) = - \sum C_i \log(C_i)$

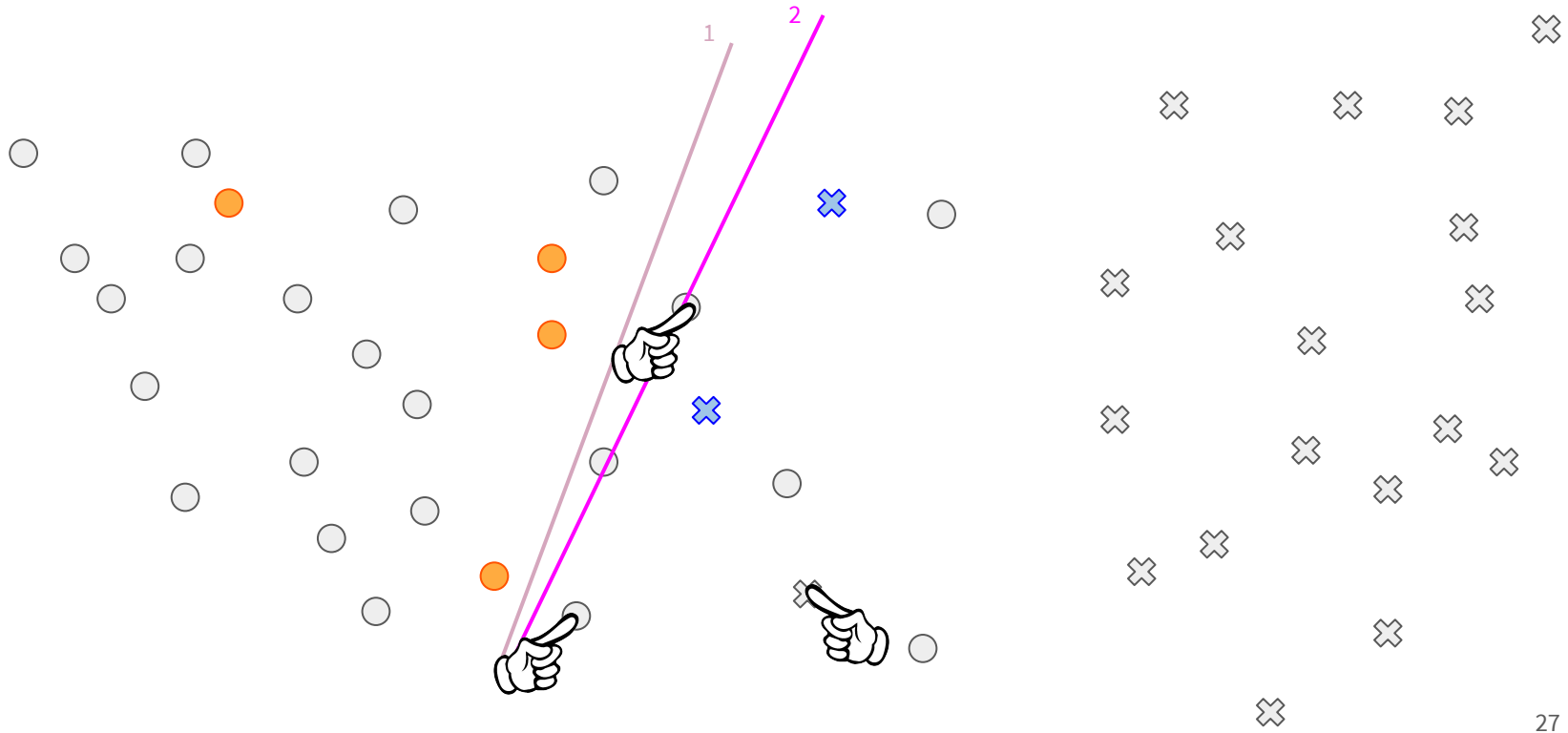
Motivation

Pushing the decision boundary



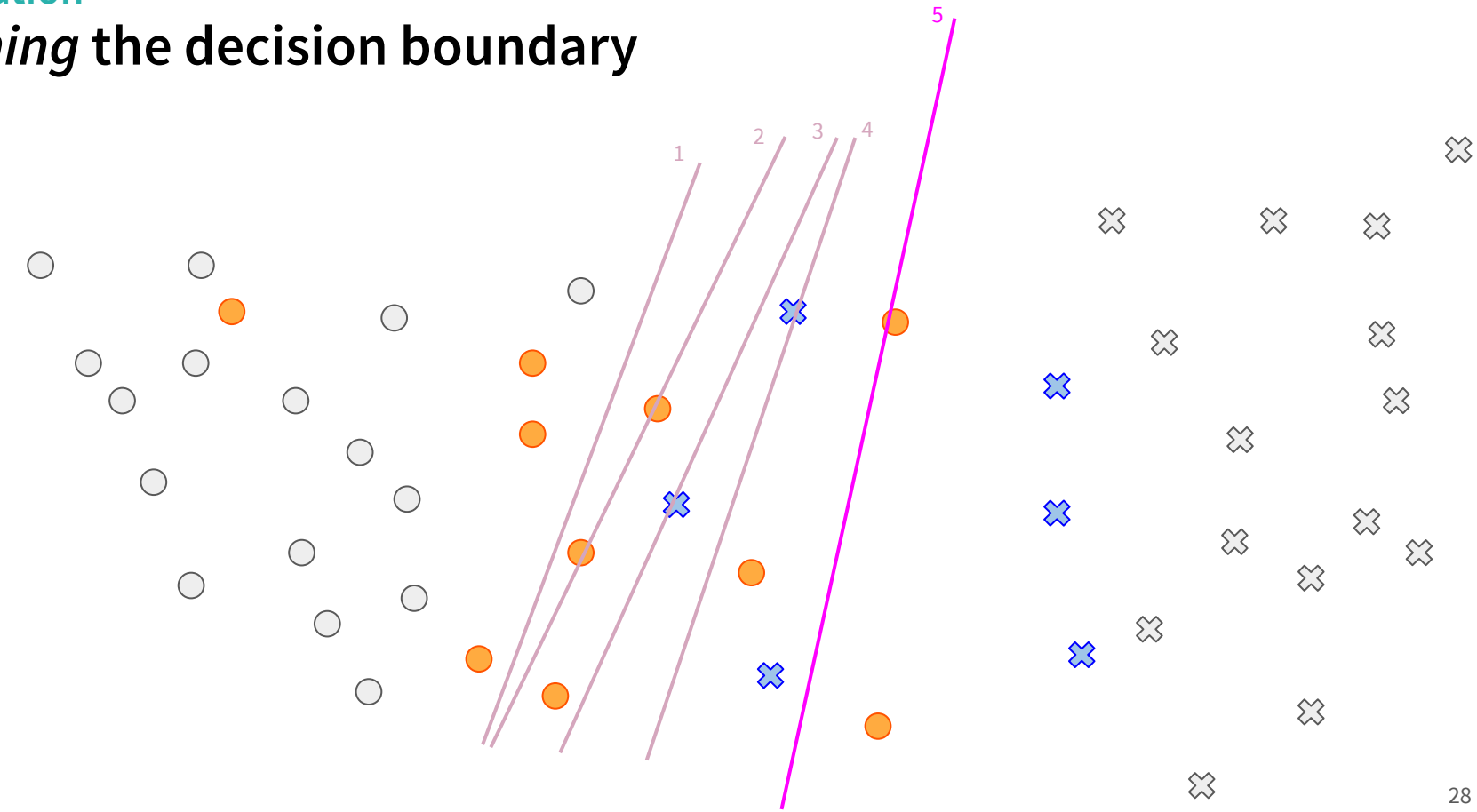
Motivation

Pushing the decision boundary



Motivation

Pushing the decision boundary



Active Learning

I get it, how do I try it now?

Various packages exists with different philosophical approaches to active learning.

Which one should I try first?

Shopping Guide to Active Learning Packages

Shopping Guide to AL Packages

The Landscape (< 2020)

modAL

- Minimalist design
- Focus on ensemble approaches
- Provides an object ActiveLearner to run experiments with in-memory caching

AliPy

- Created in Nanjing University of Aeronautics and Astronautics
- Exclusive methods: Active Learning from Data, Self-paced Active Learning
- AlExperiment object ease experiments with disk caching but no replay

Libact

- Performance oriented
- Exclusive feature: Learning active learning by learning
- Utilities provided for experiment, such as noisy labeler, but the main loop is left to the user

Learn more in our blog post

The Python Package cardinal

Dataiku's Solution: Cardinal

A package focused on **experiments** and **metrics**

- Tried and tested query strategies and metrics
- Numerous and detailed examples and experiments
- Ease research in active learning metrics by allowing experiment replay
- Human readable cache

So far in the package:

- Classical query strategies
- Two-step query strategy from [Zhdanov 2019]
- Introduction to Active Learning and examples displaying the importance of exploration

The Python Package cardinal

Query strategy interface

BaseQuerySampler

```
init(batch_size)
fit(X, y=None)
select_samples(X)
```

ScoredQuerySampler

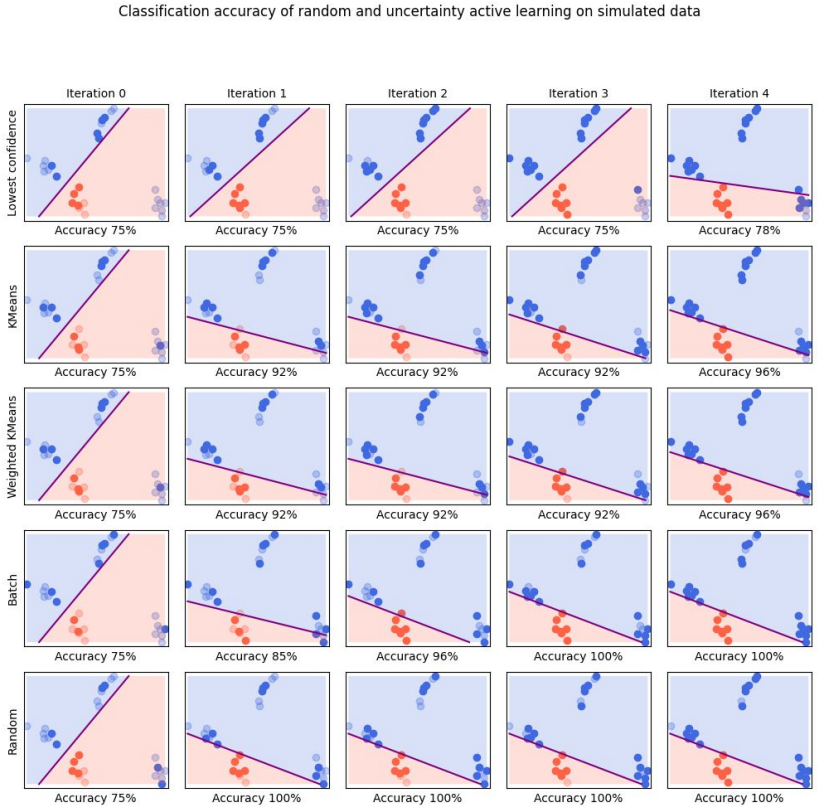
```
score_samples(X)
sample_scores_
```

MarginSampler
UncertaintySampler
EntropySampler

KCentroidSampler

```
init(batch_size)
score_samples(X, sample_weight)
```

KMeansSampler
MiniBatchKMeansSampler



ElementAI's BAAL

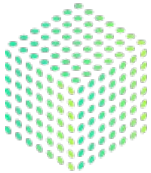
Baal's philosophy:

- Focus on Bayesian methods
- Wrappers and automated processing everywhere
  - Datasets
  - Samplers
  - Active learning loop

```
dataset = ActiveLearningDataset(your_dataset)
dataset.label_randomly(INITIAL_POOL)  # label some data
model = MCDropoutModule(your_model)
model = ModelWrapper(model, your_criterion)
active_loop = ActiveLearningLoop(dataset,
                                 get_probabilities=model.predict_on_dataset,
                                 heuristic=heuristics.BALD(shuffle_prop=0.1),
                                 ndata_to_label=NDATA_TO_LABEL)
for al_step in range(N_ALSTEP):
    model.train_on_dataset(dataset, optimizer, BATCH_SIZE, use_cuda=use_cuda)
    if not active_loop.step():
        # We're done!
        break
```

After 2020

Scikit-activeml and DISTIL



DISTIL

Deep dIverSified inTeractIve Lear

Research-oriented
Latest research strategies
Early stage of development

Production oriented
Latest fastest tried and test strategies
Ready to go to production

Goal of the tutorial

Learn the bases of Active Learning: how it works, how to apply it to an experiment, how to get a critical look on results.

Get to know the latest trends and newest methods along with their available implementations.

Exclusive: Discover how to get live insights about active learning experiments and make it easier to apply them in the real world.

But not:

- How to handle noisy oracles / multiple annotations
- Methods on complex use cases (object detection, named entity recognition)
- Methods specific to some models
- Varying labeling costs
- Stream / Pool active learning

Notebook Exercise

Hands-on

Check Out Notebook_1 in the repo

dataiku-research / active-learning-tutorial

Load and preprocess datasets.

Visualize datasets using your favorite visualization method (tSNE, UMAP, etc.).

Create your first active learning loop! Explore the caching and easy indexing capacities of cardinal.

Plot your results and analyze them.

Optional. Load your own dataset or choose one in OpenML for the rest of the tutorial

Part 2 - Advanced Query Strategies

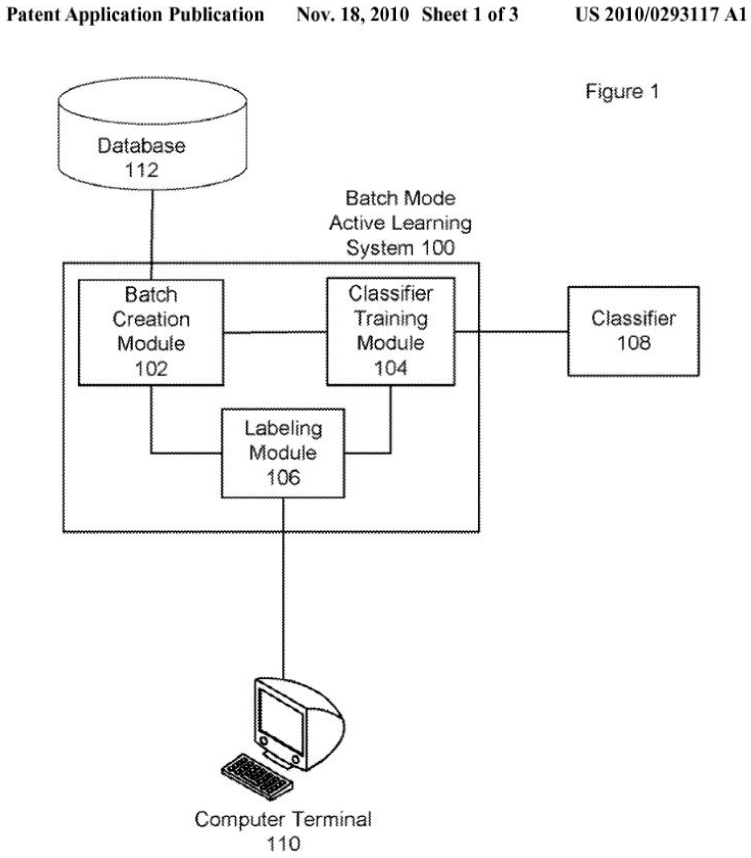
Limitations of Classical AL Samplers

Single sample vs Batch

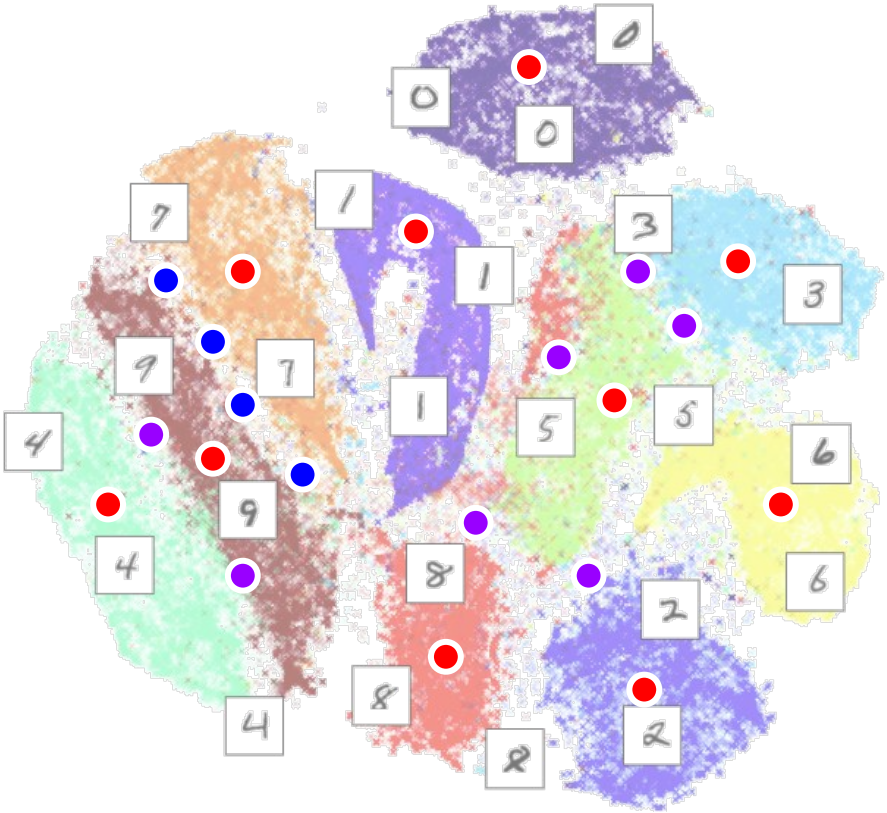
In real life, annotating samples one by one is not realistic. This is knows since Xu 2007, Settles 2011.

As with DNN, model performance depends on batch size and composition.

How to choose the best batch size and the best combination of samples to compose a batch?



Building intuition on MNIST



REPRESENTATIVENESS

Choose the most common points, explore the space

INFORMATIVENESS

Focus on hard to classify samples

DIVERSITY

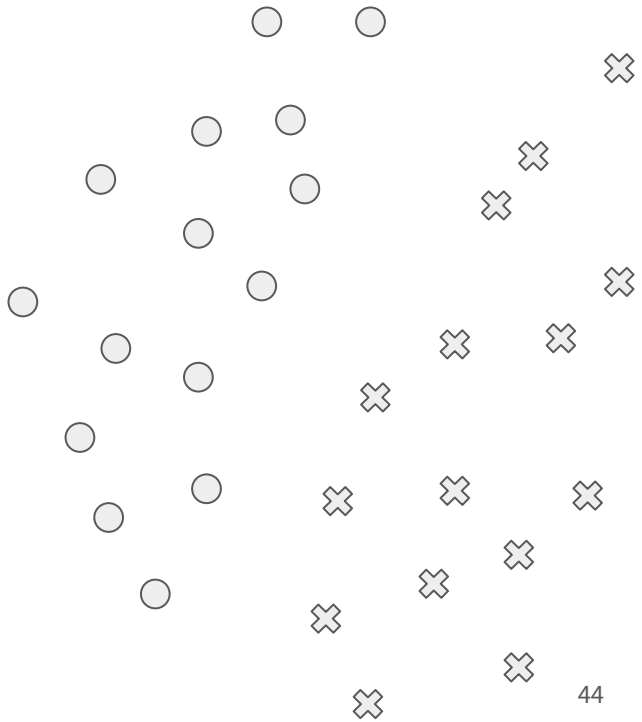
Force the selection to explore diverse examples

Diverse mini-batch Active Learning (2019)

Incremental KMeans

Principle of WKMeans [Zhdanov 2019]

Weighted KMeans proceeds as follows:



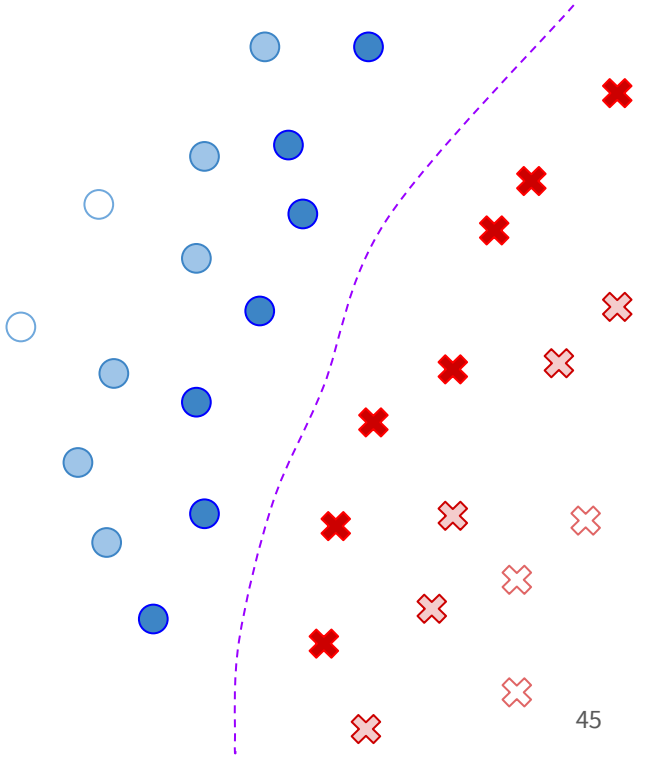
Zhdanov, Fedor. "Diverse mini-batch active learning." arXiv preprint arXiv:1901.05954 (2019).

Incremental KMeans

Principle of WKMeans

Weighted KMeans proceeds as follows:

1. Computation of uncertainty score

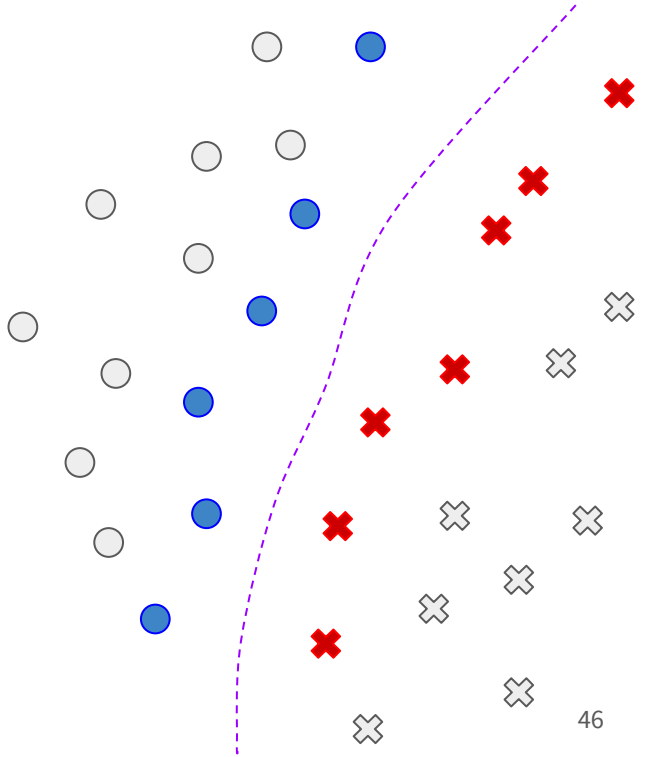


Incremental KMeans

Principle of WKMeans

Weighted KMeans proceeds as follows:

1. Computation of uncertainty score
2. Preselect a β * *batch size* samples

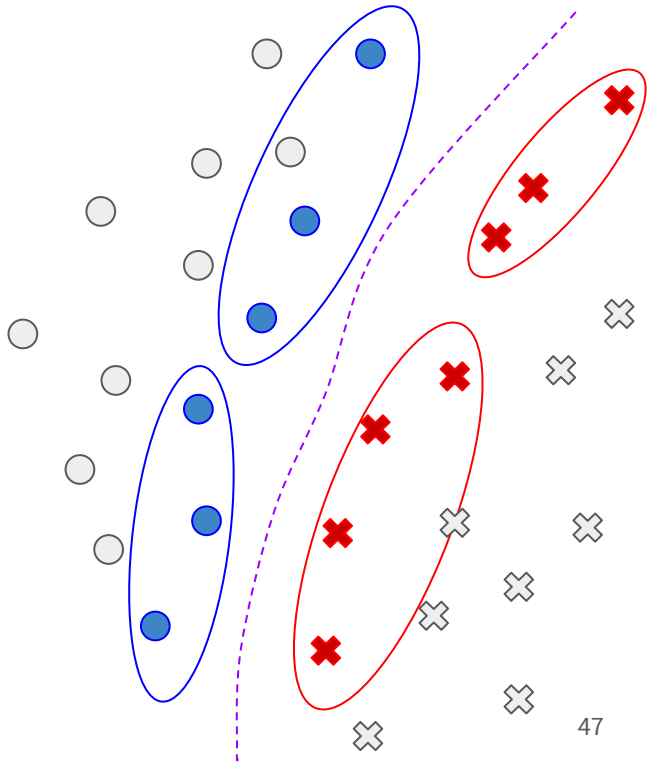


Incremental KMeans

Principle of WKMeans

Weighted KMeans proceeds as follows:

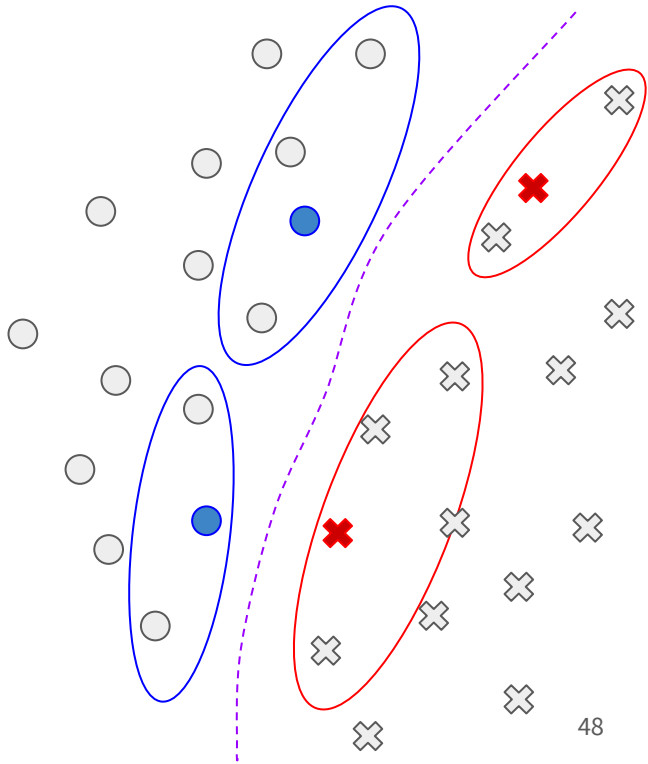
1. Computation of uncertainty score
2. Preselect a β * *batch size* samples
3. Among the preselected samples, run a KMeans with *batch size* centroids



Principle of WKMeans

Weighted KMeans proceeds as follows:

1. Computation of uncertainty score
2. Preselect a β * *batch size* samples
3. Among the preselected samples, run a KMeans with *batch size* centroids
4. Select the samples nearest to the centroids

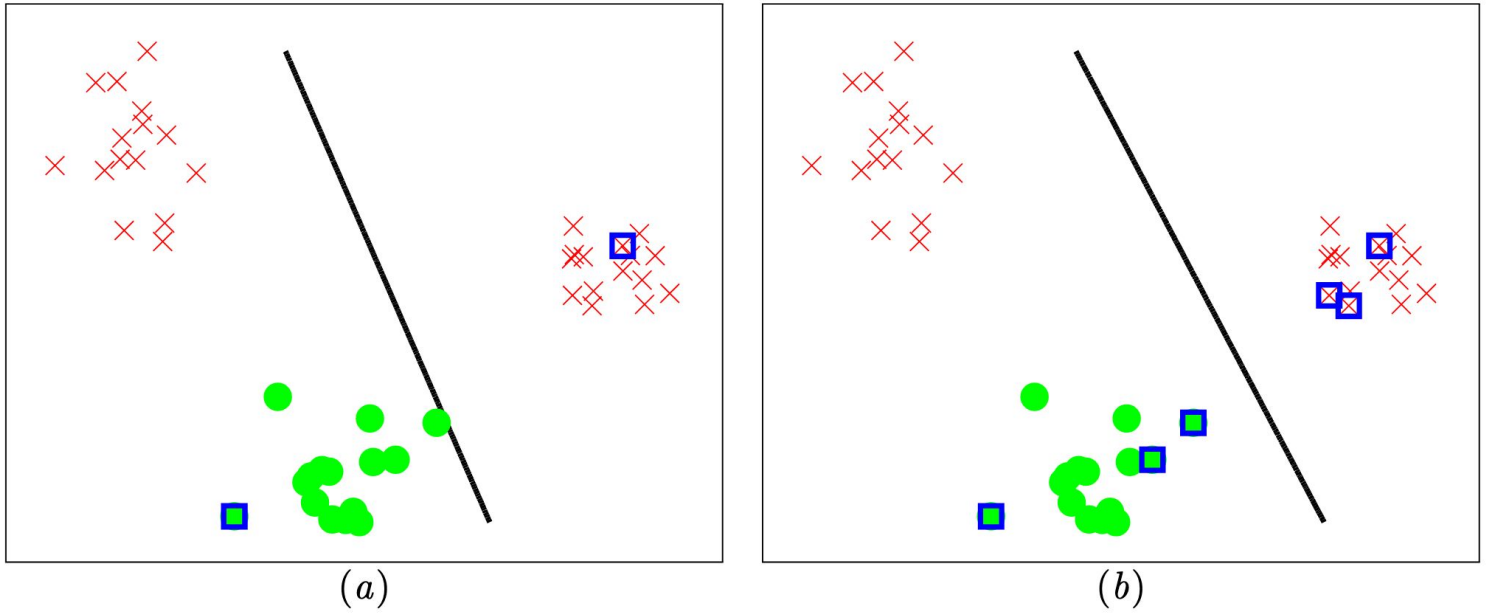


Query sampling strategy review

Li et al. 2012

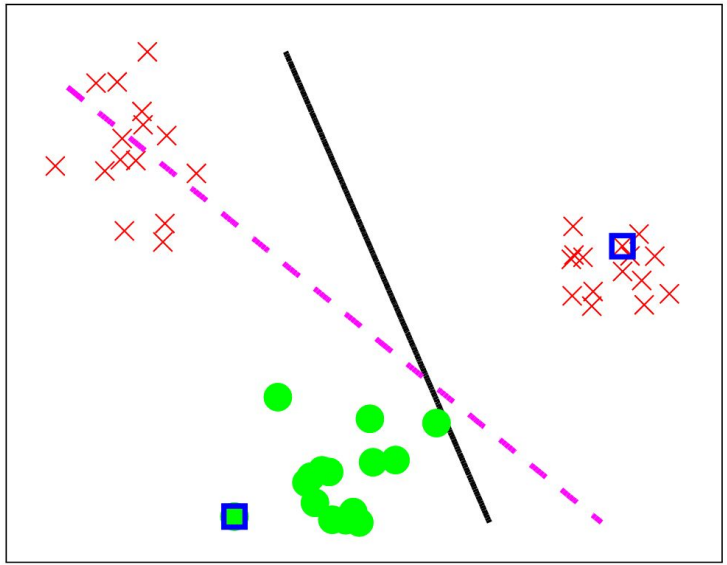
Hint SVM

SVM based uncertainty can ignore a large set of unlabeled samples.

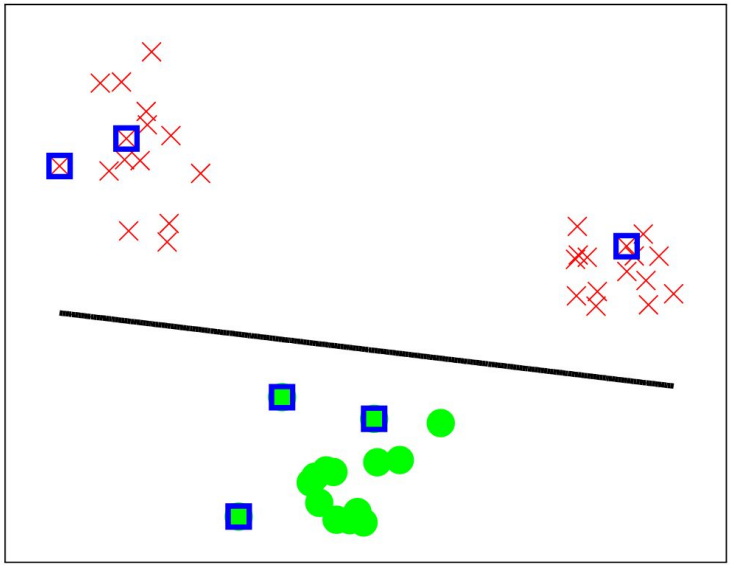


Li et al. 2012

Hint SVM



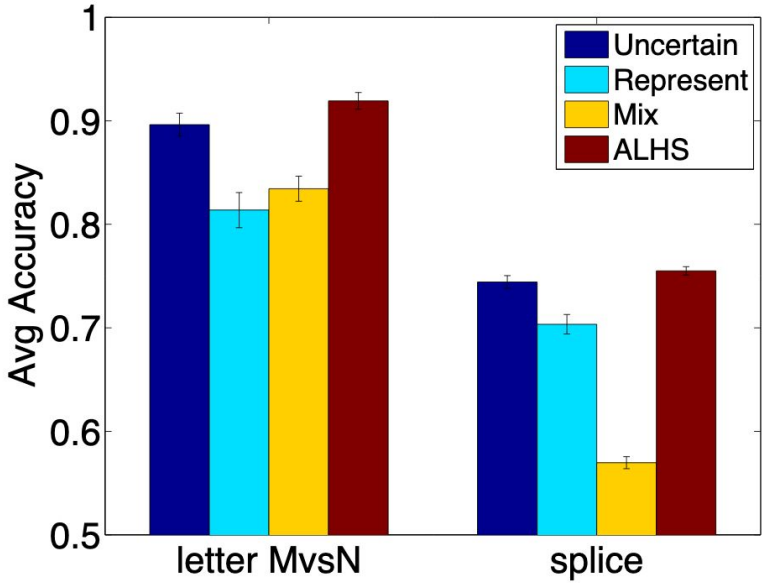
(a)

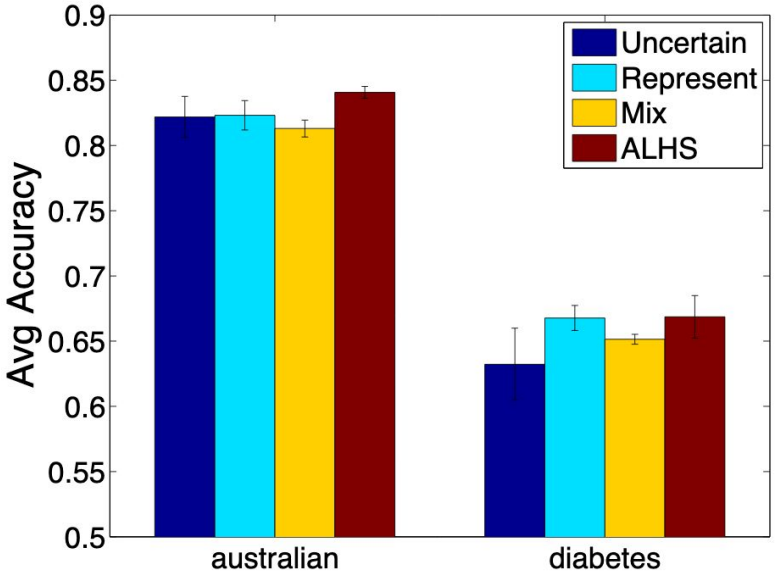


(b)

Li et al. 2012

Hint SVM





Li et al. 2012

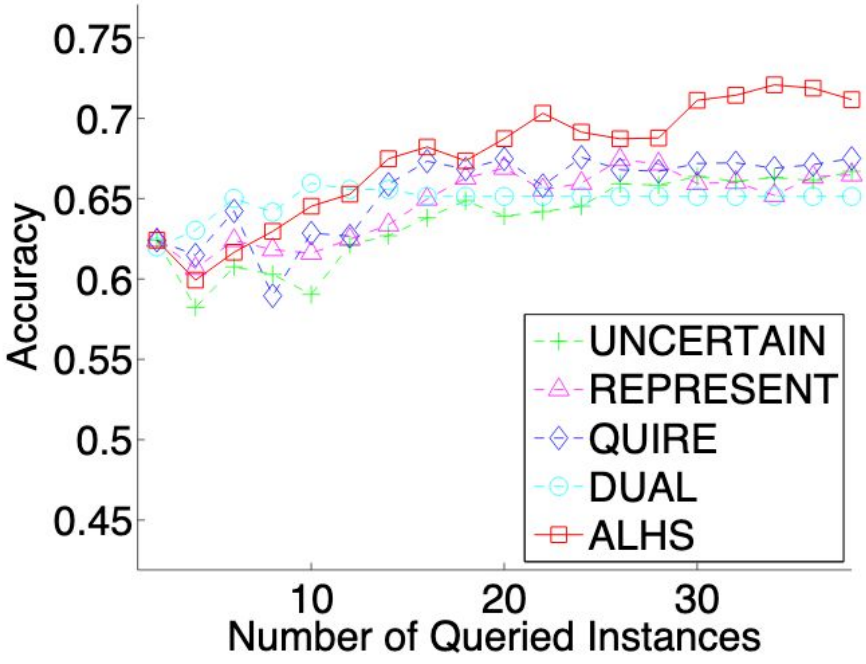
Hint SVM

Table 1: Comparison on accuracy (mean ± se) after querying 5% of unlabeled pool

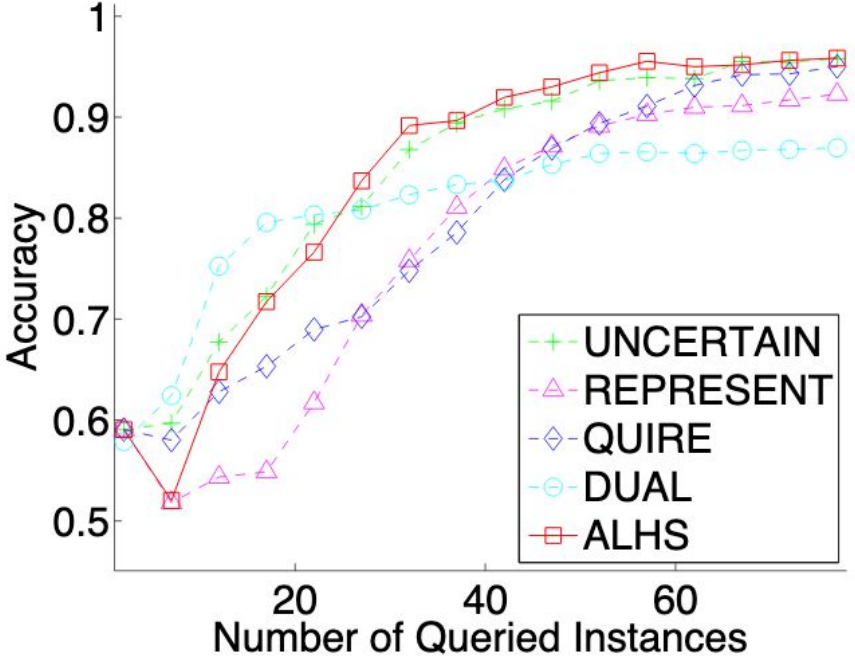
	Algorithms (%), the highest accuracy for each dataset is in boldface				
data	UNCERTAIN	REPRESENT	QUIRE	DUAL	ALHS
australian	82.188 ± 1.571	83.739 ± 0.548	82.319 ± 1.126	81.304 ± 0.647	**84.072 ± 0.454**
breast	96.334 ± 0.278	95.264 ± 0.439	**96.657 ± 0.187**	96.408 ± 0.196	96.525 ± 0.219
diabetes	63.229 ± 2.767	66.758 ± 0.505	66.771 ± 0.960	65.143 ± 0.381	**66.862 ± 1.632**
german	69.060 ± 0.497	67.240 ± 1.099	68.750 ± 0.605	69.620 ± 0.323	**69.750 ± 0.349**
letterMvsN	89.632 ± 1.103	83.463 ± 1.348	81.372 ± 1.693	83.437 ± 1.211	**91.919 ± 0.812**
letterVvsY	79.245 ± 1.176	63.523 ± 2.335	68.516 ± 2.132	76.213 ± 1.549	**79.381 ± 1.174**
segment	95.437 ± 0.367	94.390 ± 0.482	96.074 ± 0.224	86.078 ± 2.834	**96.095 ± 0.204**
splice	74.430 ± 0.606	69.117 ± 1.452	70.340 ± 0.942	56.969 ± 0.576	**75.506 ± 0.403**
wdbc	93.842 ± 3.137	95.616 ± 0.711	96.613 ± 0.230	96.056 ± 0.250	**96.921 ± 0.200**

Li et al. 2012

Hint SVM



(c) *diabetes*



(d) *leterMvsN*

Du et al. 2019

Exploring Representativeness and Informativeness for Active Learning

$$M_1(i,j) = \frac{1}{2}S(i,j) \tag{6}$$

$M_1(i,j)$ is the similarity between the i^{th} and j^{th} sample in unlabeled set. However, differing from the entry in M_1, the entry in M_2 measures the distribution between one sample and the labeled set. The formulation can be written as:

$$M_2(i) = \frac{n_t + 1}{n} \sum_{j=1}^{n_t} S(i,j) \tag{7}$$

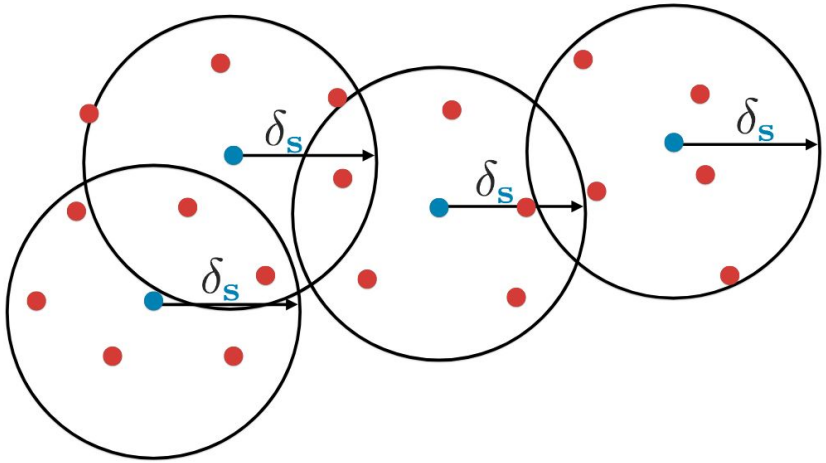
$$M_3(i) = \frac{u_t - 1}{n} \sum_{j=1}^{u_t} S(i,j)$$

Global optimization, with C being uncertainty score

$$\min_{\alpha} \alpha^T M_1 \alpha + \alpha^T (M_2 - M_3) + \beta C$$
$$s.t. \quad \alpha^T 1^{u_t} = 1, \alpha_i \in [0,1]$$

Elhamifar et al. 2018

Active Learning for Convolutional Neural Networks: A Core-Set Approach



Algorithm 1 k-Center-Greedy

Input: data $\mathbf{x}_i$, existing pool $\mathbf{s}^0$ and a budget b
Initialize $\mathbf{s} = \mathbf{s}^0$
repeat
  $u = \arg\max_{i \in [n] \setminus \mathbf{s}} \min_{j \in \mathbf{s}} \Delta(\mathbf{x}_i, \mathbf{x}_j)$
  $\mathbf{s} = \mathbf{s} \cup \{u\}$
until $|\mathbf{s}| = b + |\mathbf{s}^0|$
return $\mathbf{s} \setminus \mathbf{s}^0$

Elhamifar et al. 2018

Active Learning for Convolutional Neural Networks: A Core-Set Approach

Algorithm 2 Robust k-Center

Input: data $\mathbf{x}_i$, existing pool $\mathbf{s}^0$, budget b and outlier bound Ξ
Initialize $\mathbf{s}_g = \text{k-Center-Greedy}(\mathbf{x}_i, \mathbf{s}^0, b)$
$\delta_{2-OPT} = \max_j \min_{i \in \mathbf{s}_g} \Delta(\mathbf{x}_i, \mathbf{x}_j)$
$lb = \frac{\delta_{2-OPT}}{2}$, $ub = \delta_{2-OPT}$
repeat
  if $Feasible(b, \mathbf{s}^0, \frac{lb+ub}{2}, \Xi)$ **then**
    $ub = \max_{i,j|\Delta(\mathbf{x}_i,\mathbf{x}_j) \leq \frac{lb+ub}{2}} \Delta(\mathbf{x}_i, \mathbf{x}_j)$
  else
    $lb = \min_{i,j|\Delta(\mathbf{x}_i,\mathbf{x}_j) \geq \frac{lb+ub}{2}} \Delta(\mathbf{x}_i, \mathbf{x}_j)$
  end if
until $ub = lb$
return $\{i \; s.t. \; u_i = 1\}$

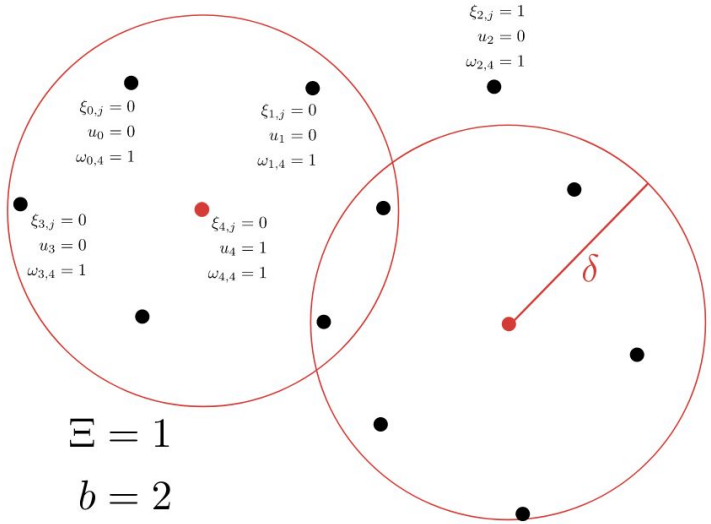


Figure 2: Visualizations of the variables. In this solution, the 4^{th} node is chosen as a center and nodes $0, 1, 3$ are in a δ ball around it. The 2^{nd} node is marked as an outlier.

Elhamifar et al. 2018

Active Learning for Convolutional Neural Networks: A Core-Set Approach

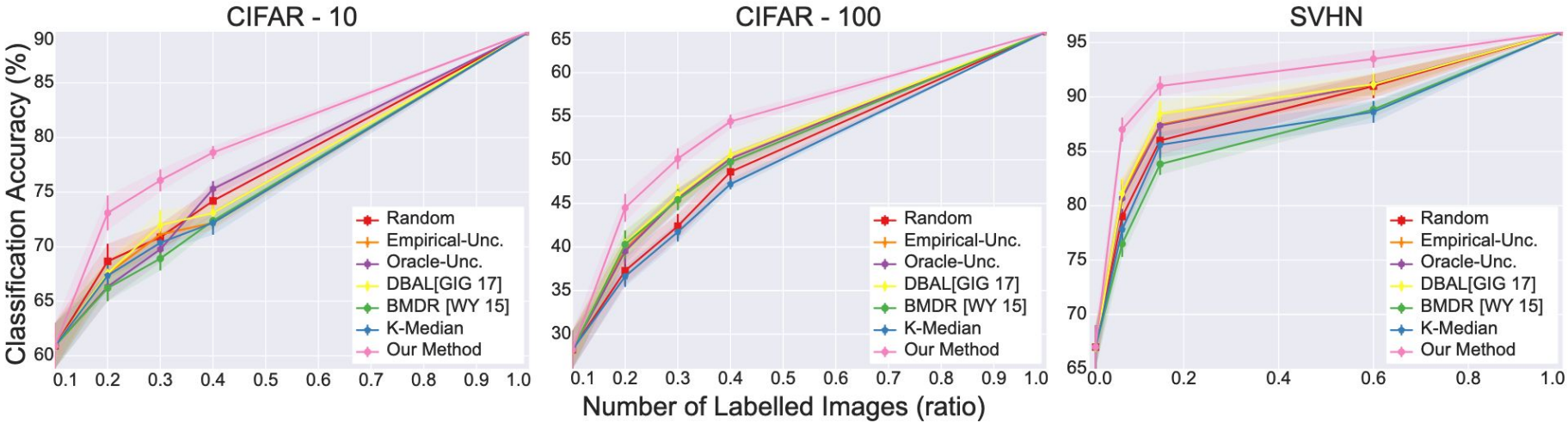
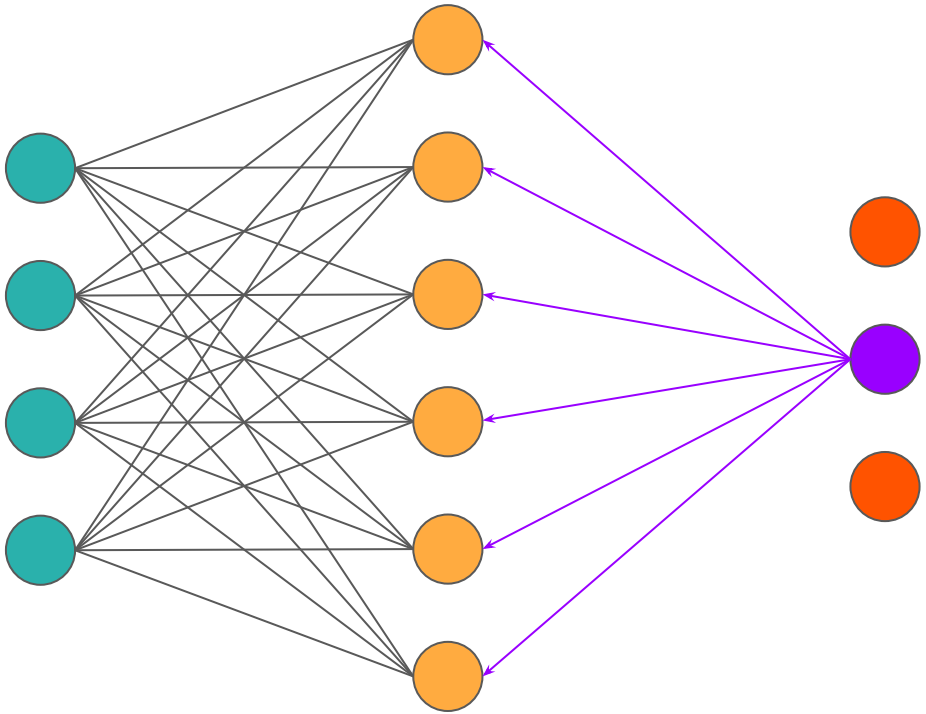


Figure 4: Results on Active Learning for Fully-Supervised Model (error bars are std-dev)

Ash et al. 2019

Deep batch active learning by diverse, uncertain gradient lower bounds



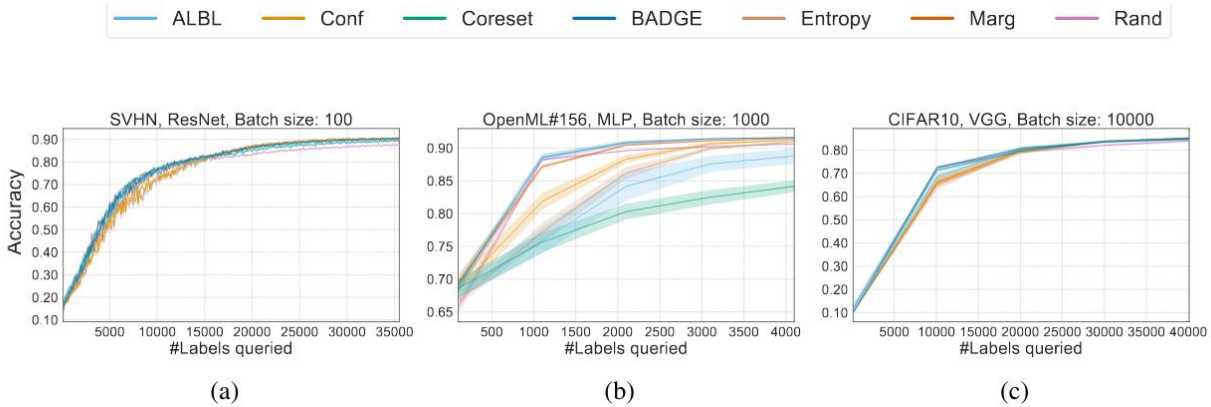
[Ash et al. 2019]

Deep batch active learning by diverse, uncertain gradient lower bounds

Algorithm 1 BADGE: Batch Active learning by Diverse Gradient Embeddings

Require: Neural network $f(x;\theta)$, unlabeled pool of examples U, initial number of examples M, number of iterations T, number of examples in a batch B.

1: Labeled dataset $S \leftarrow M$ examples drawn uniformly at random from U together with queried labels.
2: Train an initial model θ_1 on S by minimizing $\mathbb{E}_S[\ell_{\text{CE}}(f(x;\theta), y)]$.
3: **for** $t = 1, 2, \ldots, T$: **do**
4: For all examples x in $U \setminus S$:
1. Compute its hypothetical label $\hat{y}(x) = h_{\theta_t}(x)$.
2. Compute gradient embedding $g_x = \frac{\partial}{\partial \theta_{\text{out}}} \ell_{\text{CE}}(f(x;\theta), \hat{y}(x))|_{\theta=\theta_t}$, where θ_{out} refers to parameters of the final (output) layer.
5: Compute S_t, a random subset of $U \setminus S$, using the k-MEANS++ seeding algorithm on $\{g_x : x \in U \setminus S\}$ and query for their labels.
6: $S \leftarrow S \cup S_t$.
7: Train a model θ_{t+1} on S by minimizing $\mathbb{E}_S[\ell_{\text{CE}}(f(x;\theta), y)]$.
8: **end for**
9: **return** Final model θ_{T+1}.



Notebook Exercise
Hell of a notebook!

ElementAI's BAAL

Bayesian Active Learning

At its core, Bayesian learning puts a prior belief $q(\theta)$ on the posterior distribution of a model $p(\theta \mid \mathcal{D})$ - Andrew Gordon Wilson.

A useful notion is the Bayesian model averaging (BMA)

$$p(y \mid x, \mathcal{D}) = \int p(y \mid x, \theta) p(\theta \mid D) d\theta$$

ElementAI's BAAL

Bayesian Active Learning

Of course, the integral is intractable so we must approximate it. Here are some examples:

MC-Dropout[1]

By keeping Dropout on at test time, we can sample from the posterior distribution.

(Multi)SWA(G)[2]

Create an ensemble by averaging the predictions of multiple local minima gathered through the training loop.

Ensembles

Train multiple models on multiple subsets of the training data to get multiple point estimation of the posterior distribution. This can be done on shallow models as well.

Hands on

Benchmarking methods

The next exercise put you in control of the benchmark! You are free to add methods coming from other package to the benchmark and see their performance.

The devil's advocate. Want to see how hard it is to get your results published? Follow this tutorial:

- If the method you sell has no diversity, decrease the batch size
- If your method mainly use data distribution and not the model, change the model for a weaker one.

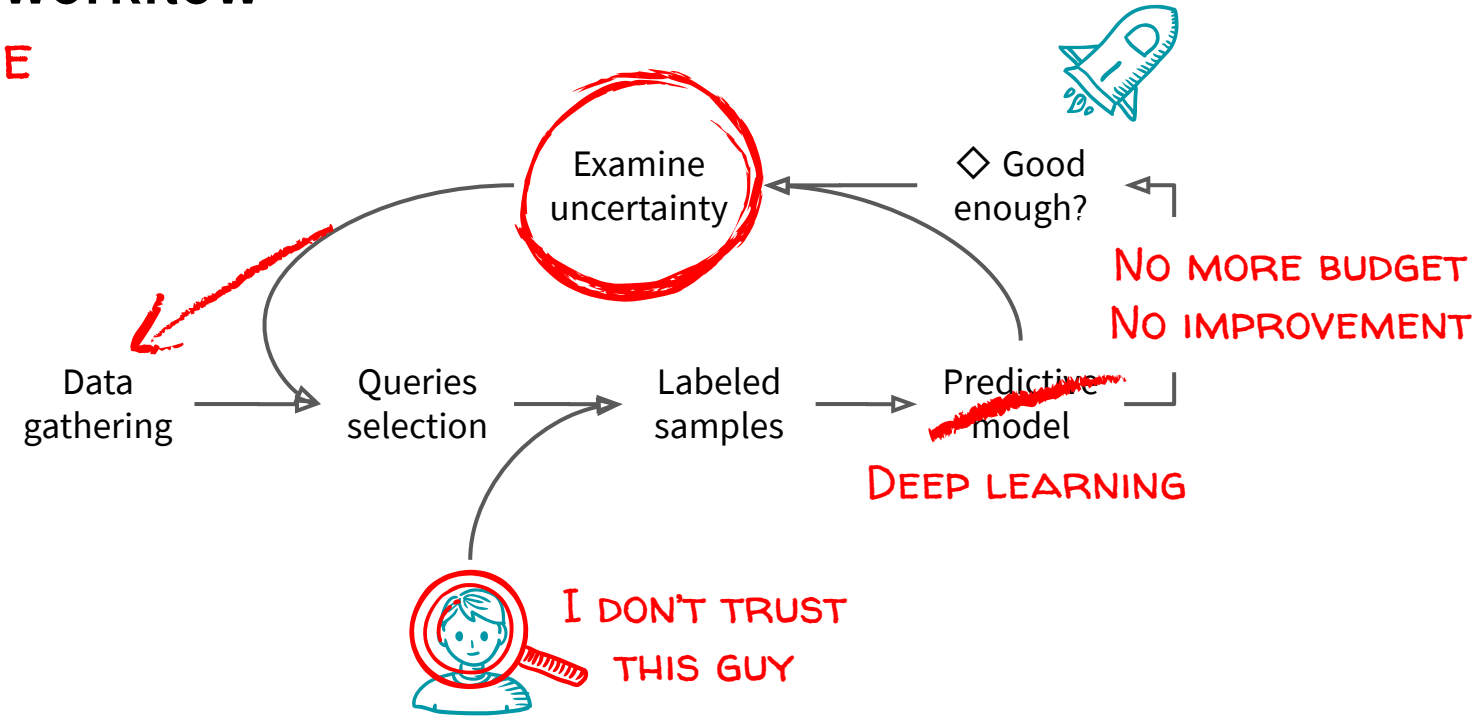
Part 3 - Active Learning in Real Life

Practical Challenges of Active Learning

Why are we better than others?

~~Typical~~ workflow

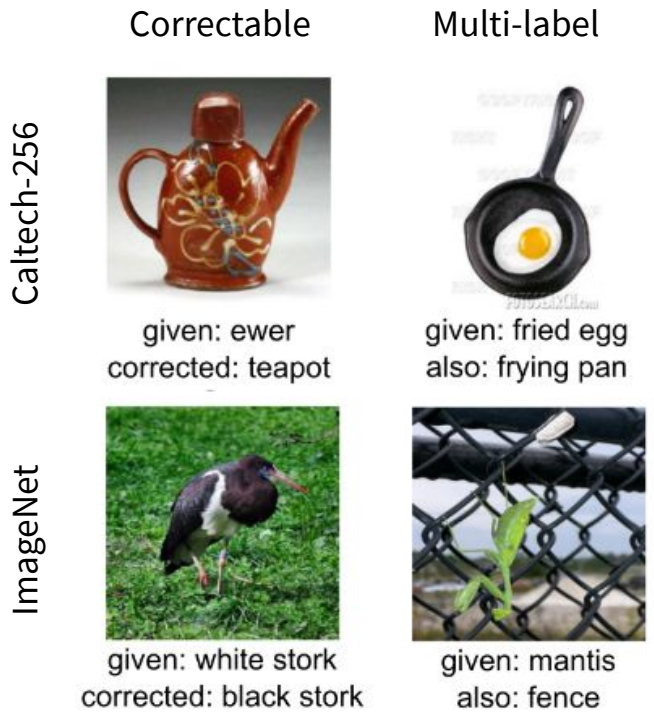
REAL LIFE



Motivation

Errors in test labels impact classifier performance [Northcutt 2021]

On ImageNet, 5.83% of errors have been identified on the test set. Fixing them greatly impact model ranking on this task.



Northcutt, Curtis G., Anish Athalye, and Jonas Mueller. "Pervasive label errors in test sets destabilize machine learning benchmarks." *arXiv preprint arXiv:2103.14749* (2021).

Motivation

Errors in test labels impact classifier performance [Northcutt 2021]

On ImageNet, 5.83% of errors have been identified on the test set (below). Fixing them greatly impact model ranking on this task (right).

Table 1: Test set errors are prominent across common benchmark datasets. Errors are estimated using confident learning (CL) and validated by human workers on Mechanical Turk.

Dataset	Modality	Size	Model	Test Set Errors				
				CL guessed	MTurk checked	validated	estimated	% error
MNIST	image	10,000	2-conv CNN	100	100 (100%)	15	-	0.15
CIFAR-10	image	10,000	VGG	275	275 (100%)	54	-	0.54
CIFAR-100	image	10,000	VGG	2235	2235 (100%)	585	-	5.85
Caltech-256	image	30,607	ResNet-152	4,643	400 (8.6%)	65	754	2.46
ImageNet*	image	50,000	ResNet-50	5,440	5,440 (100%)	2,916	-	5.83
QuickDraw	image	50,426,266	VGG	6,825,383	2,500 (0.04%)	1870	5,105,386	10.12
20news	text	7,532	TFIDF + SGD	93	93 (100%)	82	-	1.11
IMDB	text	25,000	FastText	1,310	1,310 (100%)	725	-	2.9
Amazon	text	9,996,437	FastText	533,249	1,000 (0.2%)	732	390,338	3.9
AudioSet	audio	20,371	VGG	307	307 (100%)	275	-	1.35

*Because the ImageNet test set labels are not publicly available, the ILSVRC 2012 validation set is used.

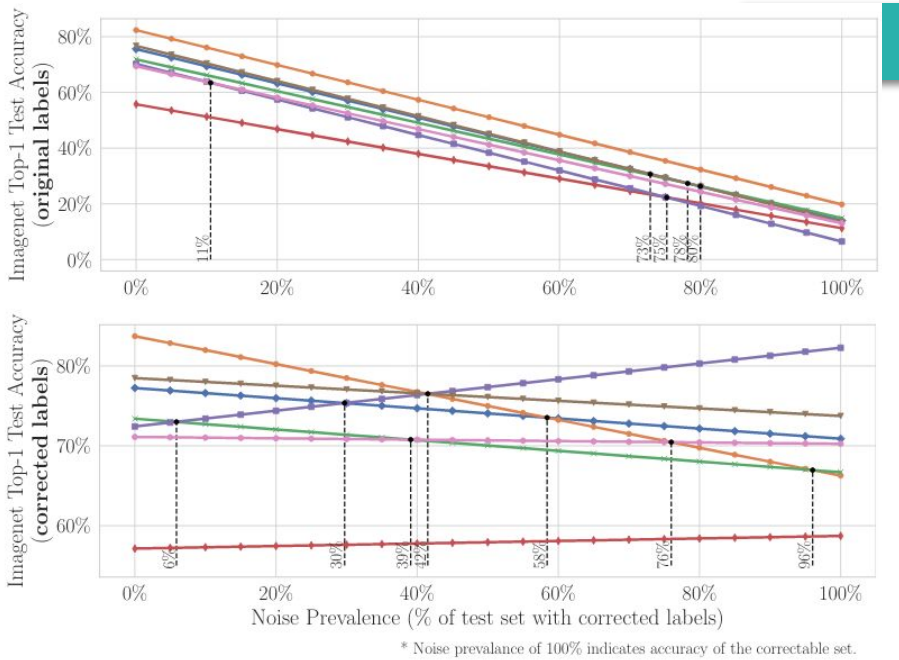


Figure 4: ImageNet top-1 original accuracy (top panel) and corrected accuracy (bottom panel) vs Noise Prevalence (with agreement threshold = 3). Vertical lines indicate noise levels at which the ranking of two models changes (in terms of original/corrected accuracy).

Northcutt, Curtis G., Anish Athalye, and Jonas Mueller. "Pervasive label errors in test sets destabilize machine learning benchmarks." *arXiv preprint arXiv:2103.14749* (2021).

Motivation

Sample selection for data cleaning

- Propublica monitors donations made by laboratories to doctors
- The status field is dirty and can lead to misclassification
- Active cleaning improves the model by removing dirty data

ProPublica collected a dataset of corporate donations to medical researchers to analyze conflicts of interest [2]. For reference, the dataset has the following schema:

```
Contribution(
 pi_specialty text, # PI's medical specialty
 drug_nm text ,   # drug brand name, null if not drug
 device_nm text,  # device brand name, null if not a device
 corp text,       # name of pharamceutical donor
 amount float,    # amount donated
 dispute bool,    # whether the research is disputed
 status text      # if the donation is allowed
                  # under research protocol
)
```

Figure 1: (a) Systematic corruption in one variable can lead to a shifted model. The dirty examples are labeled 1-5 and the cleaned examples are labeled 1'-5'. (b) Mixed dirty and clean data results in a less accurate model than no cleaning. (c) Small samples of only clean data can result in similarly issues.

Krishnan, Sanjay, et al. "Activeclean: An interactive data cleaning framework for modern machine learning." *Proceedings of the 2016 ICMD*. 2016.

Observations on Active Learning

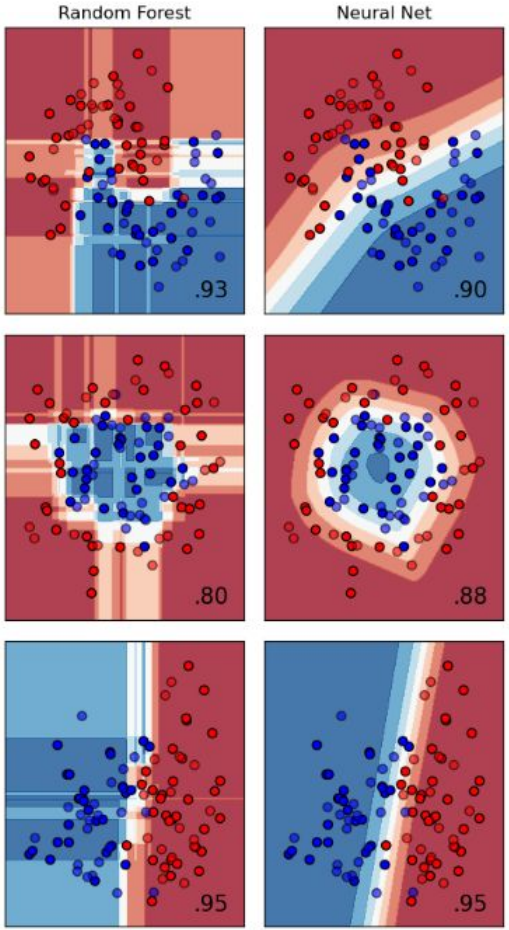
Hard to classify samples [Abraham 2020]

What has been observed

- Samplers who select too many hard-to-classify samples underperform
- Samplers who select too few hard-to-classify samples underperform
- Imposing diversity naturally tends to a good compromise

However:

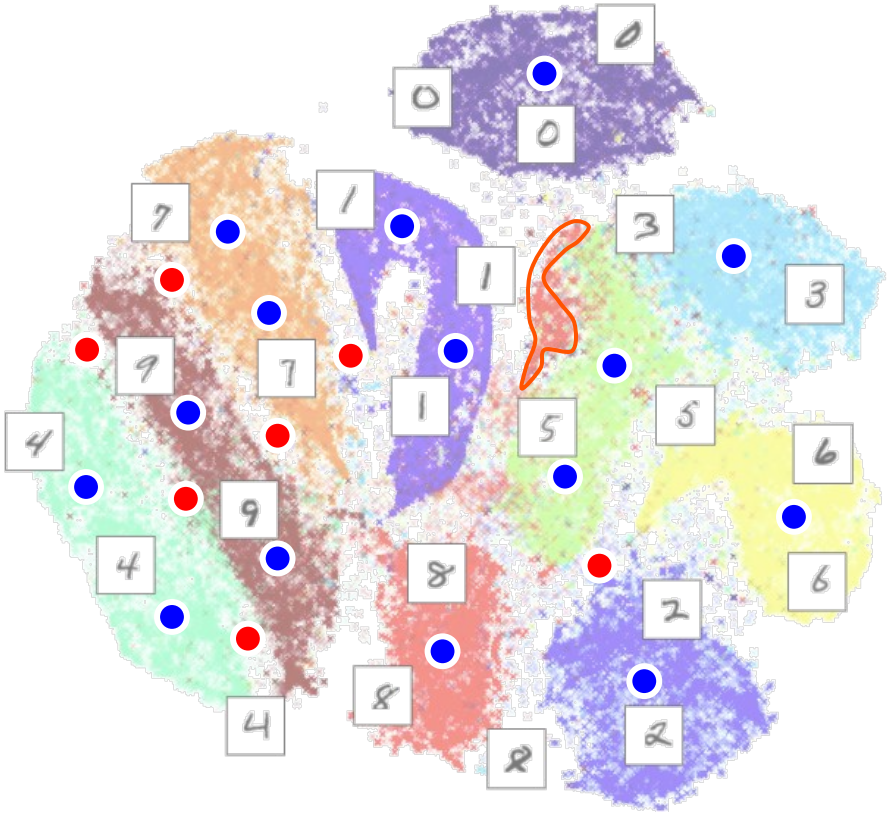
- It was an observational study with hard labels
- We want to take this into consideration in our query strategies



Classifier comparison from sklearn examples

What are noisy samples?

Building intuition on MNIST



Prototypes

Samples representative of the class

Noisy samples

Hard to classify samples

What are noisy samples?

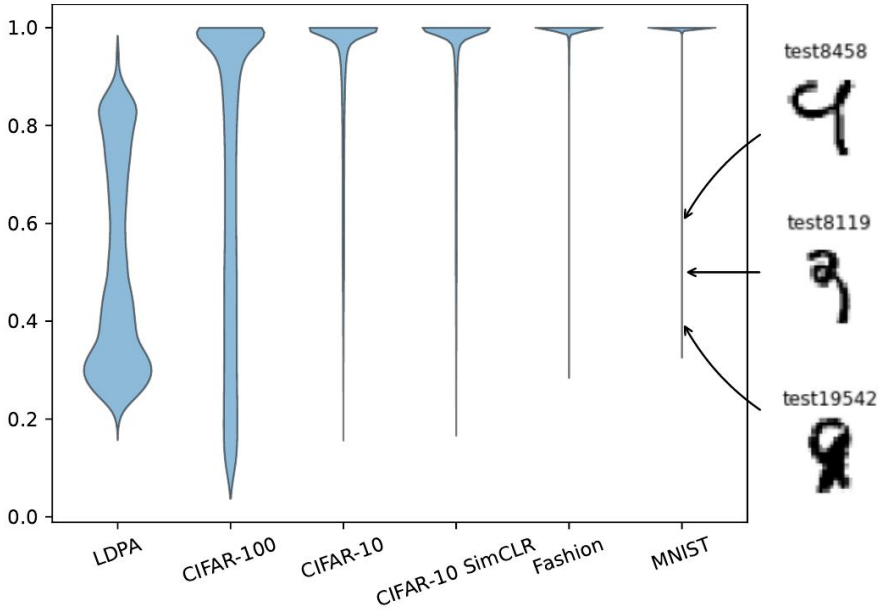
Characterizing noisy samples

Samples close to the decision boundary. Low predicted probability on their class for a *good enough* classifier h^{∞}.

$$\text{uncertainty}(x) = 1 - h^{\infty}(x)$$

Causes:

- Aleatoric noise
- Ambiguous sample
- Corrupted data



Distribution of prediction probabilities by a good model

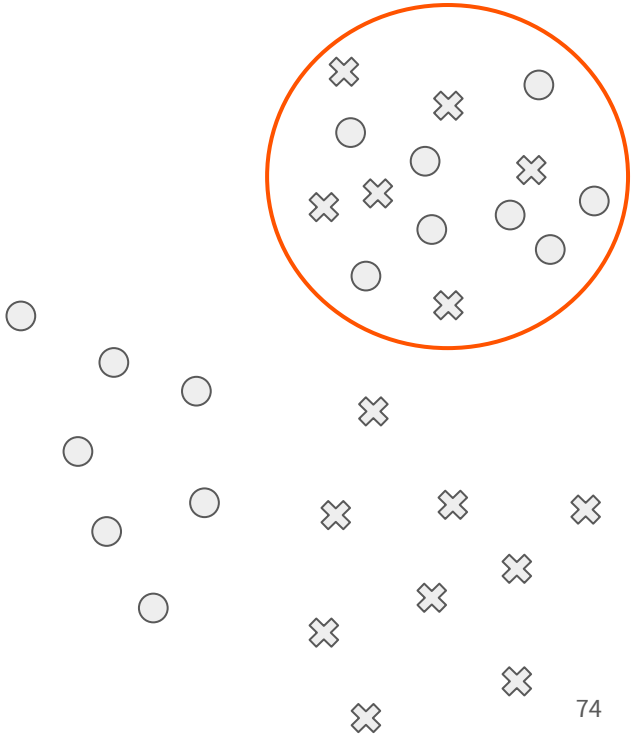
Incremental KMeans

Issues with uncertainty-based approaches

Areas containing noisy samples can act as honey pots and samples will be selected in it at each turn.

Proposed solution: Having selected samples act as repellers for selected samples.

Introducing Incremental KMeans.



Incremental KMeans

Principle of Incremental KMeans

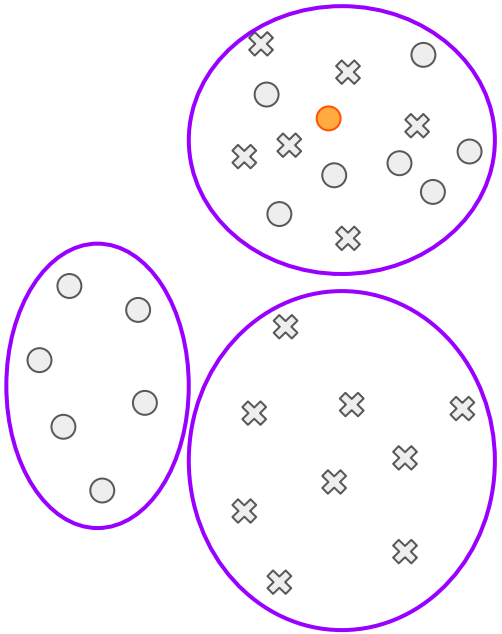
KMeans has no memory of previous selected samples.

Incremental solves this by allowing to predefine fixed clusters. Those points will stay there and act as repellers for other centroids:

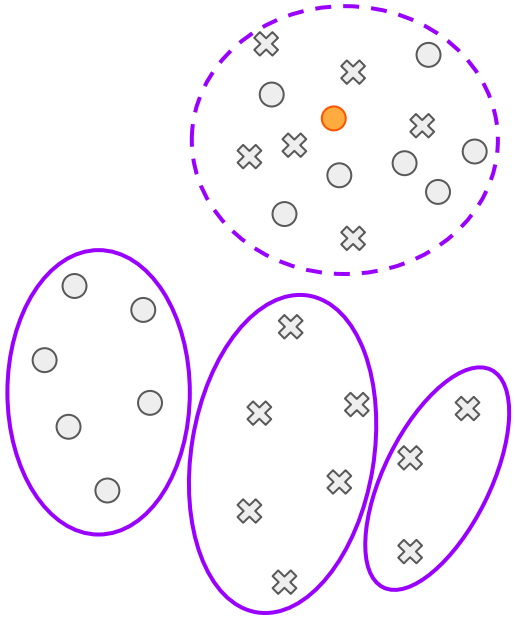
$$\text{argmax}_{\mathcal{D}_B} \text{sim}(\mathcal{D}_B, \mathcal{D}_U)$$
$$\text{subject to } \text{argmin}_{\mathcal{D}_B} \text{sim}(\mathcal{D}_B, \mathcal{D}_L)$$

Incremental KMeans does not explore the noisy area again and favor more exploration.

KMeans



Incremental KMeans



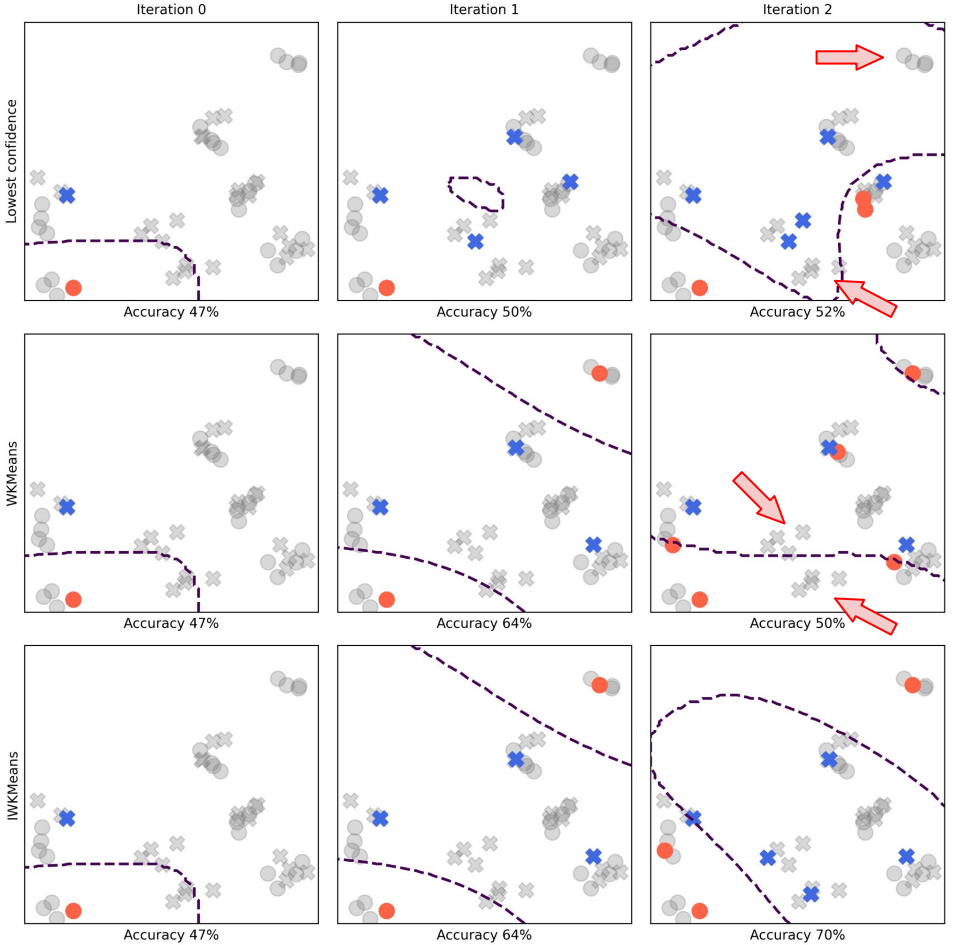
Incremental KMeans

Synthetic data

This example contains:

- 2 classes
- 6 single-class blobs
- 3 two-class blobs

We run two iterations of active learning and observe that all methods with a strong focus on uncertainty have focused on noisy blobs and missed single-class ones.



Metrics for Active Learning Monitoring

Metrics for Active Learning

How to get insight on an active learning experiment?

Classical cross-validation

Yields practical problems:

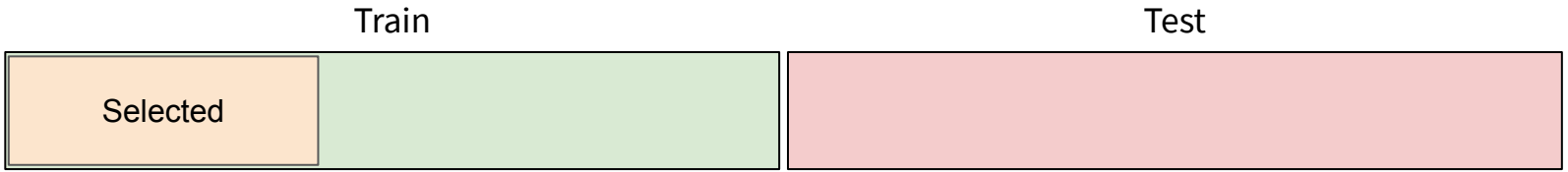
- The labeled set is usually small
- Its size grows at each iteration
- The labeled set is biased by the query strategy

Independant labeled test set

Yields business problems:

- Ideal solution from a practical point of view
- The test set is a *lost budget*
- Used in most research papers to compare methods

Can we get insights from an independant unlabeled test set?



Metrics for Active Learning

Did we explore the sample space enough?

Intuition: We measure exploration as the distance between our test set and labeled samples. We are particularly interested in its gradient. It is decreasing by definition.

$$\mathrm{EG}(t) = \sum_{\mathbf{x} \in \mathcal{D}_{Te}} d(\mathbf{x}, \mathcal{D}_{L_{t-1}}) - \sum_{x \in \mathcal{D}_{Te}} d(\mathbf{x}, \mathcal{D}_{L_t})$$

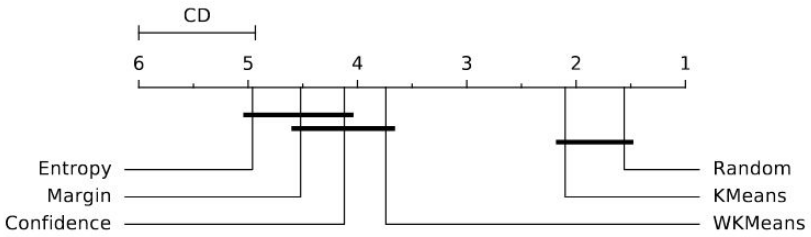


Fig. 6. Comparison of AUC of exploration scores.

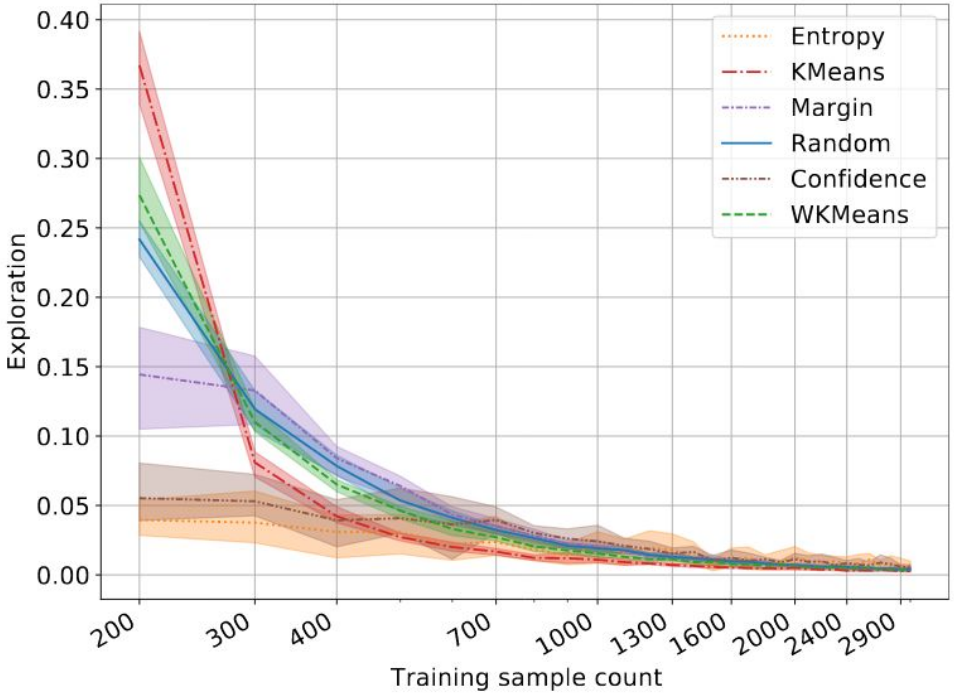
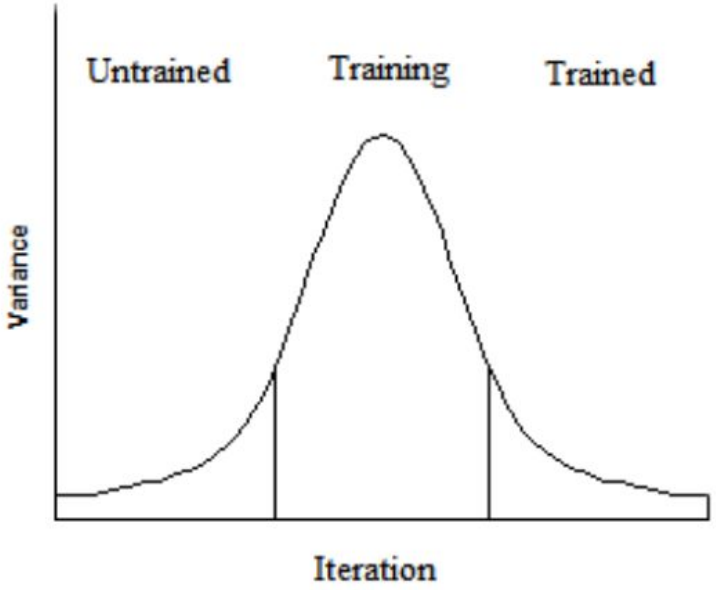


Fig. 7. Exploration metric on **LDPA**.

Metrics for Active Learning

When should we stop labeling? [Ghayoomi 2010]

Idea: Look at the variance of uncertainty score on the batch of selected samples



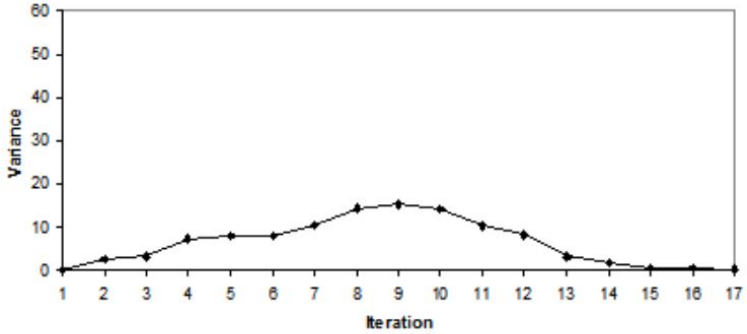


Figure 5: Variance curve of the verb *rise* for 5 folds

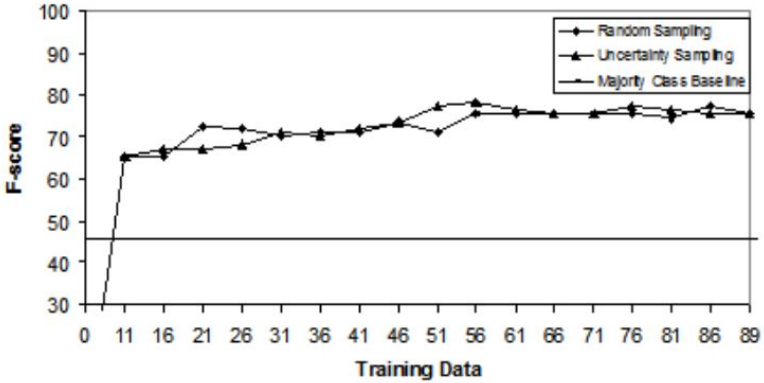
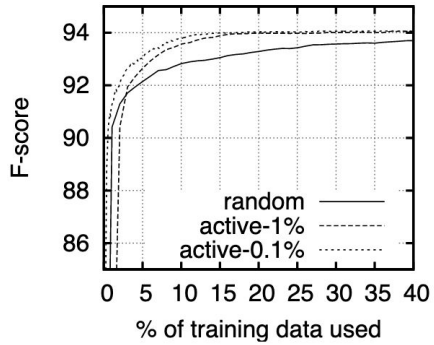
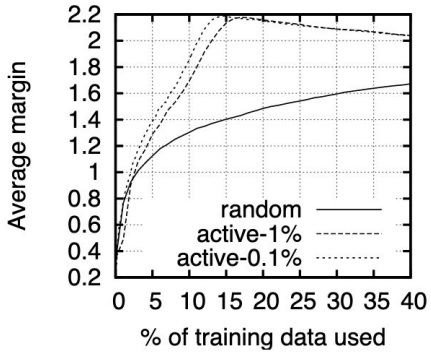


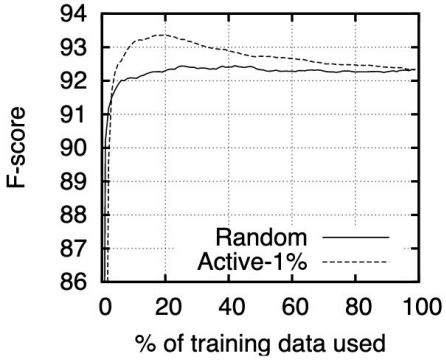
Figure 3: Learning curve of the verb *rise* for 5 folds

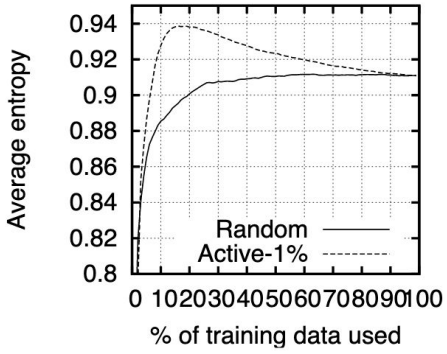
Metrics for Active Learning

When should we stop labeling? [Vlachos 2008]

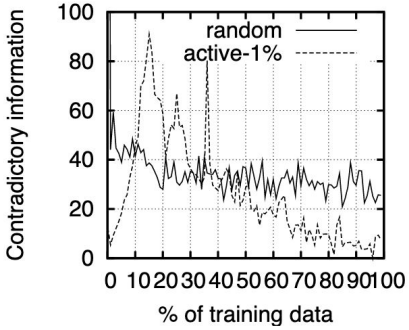


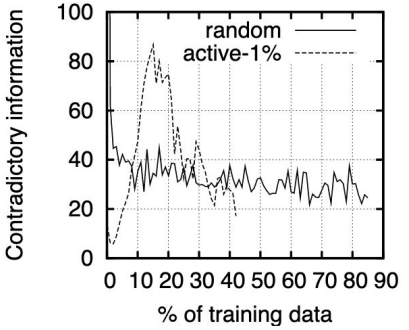






$$Contradictory\_information(t) = \sum_{i \in i^t} \frac{| f^t(x_i) |}{| f^t(x) |}$$





When should we stop labeling? [Abraham 2020]

Metrics for Active Learning

With independent labeled set: Accuracy

Accuracy is one of the most common performance metric.

$$\text{Accuracy}(t) = \frac{1}{|\mathcal{D}_{Te}|} \sum_{(\mathbf{x},y)\in\mathcal{D}_{Te}} 1_{[y = h_t(\mathbf{x})]}$$

Note that our proxy metric has not been tested against F-score and may require modifications in that case.

With independent unlabeled set: Contradiction ratio

Intuition: The increase in accuracy is bounded by the number of samples for which the label has changed. This measure is called contradiction ratio.

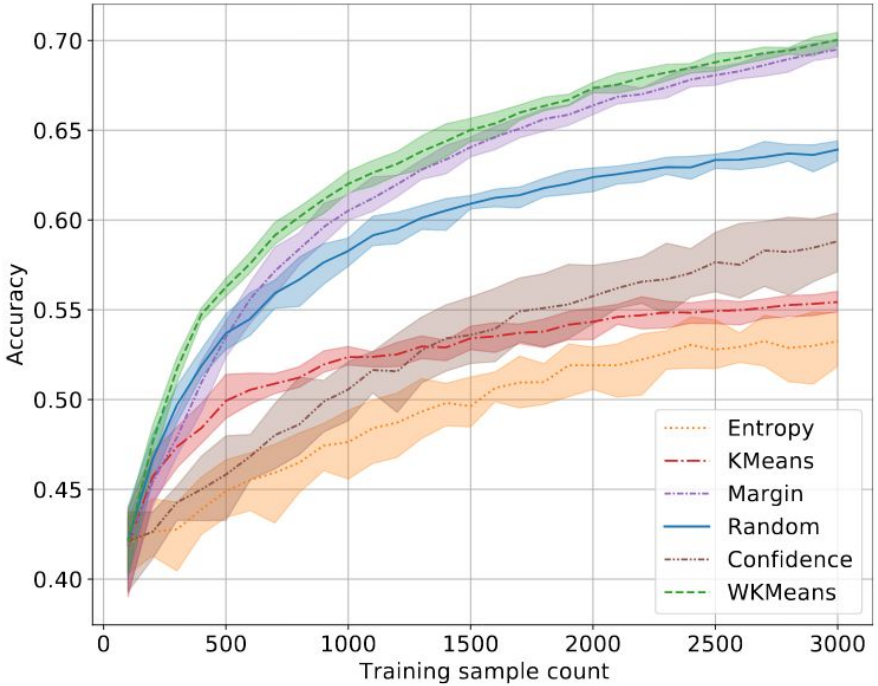
$$\text{C}(t) = \frac{1}{|\mathcal{D}_{Te}|} \sum_{(x,y)\in\mathcal{D}_{Te}} \mathbb{1}_{[h_{t-1}(x) = h_t(x)]}$$

This measures give two pieces of information:

- If it is near 0, we do not expect great improvements
- If it is flatlining, our query strategy may be stuck in a local minimum

Metrics for Active Learning

When should we stop labeling? [Abraham 2020]



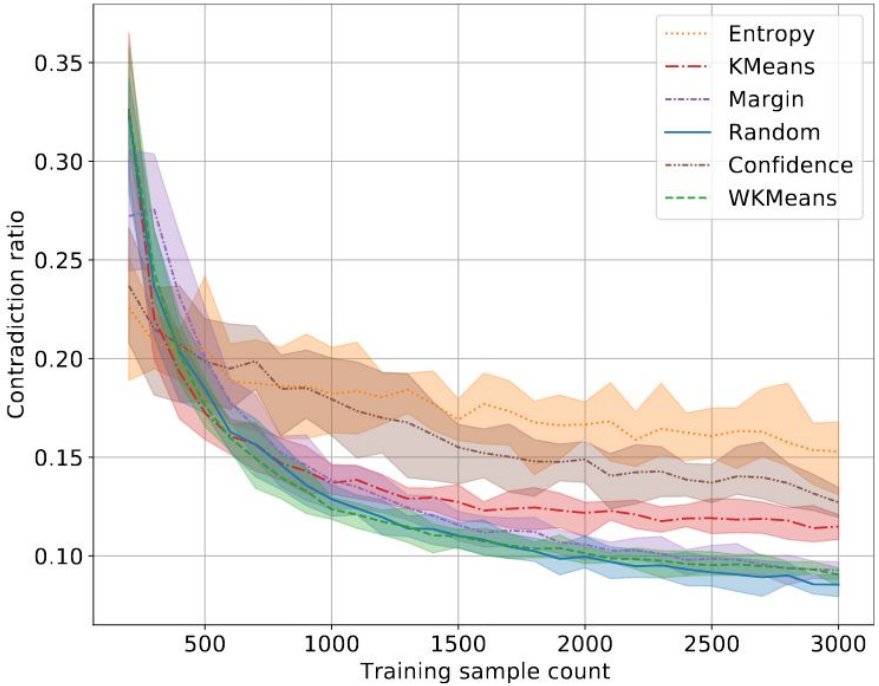


Fig. A.6. Accuracy and contradiction ratio for **LDPA**

Metrics for Active Learning

When should we stop labeling? [Abraham 2020]

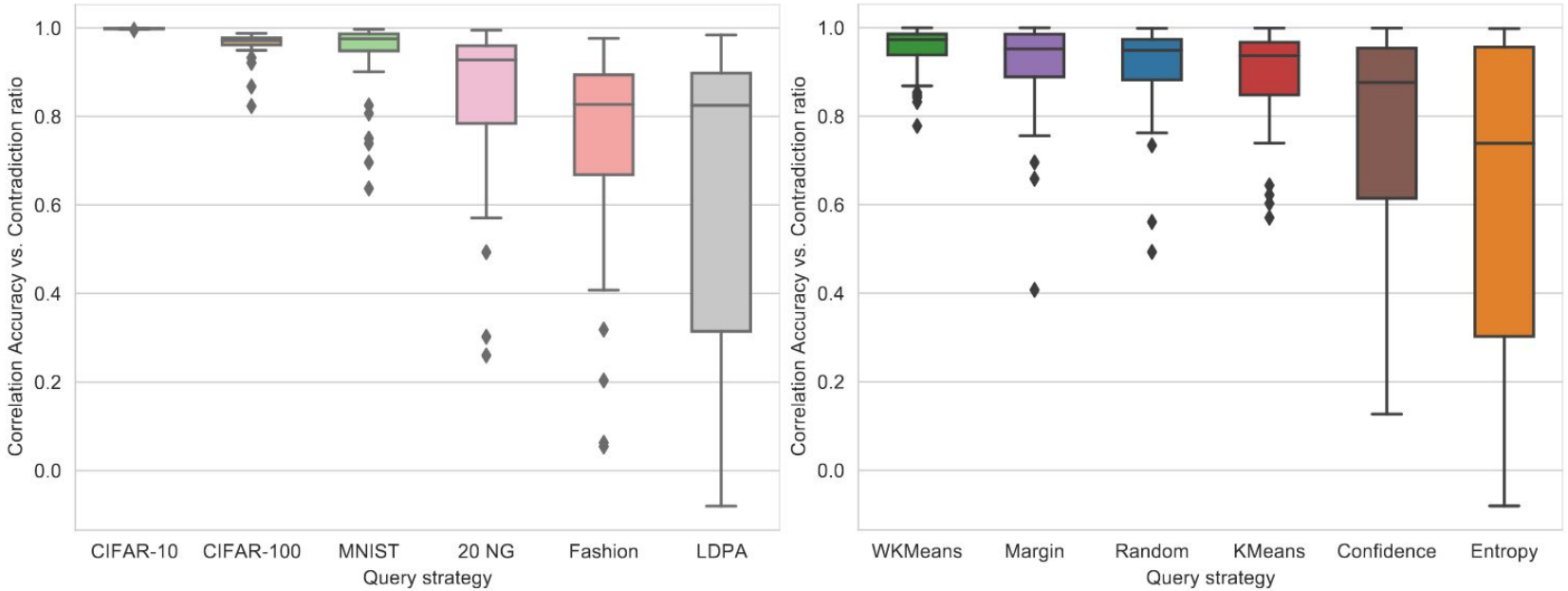


Fig. 4. Correlations between accuracy and contradictions. *Left.* By dataset. *Right.* By method.

Metrics for Active Learning

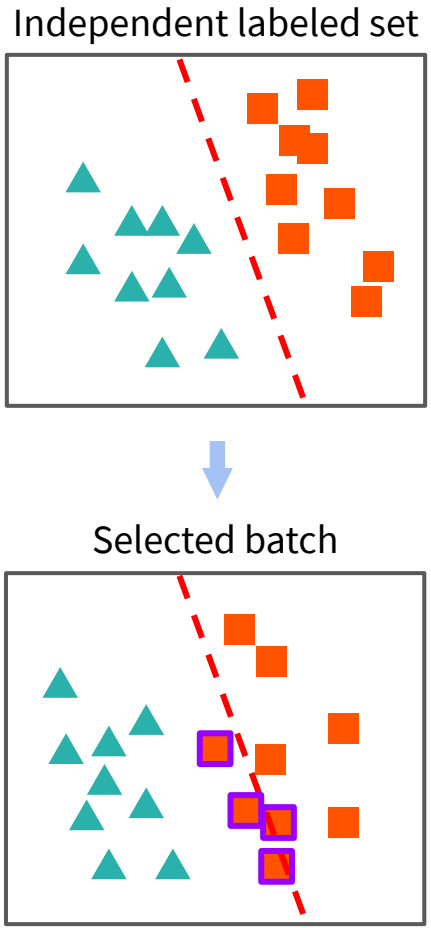
Hard to classify samples

Intuition: No classification task is perfect even with a large training set. We call samples *hard to classify* if a classifier trained on a large part of the dataset fails to classify them.

Train a reference classifier on our independent labeled set

Use it to identify hard to classify samples in a selected batch.

Hard to classify sample are likely to be selected by uncertainty methods. Do they have an effect on Active Learning experiments?



Metrics for Active Learning

Hard to classify samples [Abraham 2020]

With independent labeled set: Easiness

Easiness ratio using a classifier trained on our labeled set.

$$\text{reverse\_acc}(\mathcal{D}_B) = \frac{1}{|\mathcal{D}_B|} \sum_{(\mathbf{x},y)\in\mathcal{D}_B} \mathbb{1}_{h_{\mathcal{D}_{Te}}(\mathbf{x})=y}$$

Note that we have access to the label of samples selected in the batch after sending them to the oracle.

With independent unlabeled set: Agreement

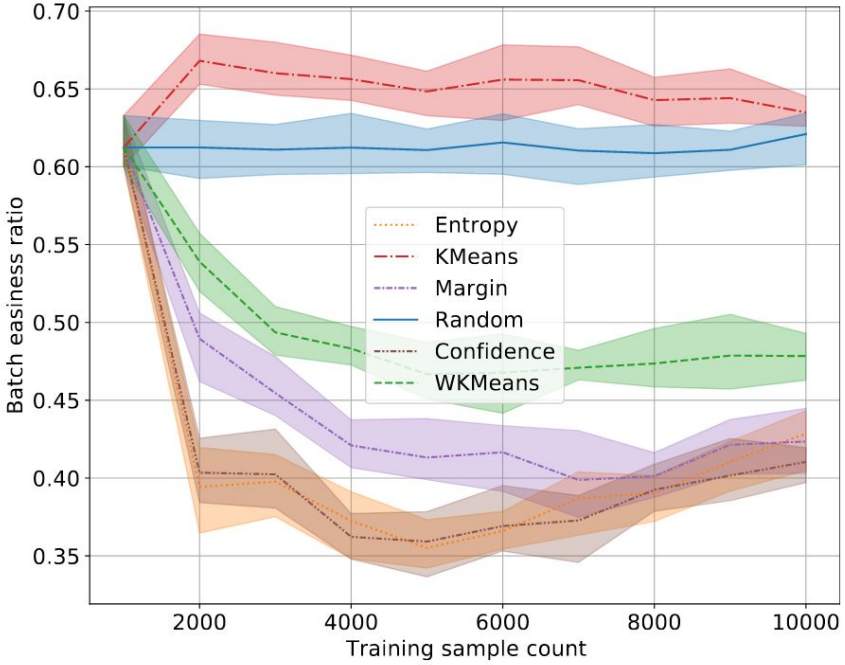
Intuition: Since we have no ground truth, we consider the agreement between our classifier and a 1-nearest-neighbor classifier train on labeled samples.

$$\kappa(\mathcal{D}_B) = \frac{1}{|\mathcal{D}_B|} \sum_{x\in\mathcal{D}_B} \mathbb{1}_{h(x)=h_{NN}(x)}$$

At the beginning, both classifiers have high uncertainty, making this score less reliable. We expect the scores to become increasingly reliable.

Metrics for Active Learning

Hard to classify samples [Abraham 2020]



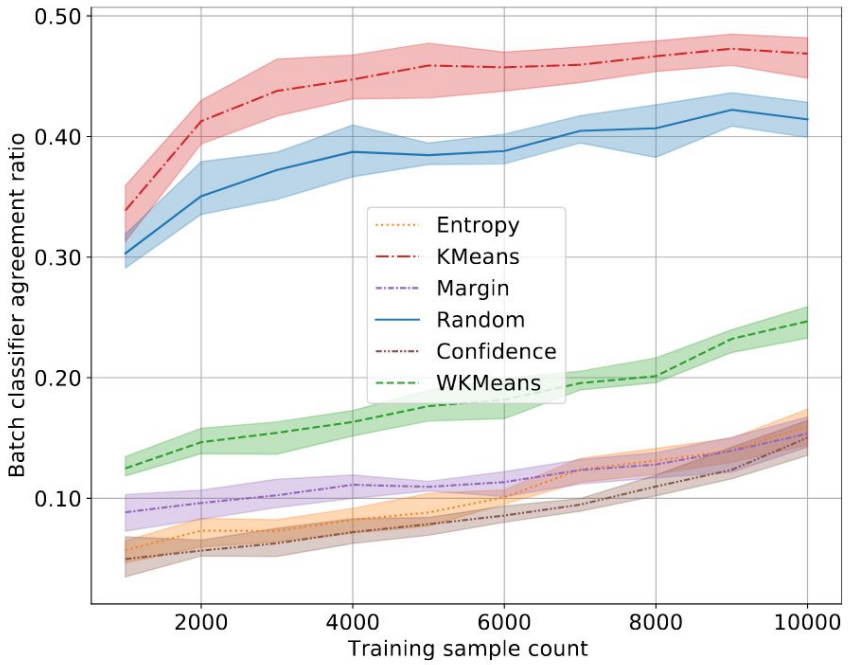


Fig. A.20. Batch easiness and batch classifier agreement for **CIFAR-100**

Metrics for Active Learning

Hard to classify samples [Abraham 2020]

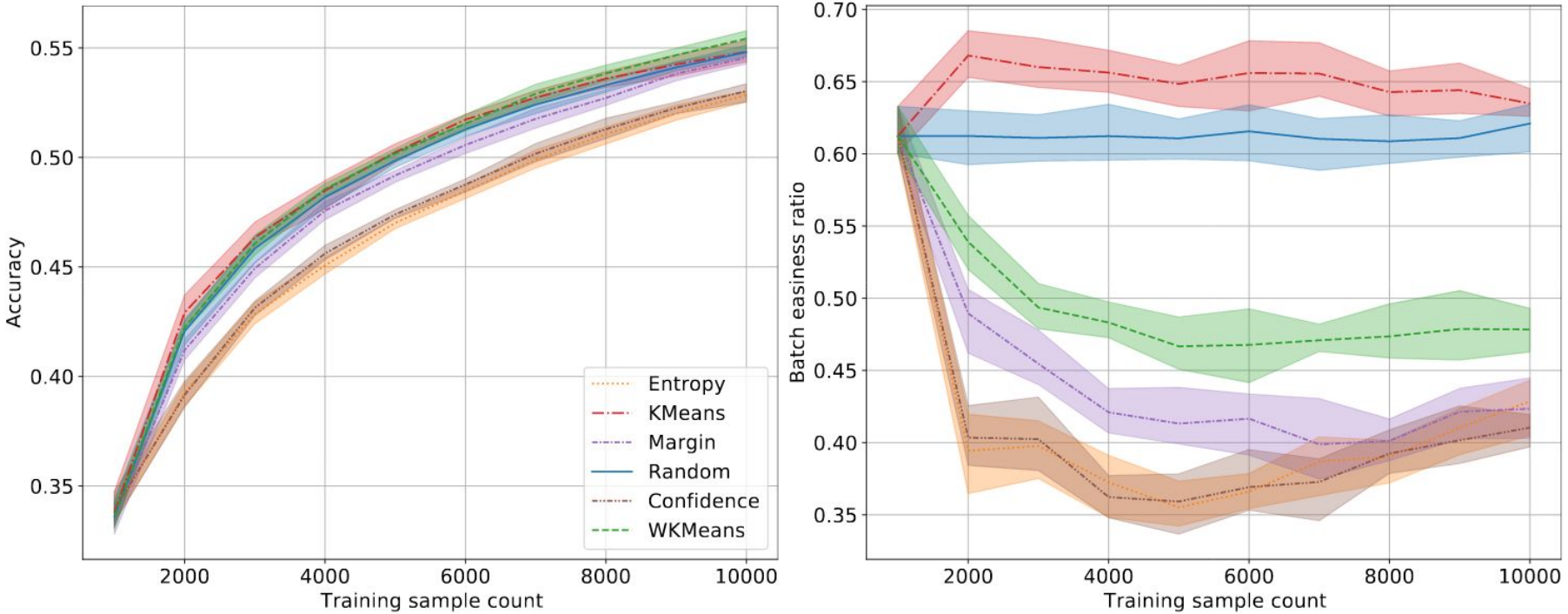


Fig. 8. Accuracy of different strategies (*left*) and easiness ratio (*right*) on **CIFAR-100**. Legends are the same for both plots.

Metrics for Active Learning

Hard to classify samples [Abraham 2020]

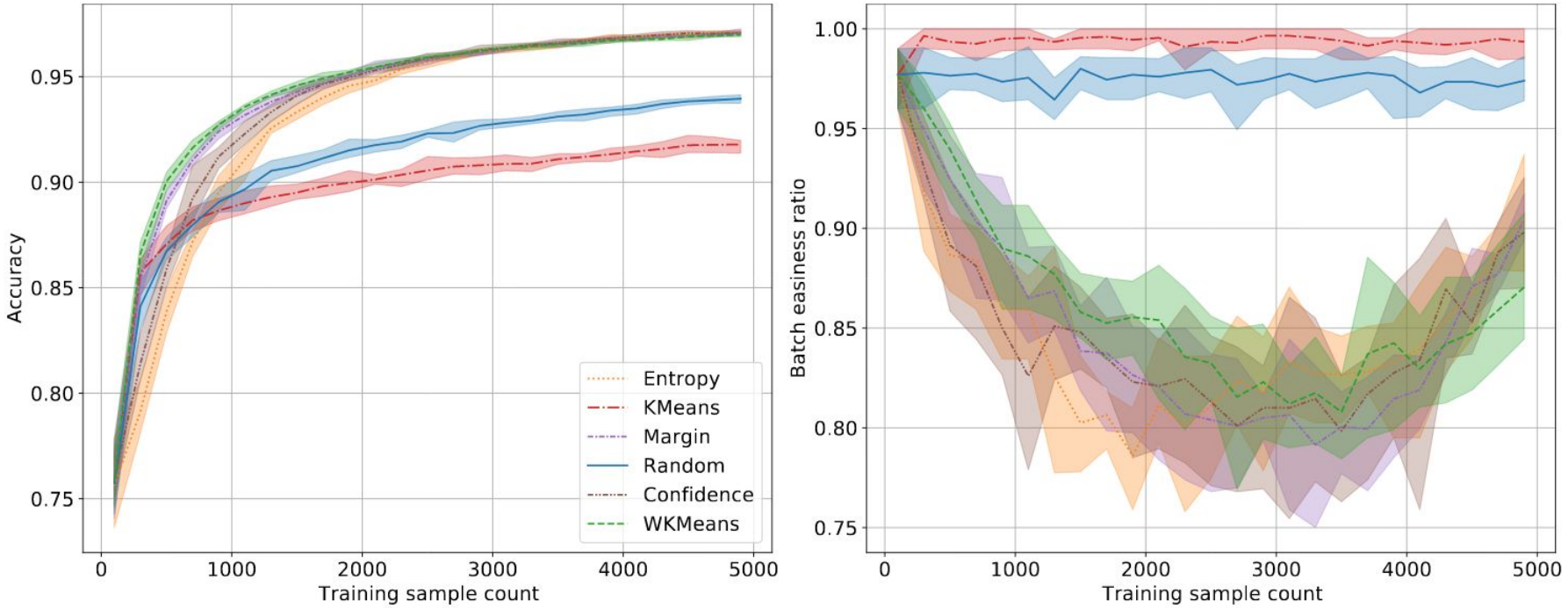


Fig. 9. Accuracy of different strategies (*left*) and easiness ratio (*right*) on **MNIST**. Legends are the same for both plots.

Metrics for Active Learning

Hard to classify samples [Abraham 2020]

Agreement seems to be a good proxy on reverse accuracy. The best performing models are in a sweet spot.

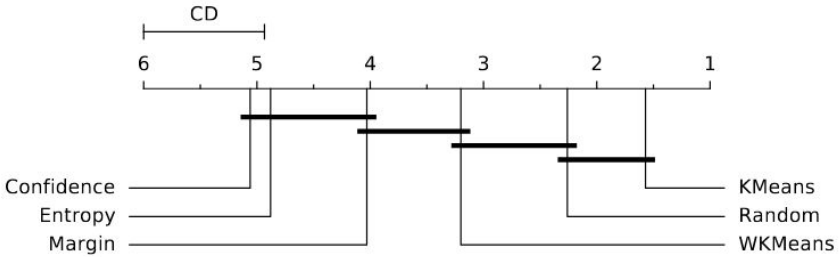


Fig. 10. Ranking of strategies selecting the easiest samples.

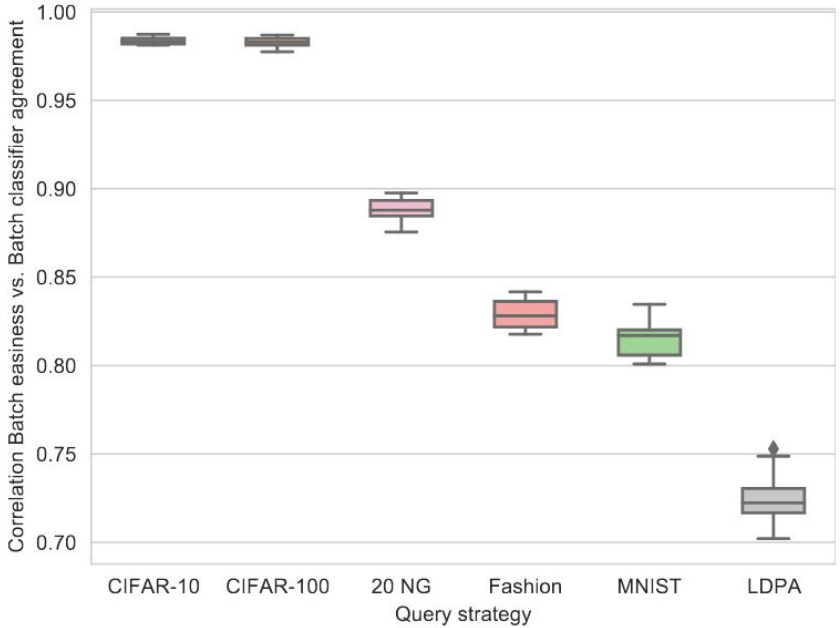


Fig. 11. Correlation between strategy ranking determined by the agreement metric and measured sample difficulty

Metrics for Active Learning

Recommended procedure

1. Set aside an independent set of labels, ideally half of the data.
2. While Random/KMeans explore more than the desired strategy, keep exploring.
3. When running the experiment, keep an eye on agreement and contradictions:
    a. If contradictions are not decreasing, change for a strategy selecting easier batches
    b. If agreement is above random, change for a strategy selecting harder batches
    c. If contradictions are low, stop labeling

But:

- We sometimes rely on comparison to other methods. Is that reliable?
- Does it generalize to other tasks such as object detection or named entity recognition?
- Can it help designing a query strategy?

Beyond Active Learning

Weak / group labeling

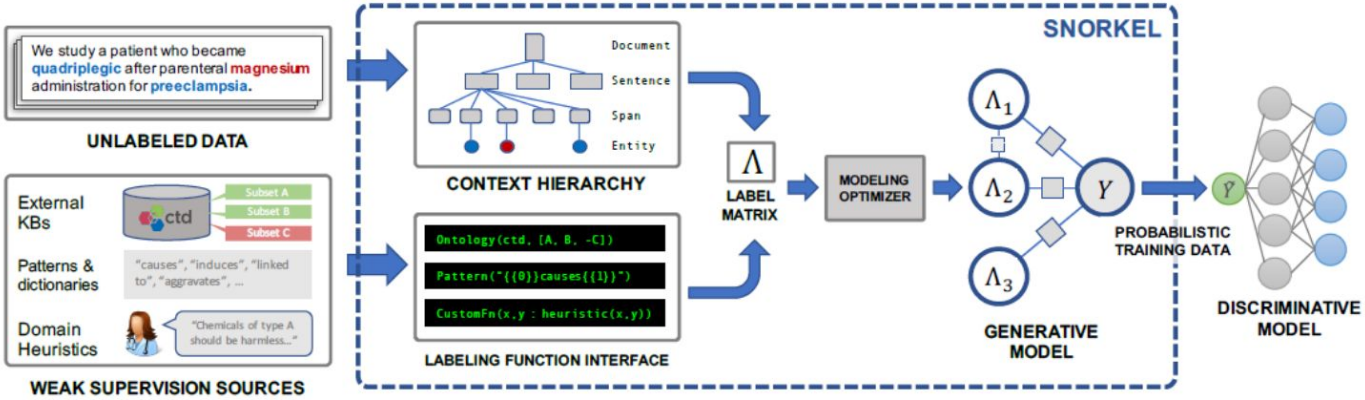


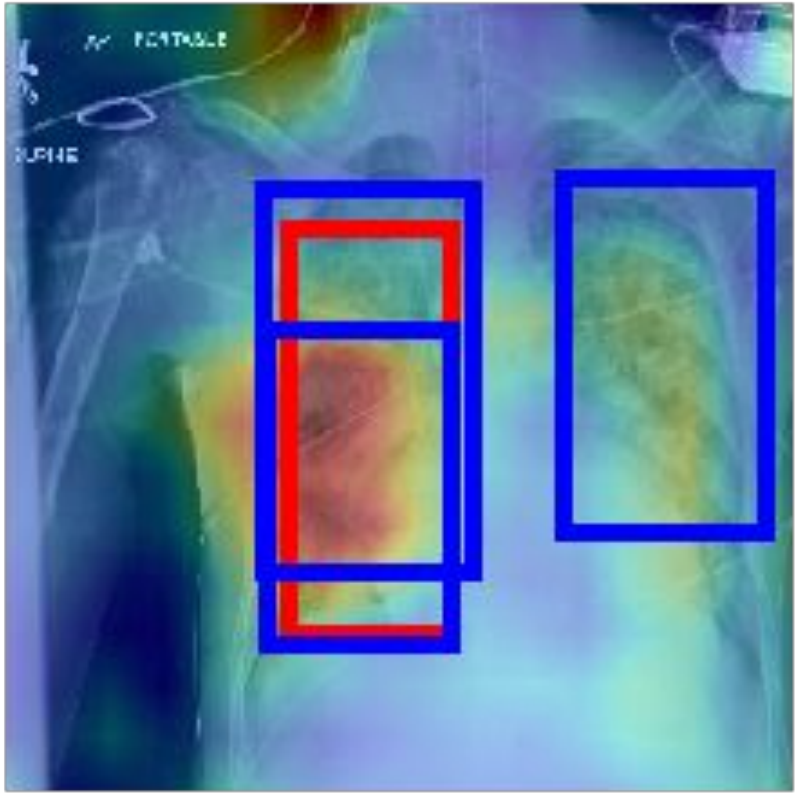
Table 3: Evaluation of Snorkel on relation extraction tasks from text. Snorkel's generative and discriminative models consistently improve over distant supervision, measured in F1, the harmonic mean of precision (P) and recall (R). We compare with hand-labeled data when available, coming within an average of 1 F1 point.

	Distant Supervision			Snorkel (Gen.)				Snorkel (Disc.)				Hand Supervision		
Task	P	R	**F1**	P	R	**F1**	**Lift**	P	R	**F1**	**Lift**	P	R	**F1**
Chem	11.2	41.2	17.6	78.6	21.6	33.8	+16.2	87.0	39.2	54.1	+36.5	-	-	-
EHR	81.4	64.8	72.2	77.1	72.9	74.9	+2.7	80.2	82.6	81.4	+9.2	-	-	-
CDR	25.5	34.8	29.4	52.3	30.4	38.5	+9.1	38.8	54.3	45.3	+15.9	39.9	58.1	47.3
Spouses	9.9	34.8	15.4	53.5	62.1	57.4	+42.0	48.4	61.6	54.2	+38.8	47.8	62.5	54.2

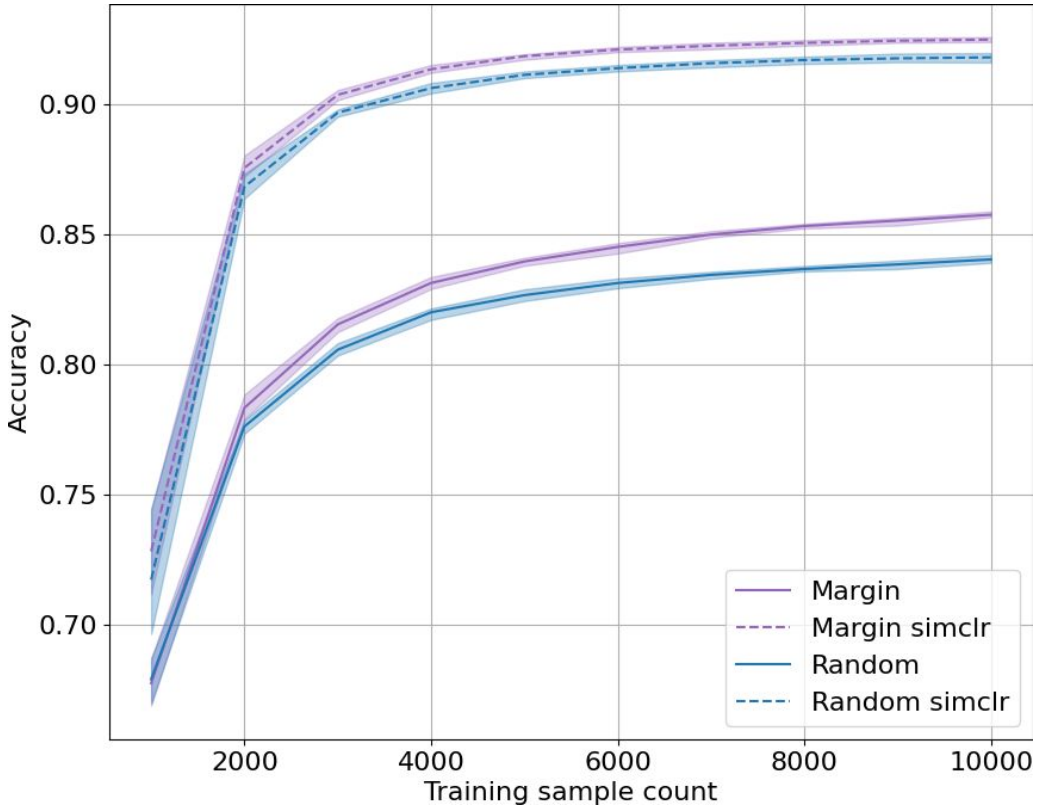
Using an auxiliary task

Goal: Can we make radiologist's job easier?

- RNSA pneumonia challenge
- Trained a pre-labeling technique:
  - using annotated data
  - using an auxiliary task
- Red: ground truth
- Blue: Predicted by Retina-Net pretrained on Coco
- Heatmap: Prediction weights of the last conv layer of a VGG16 trained on a simple classification task



Self supervision



Final exercise

Challenge

Come and show your best shot!

Active Learning in the Industry

Nader Salman, Project Manager Data Science Platform at Schlumberger

Active Learning in the Industry

Schlumberger

Defining noisy samples

Observations on Active Learning

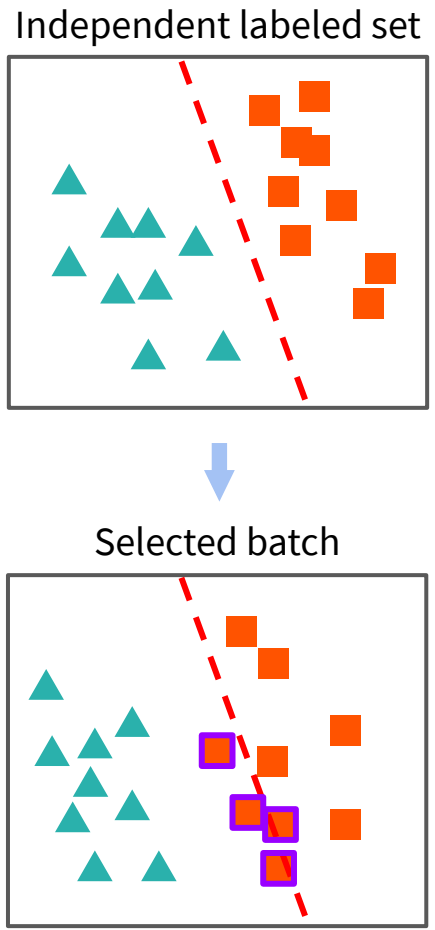
Hard to classify samples [Abraham 2020]

Intuition: No classification task is perfect even with a large training set. We call samples *hard to classify* if a classifier trained on a large part of the dataset fails to classify them.

Train a reference classifier on our independent labeled set

Use it to identify hard to classify samples in a selected batch.

$$\text{reverse\_acc}(\mathcal{D}_B) = \frac{1}{|\mathcal{D}_B|} \sum_{(\mathbf{x},y)\in\mathcal{D}_B} \mathbb{1}_{h_{\mathcal{D}_{Te}}(\mathbf{x})=y}$$



Abraham, Alexandre, and Léo Dreyfus-Schmidt. "Rebuilding Trust in Active Learning with Actionable Metrics." 2020 International Conference on Data Mining Workshops (ICDMW). IEEE, 2020.

Observations on Active Learning

Hard to classify samples [Abraham 2020]

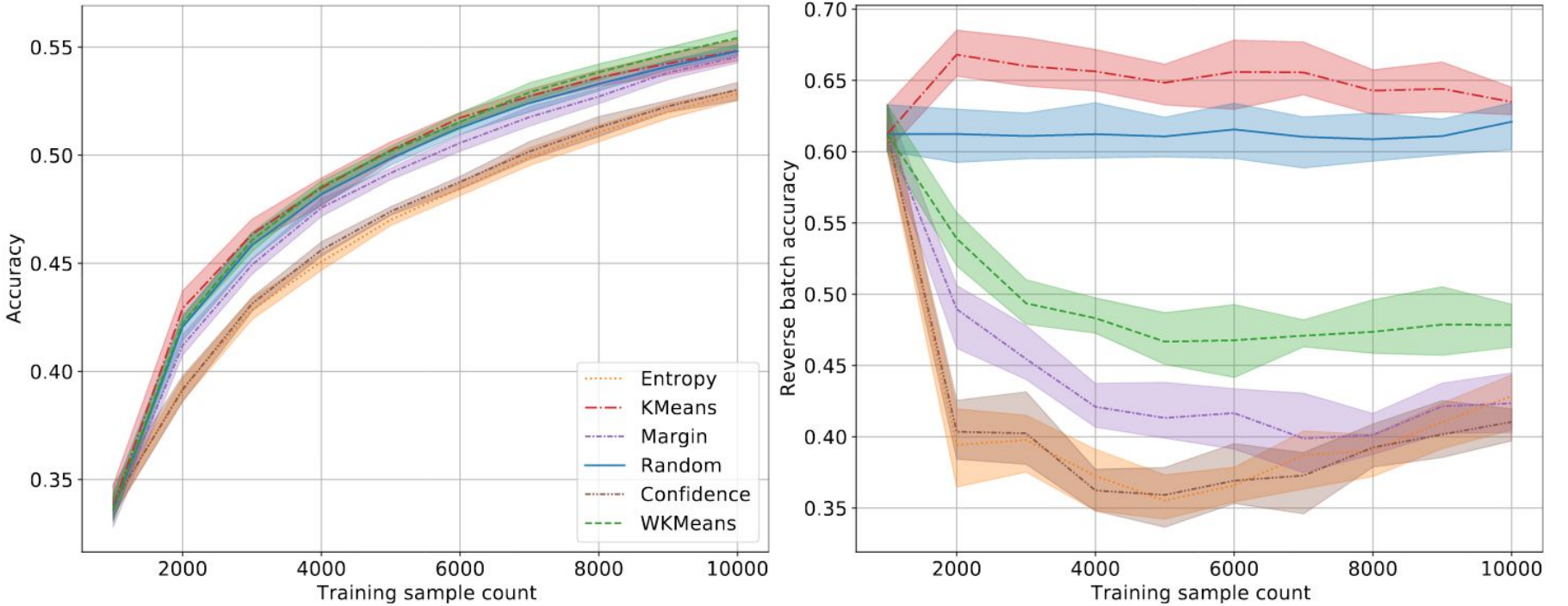
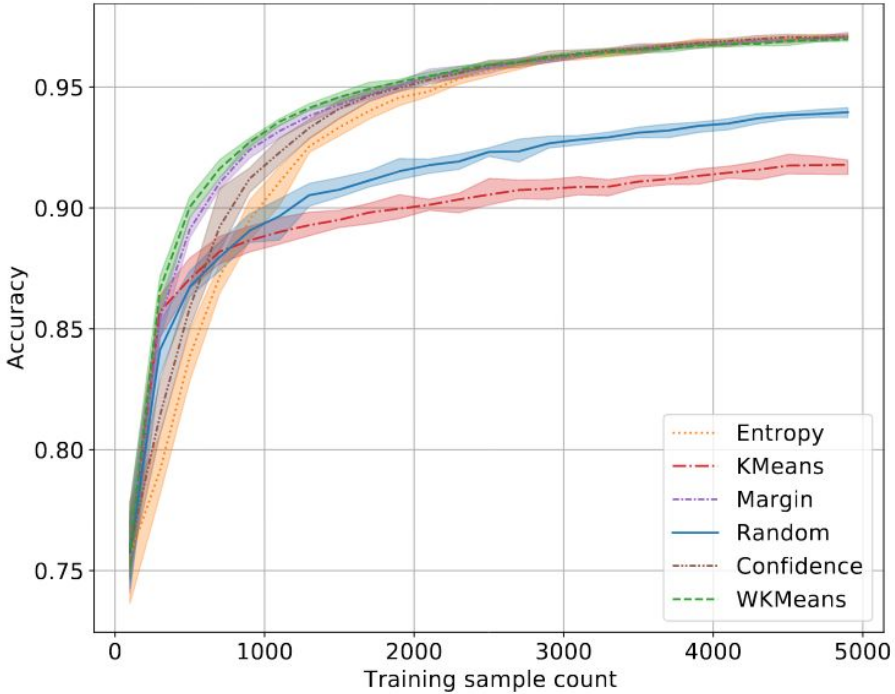


Fig. 8. Global accuracy of different strategies (*left*) and reverse batch accuracy (*right*) on **CIFAR-100**. Legends are the same for both plots.

Observations on Active Learning

Hard to classify samples [Abraham 2020]



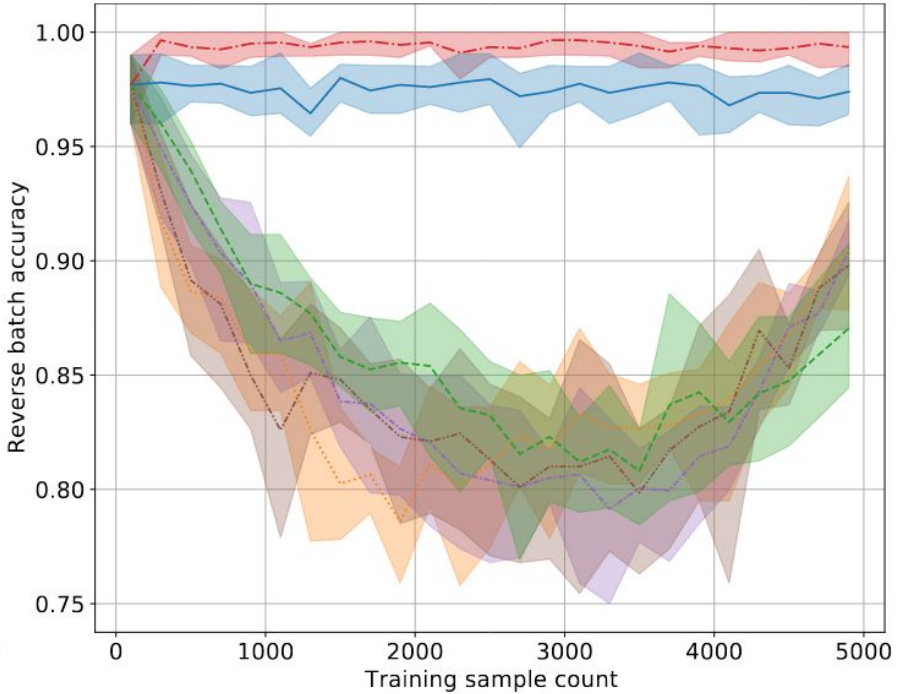


Fig. 9. Global accuracy of different strategies (*left*) and reverse batch accuracy (*right*) on **MNIST**. Legends are the same for both plots.

Experiments

Experiments

Synthetic data

Task parameters:

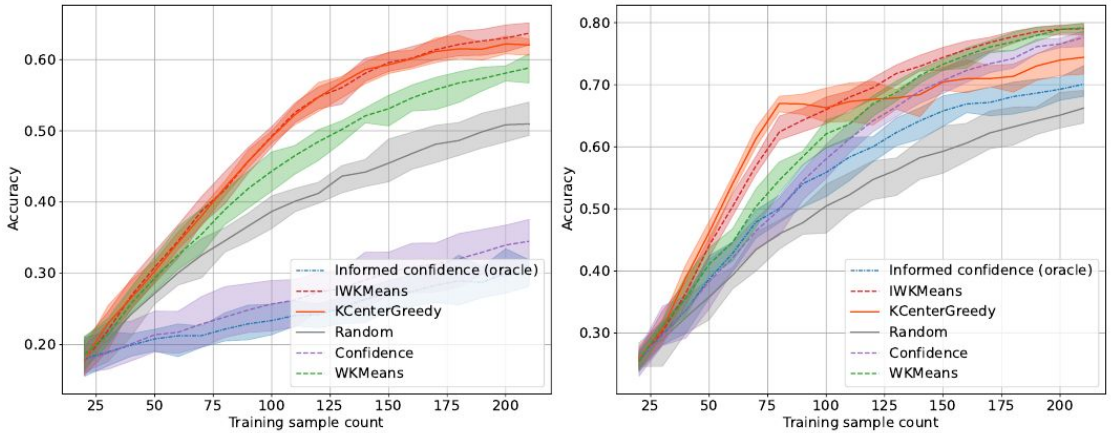
- 10k samples
- 10 classes
- batch size of 20

Low dimensional setting:

- 2 features
- 200 blobs including 100 noisy

High dimensional setting:

- 40 features
- 90 blobs including 30 noisy



(a) 2 features (b) 40 features

Fig. 2. Test accuracy on synthetic problems.

Table 1. AUC and ratio of noisy samples per method. Standard deviation is in parenthesis. Best answers in terms of accuracy (higher) and Noisy Sample Ratio (lower) are in bold.

Dataset	Metric	Random	KCenter	Confidence	IConfidence	WKMeans	IWKMeans
Noisy LD	AUC	38.6 (1.5)	47.9 (0.5)	26.7 (2.5)	24.5 (1.4)	44.0 (1.2)	**48.1** (1.0)
Noisy LD	NSR	50.3 (4.1)	42.4 (2.0)	38.9 (6.5)	**10.1** (4.6)	43.5 (2.6)	39.3 (1.8)
Noisy HD	AUC	50.7 (2.1)	61.7 (1.1)	58.0 (1.2)	55.0 (1.5)	60.6 (0.9)	**63.2** (0.6)
Noisy HD	NSR	35.0 (3.0)	24.5 (1.5)	25.6 (1.5)	**3.2** (1.1)	33.4 (1.5)	26.9 (1.8)

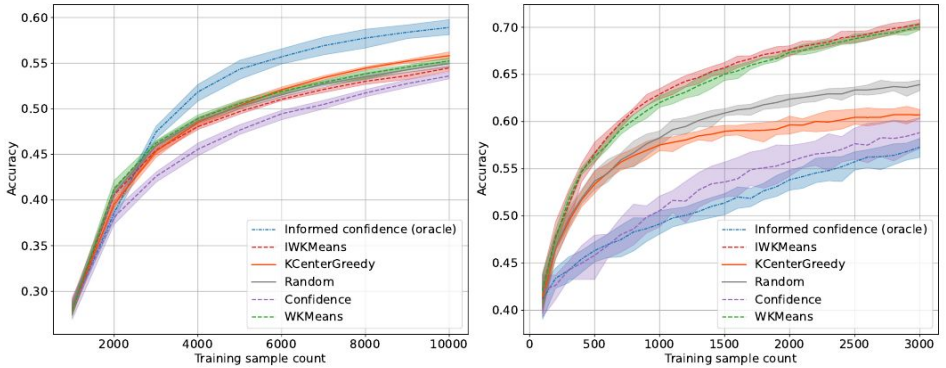
Experiments

Real tasks

IWKMeans is not significantly different than WKMeans

IConfidence has highest RBA and leads the way except on LDPA

Second best are WKMeans and IWKMeans with lower RBA



(c) CIFAR 100 (d) LDPA

Table 2. Area under the curve for accuracy (AUC) and reverse batch accuracy (RBA) per method averaged over all repetitions. Standard deviation is in parenthesis. Bold values are statistically significantly higher than the others based on a Friedman test with Nemenyi post-hoc test which details are available in Fig. A5 in appendix.

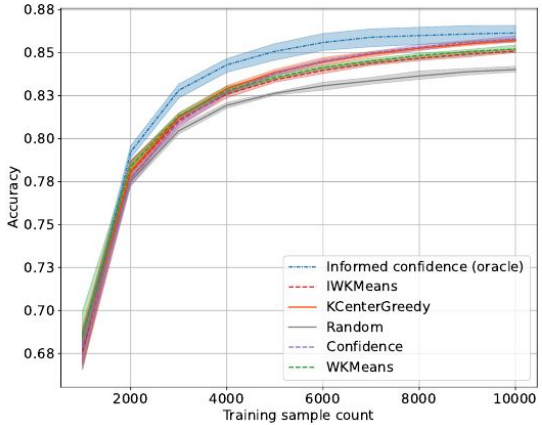
Dataset	Metric	Random	KCenter	Confidence	IConfidence	WKMeans	IWKMeans
LDPA	AUC	**59.0** (0.5)	57.2 (0.5)	51.9 (1.1)	51.2 (0.8)	**63.1** (0.3)	**63.6** (0.3)
LDPA	RBA	67.1 (0.7)	49.3 (2.3)	51.6 (2.0)	**98.9** (0.1)	67.8 (1.1)	67.6 (1.1)
Cifar10	AUC	80.9 (0.2)	**82.0** (0.2)	**81.9** (0.2)	**82.9** (0.4)	81.8 (0.2)	81.6 (0.2)
Cifar10	RBA	91.5 (4.8)	81.5 (10.7)	80.5 (12.6)	**94.9** (3.5)	85.2 (9.0)	85.3 (9.1)
Cifar10S	AUC	88.8 (0.2)	89.2 (0.2)	**89.5** (0.2)	**89.6** (0.3)	**89.4** (0.2)	**89.5** (0.3)
Cifar10S	RBA	93.5 (1.3)	87.5 (1.8)	80.0 (3.6)	**96.5** (0.8)	86.2 (2.8)	87.9 (2.3)
MNIST	AUC	90.9 (0.2)	91.2 (0.3)	**93.5** (0.2)	**93.8** (0.3)	**94.2** (0.1)	**94.2** (0.1)
MNIST	RBA	**97.6** (0.2)	96.6 (0.4)	92.3 (8.1)	**97.7** (2.5)	88.1 (0.4)	86.9 (0.6)
Fashion	AUC	82.4 (0.2)	79.3 (0.3)	**83.5** (0.3)	**85.0** (1.0)	**84.3** (0.1)	**84.3** (0.1)
Fashion	RBA	88.1 (0.4)	**90.8** (9.7)	82.3 (15.9)	**91.3** (7.3)	70.6 (0.7)	69.2 (0.7)
Cifar100	AUC	48.5 (0.3)	**48.3** (0.2)	46.2 (0.2)	**50.8** (0.6)	**48.9** (0.2)	49.0 (0.3)
Cifar100	RBA	69.4 (9.2)	71.2 (14.1)	55.6 (15.6)	**88.8** (5.8)	70.7 (9.2)	70.0 (9.9)

Experiments

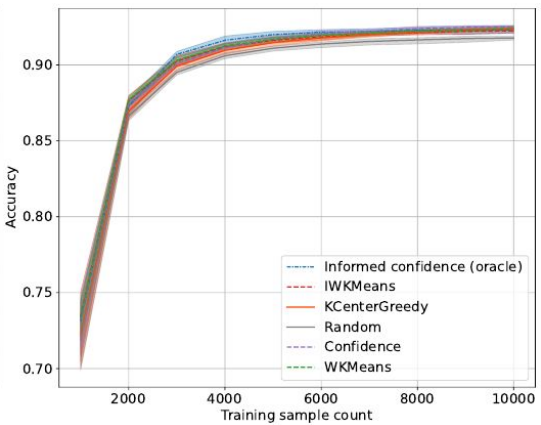
Real tasks

On MNIST and Fashion MNIST, IWKmeans and WKMeans start above IConfidence but are then outperformed.

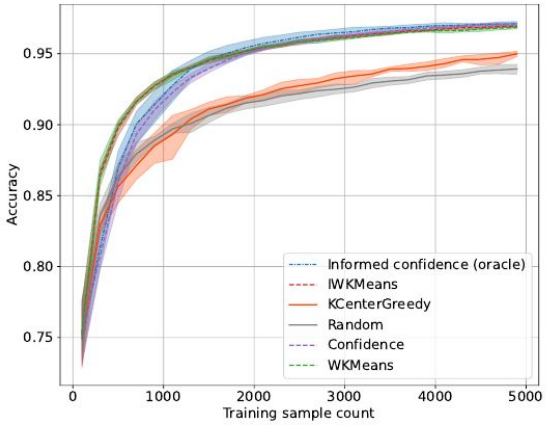
KCenterGreedy is not good on image tasks without a CNN.



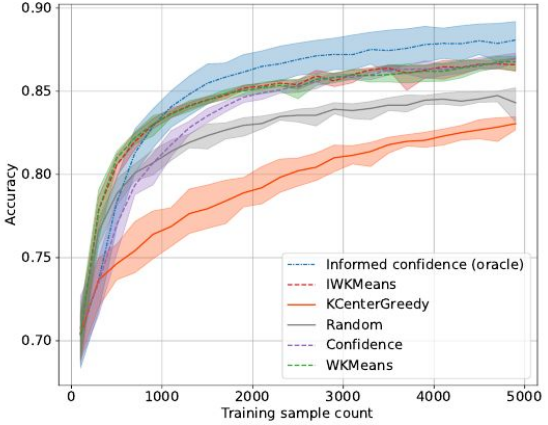
(a) CIFAR 10, ImageNet embedding



(b) CIFAR 10, SimCLR embedding



(e) MNIST



(f) Fashion MNIST

Future work

Informed Confidence

Informed confidence provides a ground truth for AL metrics. More samplers can be turned into their informed counterpart: Margin, Entropy, WKMeans...

The Kappa metric proposed in [Abraham 2020] may help creating a sampler based on the same intuition but applicable on live experiments.

The above has strong connexion with *trust scores* [Jiang 2018] that we will make explicit by designing a new sampler.

IWKMeans

From our observations, IWKMeans seems a bit more stable than WKMeans, we do not drop the method.

Despite testing several variants, we were not able to find a better formula but it could benefit of a better identification of noisy samples.

THANKS FOR YOUR ATTENTION, DO YOU HAVE ANY QUESTIONS ?

Jiang, Heinrich, et al. "To trust or not to trust a classifier." NeurIPS (2018).

Metrics for Active Learning

How to get insight on an active learning experiment?

Classical cross-validation

Yields practical problems:

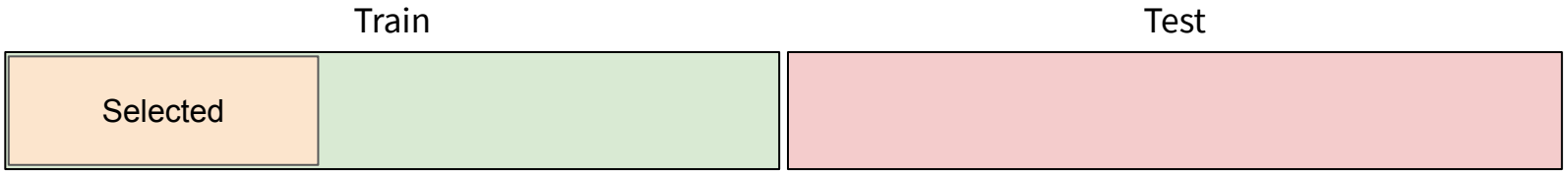
- The labeled set is usually small
- Its size grows at each iteration
- The labeled set is biased by the query strategy

Independant labeled test set

Yields business problems:

- Ideal solution from a practical point of view
- The test set is a *lost budget*
- Used in most research papers to compare methods

Can we get insights from an independant unlabeled test set?



Metrics for Active Learning

Did we explore the sample space enough?

Intuition: We measure exploration as the distance between our test set and labeled samples. We are particularly interested in its gradient. It is decreasing by definition.

$$\mathrm{EG}(t) = \sum_{\mathbf{x} \in \mathcal{D}_{Te}} d(\mathbf{x}, \mathcal{D}_{L_{t-1}}) - \sum_{x \in \mathcal{D}_{Te}} d(\mathbf{x}, \mathcal{D}_{L_t})$$

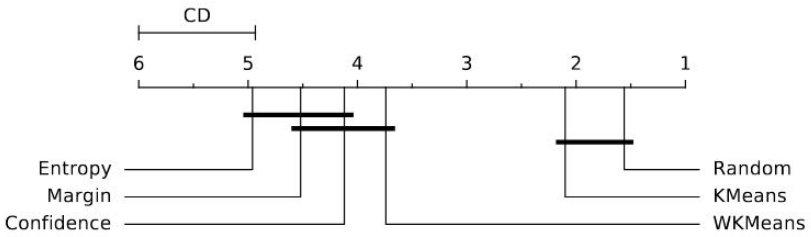


Fig. 6. Comparison of AUC of exploration scores.

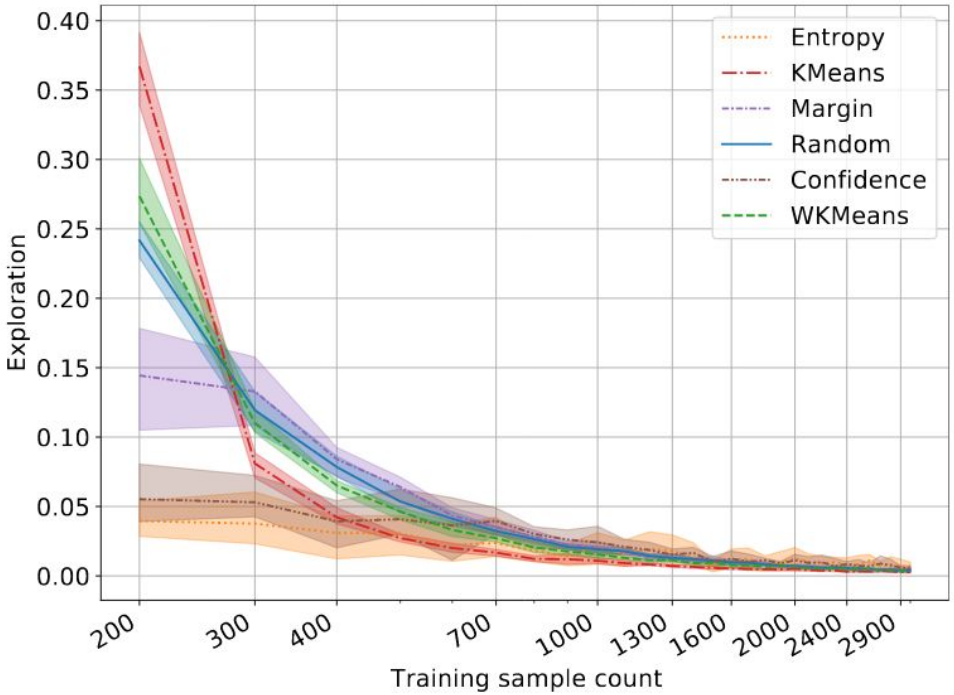


Fig. 7. Exploration metric on **LDPA**.

When should we stop labeling? [Abraham 2020]

Metrics for Active Learning

With independent labeled set: Accuracy

Accuracy is one of the most common performance metric.

$$\text{Accuracy}(t) = \frac{1}{|\mathcal{D}_{Te}|} \sum_{(\mathbf{x},y)\in\mathcal{D}_{Te}} 1_{[y = h_t(\mathbf{x})]}$$

Note that our proxy metric has not been tested against F-score and may require modifications in that case.

With independent unlabeled set: Contradiction ratio

Intuition: The increase in accuracy is bounded by the number of samples for which the label has changed. This measure is called contradiction ratio.

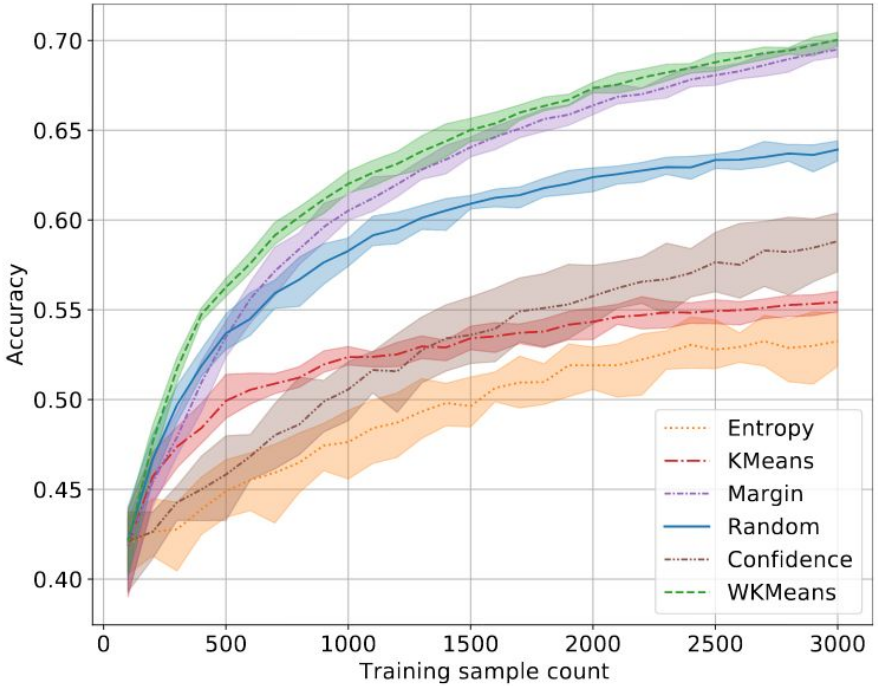
$$\text{C}(t) = \frac{1}{|\mathcal{D}_{Te}|} \sum_{(x,y)\in\mathcal{D}_{Te}} \mathbb{1}_{[h_{t-1}(x) = h_t(x)]}$$

This measures give two pieces of information:

- If it is near 0, we do not expect great improvements
- If it is flatlining, our query strategy may be stuck in a local minimum

Metrics for Active Learning

When should we stop labeling? [Abraham 2020]



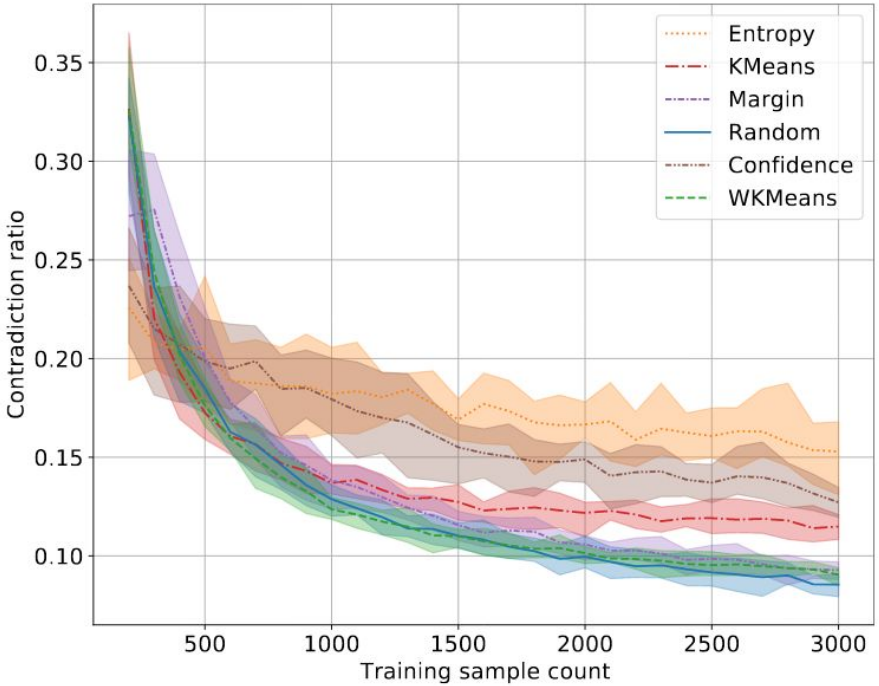


Fig. A.6. Accuracy and contradiction ratio for **LDPA**

Metrics for Active Learning

When should we stop labeling? [Abraham 2020]

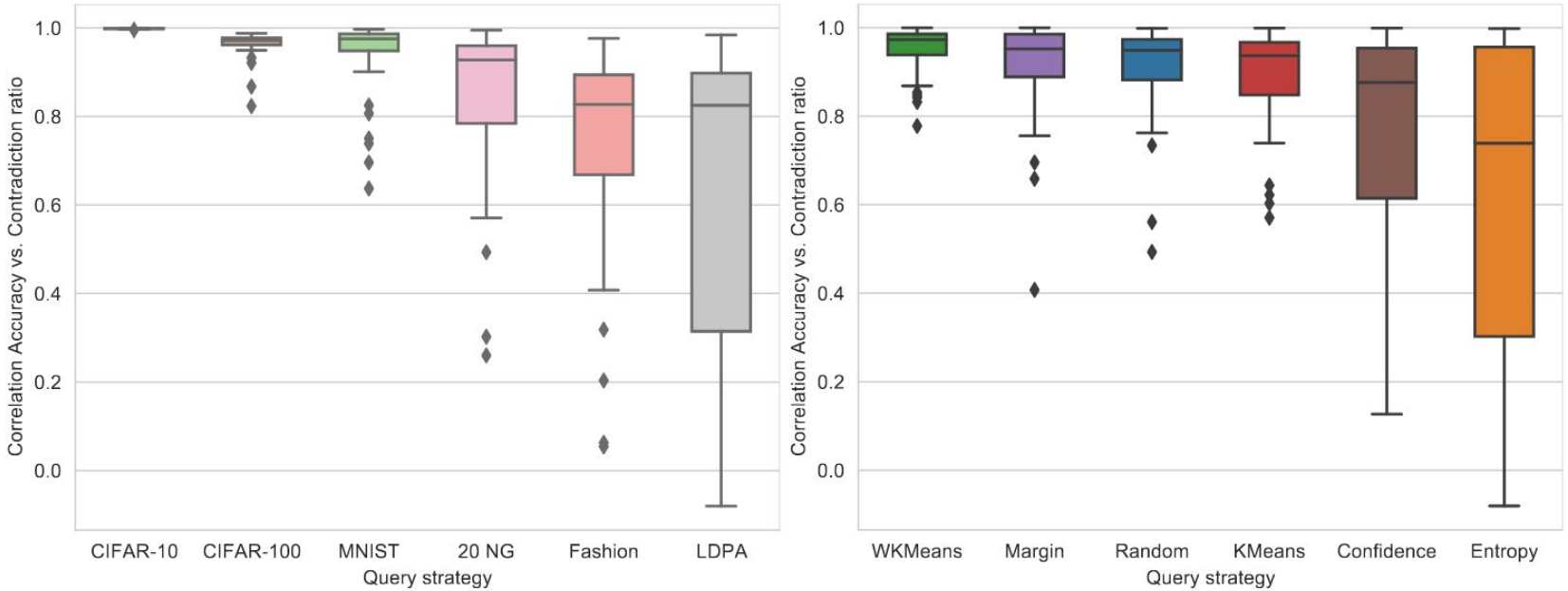


Fig. 4. Correlations between accuracy and contradictions. *Left.* By dataset. *Right.* By method.

Metrics for Active Learning

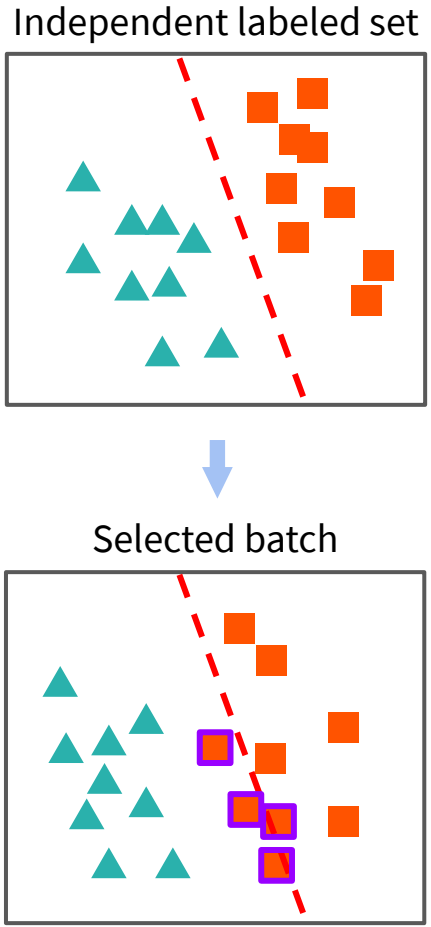
Hard to classify samples

Intuition: No classification task is perfect even with a large training set. We call samples *hard to classify* if a classifier trained on a large part of the dataset fails to classify them.

Train a reference classifier on our independent labeled set

Use it to identify hard to classify samples in a selected batch.

Hard to classify sample are likely to be selected by uncertainty methods. Do they have an effect on Active Learning experiments?



Metrics for Active Learning

Hard to classify samples [Abraham 2020]

With independent labeled set: Easiness

Easiness ratio using a classifier trained on our labeled set.

$$\text{reverse\_acc}(\mathcal{D}_B) = \frac{1}{|\mathcal{D}_B|} \sum_{(\mathbf{x},y)\in\mathcal{D}_B} \mathbb{1}_{h_{\mathcal{D}_{Te}}(\mathbf{x})=y}$$

Note that we have access to the label of samples selected in the batch after sending them to the oracle.

With independent unlabeled set: Agreement

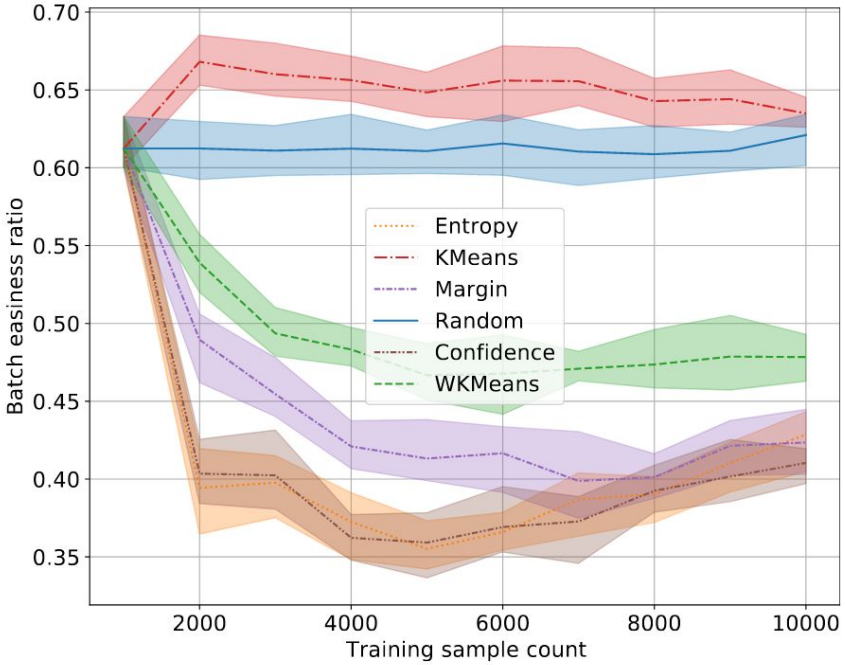
Intuition: Since we have no ground truth, we consider the agreement between our classifier and a 1-nearest-neighbor classifier train on labeled samples.

$$\kappa(\mathcal{D}_B) = \frac{1}{|\mathcal{D}_B|} \sum_{x\in\mathcal{D}_B} \mathbb{1}_{h(x)=h_{NN}(x)}$$

At the beginning, both classifiers have high uncertainty, making this score less reliable. We expect the scores to become increasingly reliable.

Metrics for Active Learning

Hard to classify samples [Abraham 2020]



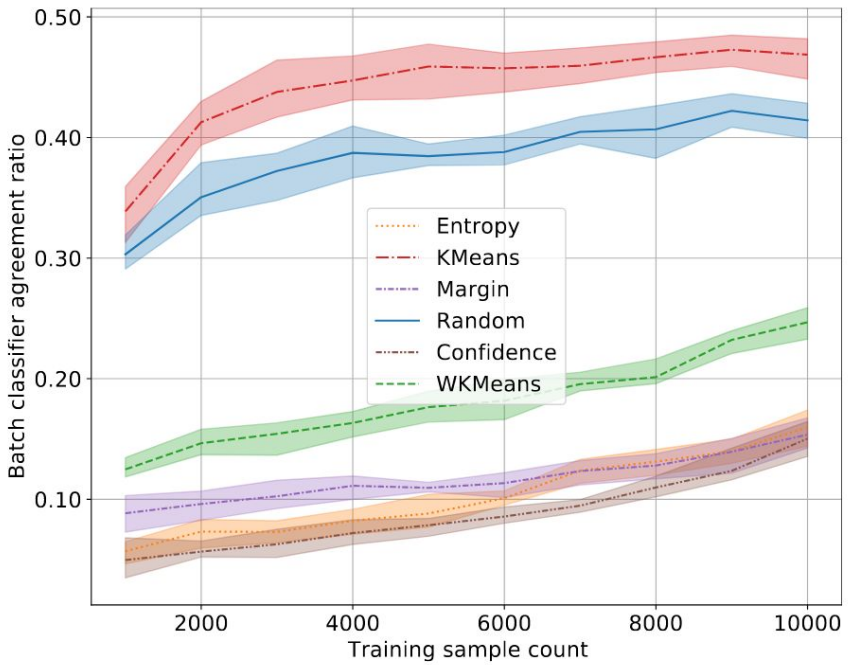


Fig. A.20. Batch easiness and batch classifier agreement for **CIFAR-100**

Metrics for Active Learning

Hard to classify samples [Abraham 2020]

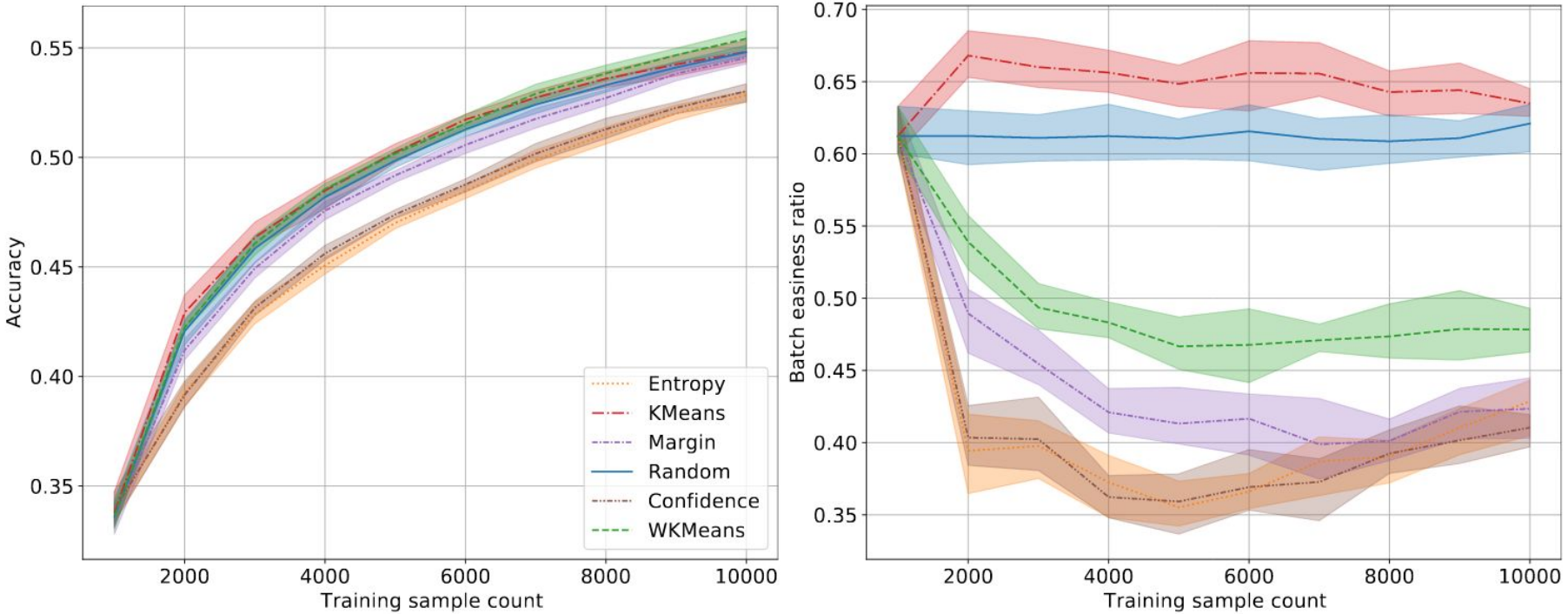


Fig. 8. Accuracy of different strategies (*left*) and easiness ratio (*right*) on **CIFAR-100**. Legends are the same for both plots.

Metrics for Active Learning

Hard to classify samples [Abraham 2020]

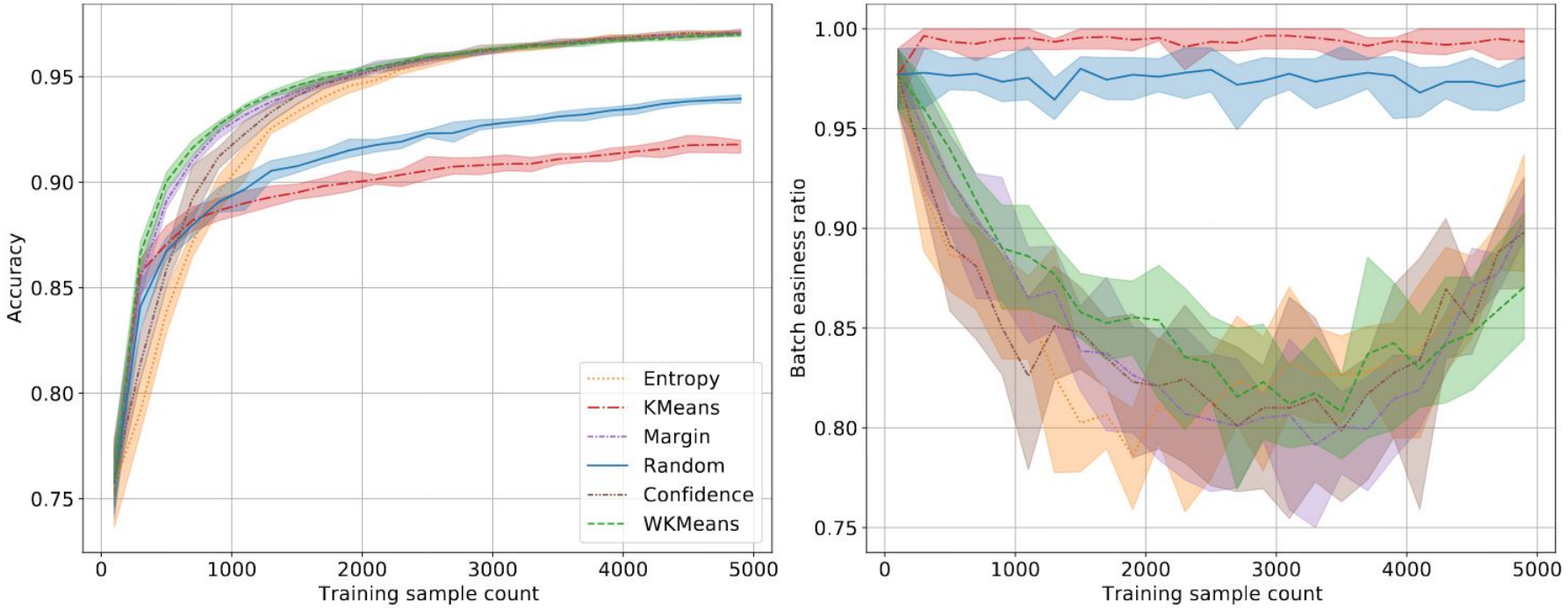


Fig. 9. Accuracy of different strategies (*left*) and easiness ratio (*right*) on **MNIST**. Legends are the same for both plots.

Metrics for Active Learning

Hard to classify samples [Abraham 2020]

Agreement seems to be a good proxy on reverse accuracy. The best performing models are in a sweet spot.

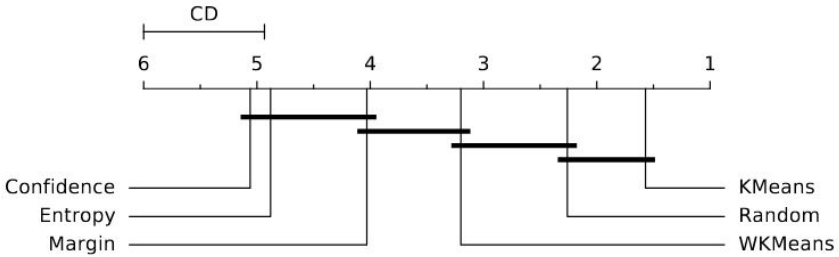


Fig. 10. Ranking of strategies selecting the easiest samples.

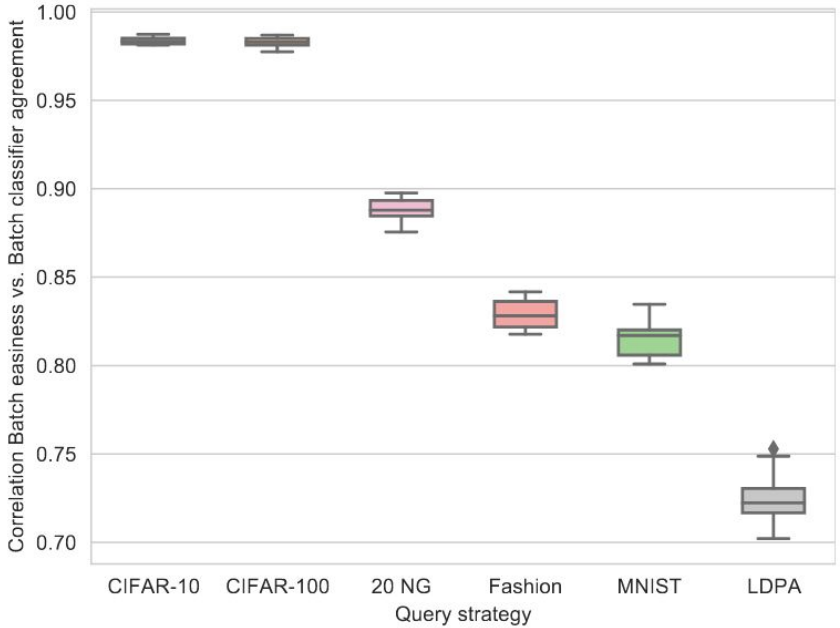


Fig. 11. Correlation between strategy ranking determined by the agreement metric and measured sample difficulty

Metrics for Active Learning

Recommended procedure

1. Set aside an independent set of labels, ideally half of the data.
2. While Random/KMeans explore more than the desired strategy, keep exploring.
3. When running the experiment, keep an eye on agreement and contradictions:
    a. If contradictions are not decreasing, change for a strategy selecting easier batches
    b. If agreement is above random, change for a strategy selecting harder batches
    c. If contradictions are low, stop labeling

But:

- We sometimes rely on comparison to other methods. Is that reliable?
- Does it generalize to other tasks such as object detection or named entity recognition?
- Can it help designing a query strategy?

Conclusion

Active Learning complex setup induces variability in results.

Experiments are hard to interpret and reproduce.

Choosing a method when starting a new task is difficult.

Metrics and insights can help understanding experiments.

Our metrics have been proven useful in our set of experiments.

Rebuilds trust in Active Learning among industrial practitioners.

Cardinal helps reproducibility and metrics research

Features common query strategies in an experimental framework.

Do not hesitate to contribute!

(... also, we are hiring.)

Thanks for your attention. Any questions?

References

[Smailagic 2018] Smailagic, Asim, et al. "MedAL: Accurate and robust deep active learning for medical image analysis." 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA). IEEE, 2018.

[Li 2012] Li, Chun-Liang, Chun-Sung Ferng, and Hsuan-Tien Lin. "Active learning with hinted support vector machine." Asian Conference on Machine Learning. 2012.

[Kottke 2017] Kottke, Daniel, et al. "Challenges of reliable, realistic and comparable active learning evaluation." Proceedings of the Workshop and Tutorial on Interactive Adaptive Learning. 2017.

[Du 2015] Du, Bo, et al. "Exploring representativeness and informativeness for active learning." IEEE transactions on cybernetics 47.1 (2015): 14-26.

[Du 2020] Munjal, Prateek, et al. "Towards Robust and Reproducible Active Learning Using Neural Networks." arXiv (2020): arXiv-2002.

[Sener 2017] Sener, Ozan, and Silvio Savarese. "Active learning for convolutional neural networks: A core-set approach." arXiv preprint arXiv:1708.00489 (2017).

References

[Yoo 2019] Yoo, Donggeun, and In So Kweon. "Learning loss for active learning." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019.

[Zhdanov 2019] Zhdanov, Fedor. "Diverse mini-batch Active Learning." arXiv preprint arXiv:1901.05954 (2019).

[Abraham 2020] Abraham, Alexandre, et al.. “Rebuilding Trust in Active Learning with Actionable Metrics” International Conference on Data Mining, IncrLearn workshop. 2020.

[Huang 2010] Huang, Sheng-Jun, Rong Jin, and Zhi-Hua Zhou. "Active learning by querying informative and representative examples." Advances in neural information processing systems. 2010.

[Ghayoomi 2010] Ghayoomi, Masood. "Using variance as a stopping criterion for active learning of frame assignment." Proceedings of the NAACL HLT 2010 Workshop on Active Learning for Natural Language Processing. 2010.

[Vlachos 2008] Vlachos, Andreas. "A stopping criterion for active learning." Computer Speech & Language 22.3 (2008): 295-312.

[Settles 2009] Settles, Burr. Active learning literature survey. University of Wisconsin-Madison Department of Computer Sciences, 2009.

About us…

What is Dataiku? (🦄)

Dataiku is the platform democratizing access to data and enabling enterprises to build their own path to AI.

450+ employees across the globe

300+ customers across industries

120% annual growth

$100 million Series D funding round (Aug 2020)

Why join us? (Yes, we're hiring!)

Being a tech company doesn't mean it's all about technology and processes.
At Dataiku, we truly believe that people (including our people!) are a critical piece of the equation.

💡Learn 🌱Grow 💥Make an impact

and much more… 🧘 🥳 🎮 🛫 🍕

Estimating uncertainty

Classification uncertainty

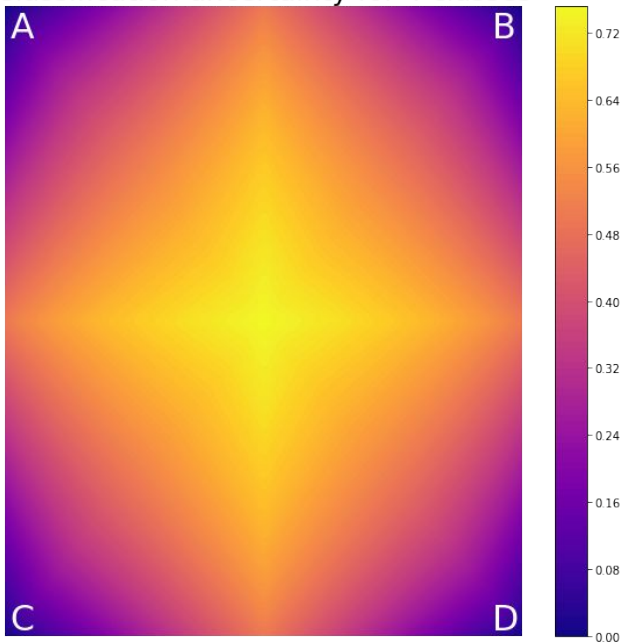
$$U(x) = 1 - P(\hat{x}|x)$$

$\hat{x}$ being the most likely class

Probability of selecting a sample depends on how confident the classifier is.

A	B	C	D	U
0.1	0.2	0.3	**0.4**	**0.6**
0.0	0.05	0.55	0.6	0.4
0.15	0.45	0.2	0.2	0.55

Classification uncertainty for 4 classes



Estimating uncertainty

Classification margin

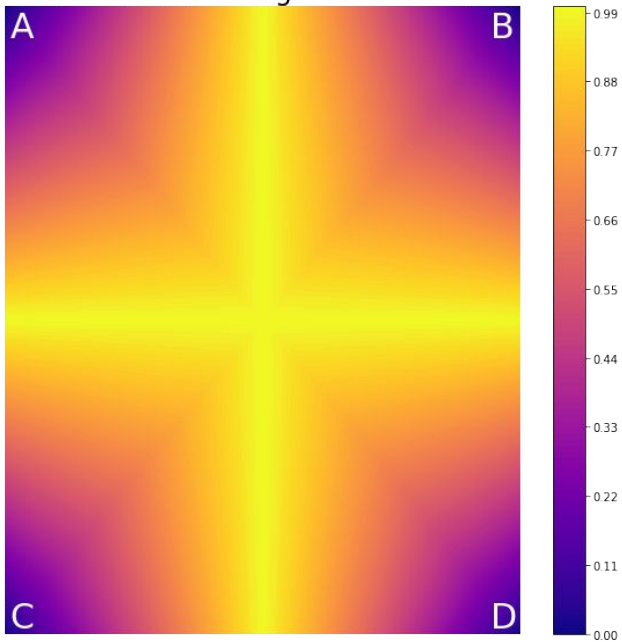
$$M(x) = 1 - (P(\hat{x}_1|x) - P(\hat{x}_2|x))$$

$\hat{x}_1, \hat{x}_2$ being the two most likely classes

Margin is the difference between the first and second class probabilities.

A	B	C	D	M
0.1	0.2	0.3	0.4	0.9
0.0	0.05	**0.55**	**0.6**	**0.95**
0.15	0.45	0.2	0.2	0.75

Classification margin for 4 classes



Estimating uncertainty

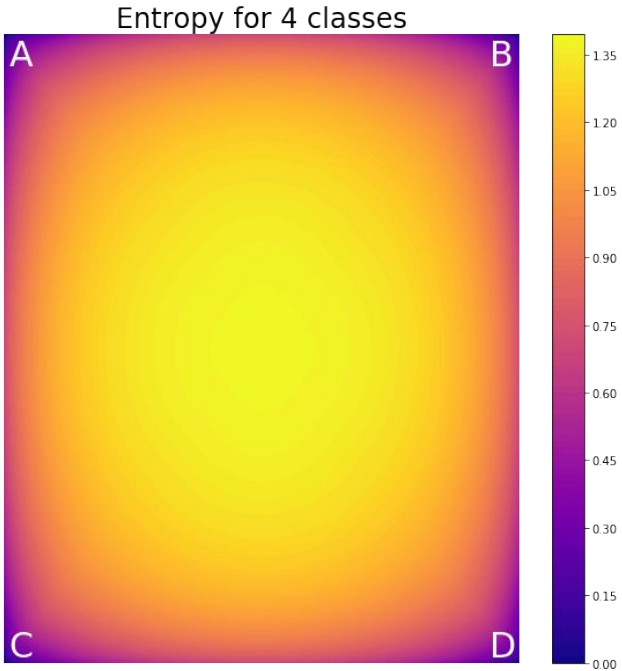
Entropy

$$H(x) = \frac{-\sum_k P(\hat{x}_k|x)\ log(P(\hat{x}_k|x))}{-\sum_k 1/k\ log(1/k)}$$

$\hat{x}_k$ being the *k*th class

Entropy of probability values.

A	B	C	D	H
0.1	0.2	0.3	0.4	0.92
0.0	0.05	0.55	0.6	0.60
0.15	0.45	0.2	0.2	**0.93**

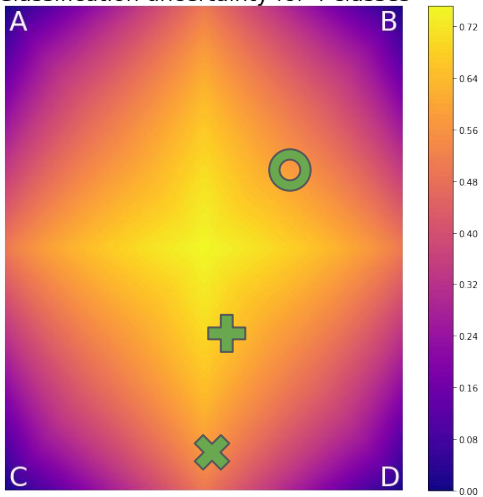


Estimating uncertainty

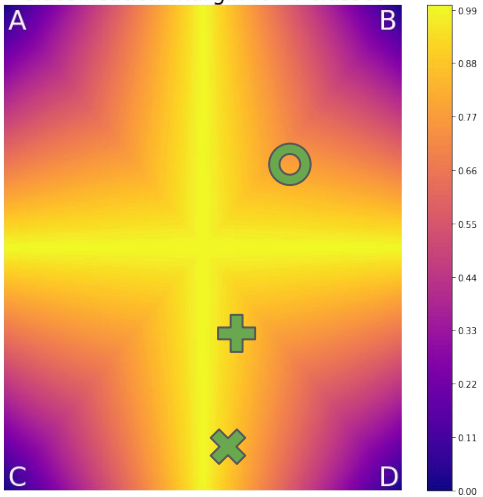
Graphical representation

A	B	C	D	
0.1	0.2	0.3	0.4	✚
0.0	0.05	0.55	0.6	✖
0.15	0.45	0.2	0.2	○

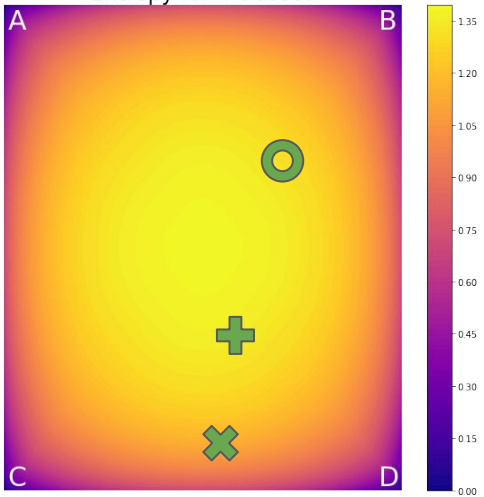
Classification uncertainty for 4 classes



Classification margin for 4 classes

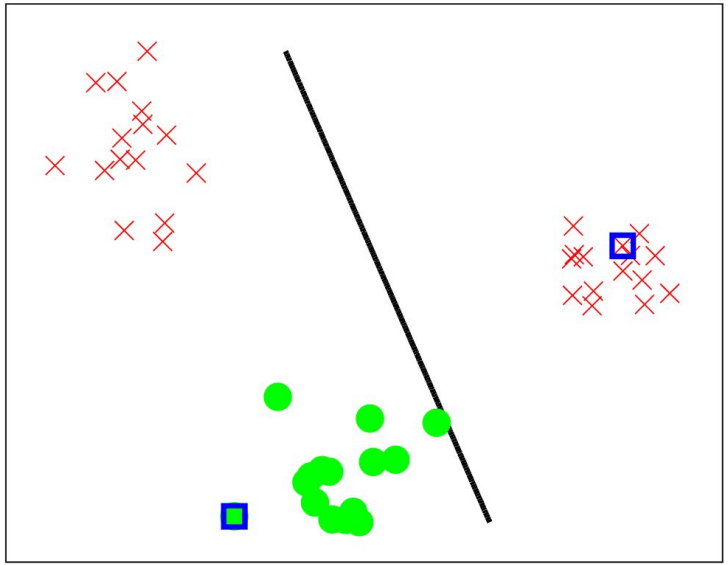


Entropy for 4 classes

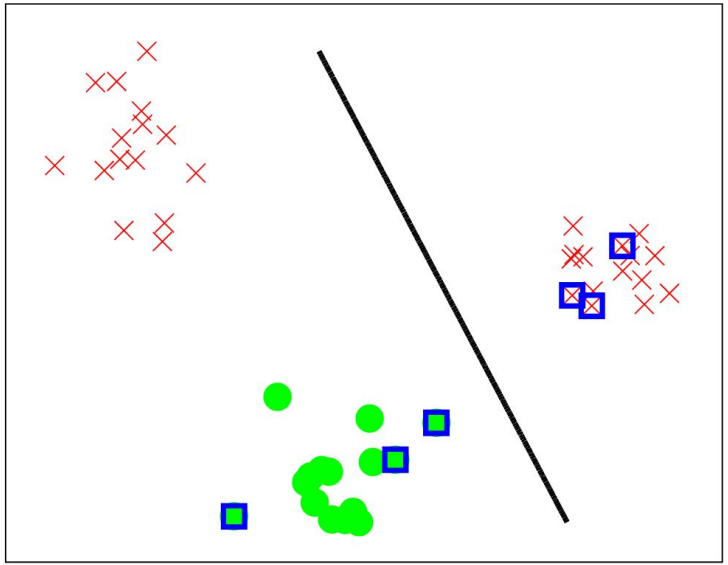


Active Learning with Hinted Support Vector Machine [Li 2012]

SVM based uncertainty can ignore a large set of unlabeled samples.

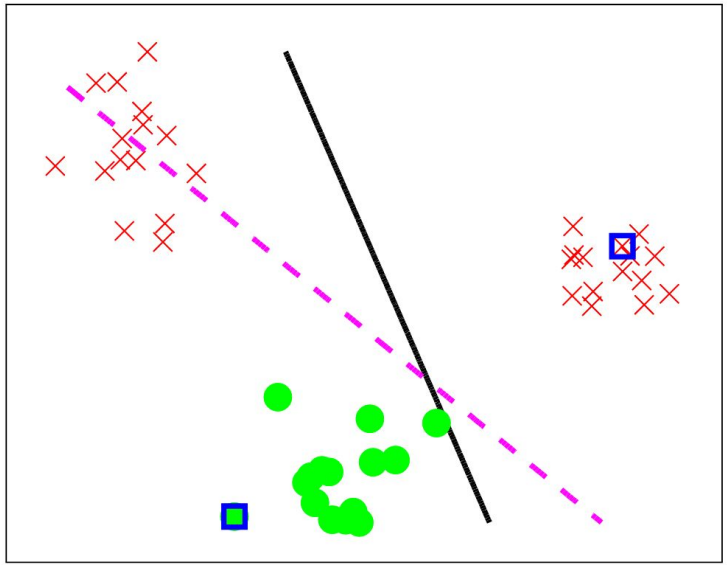


(a)

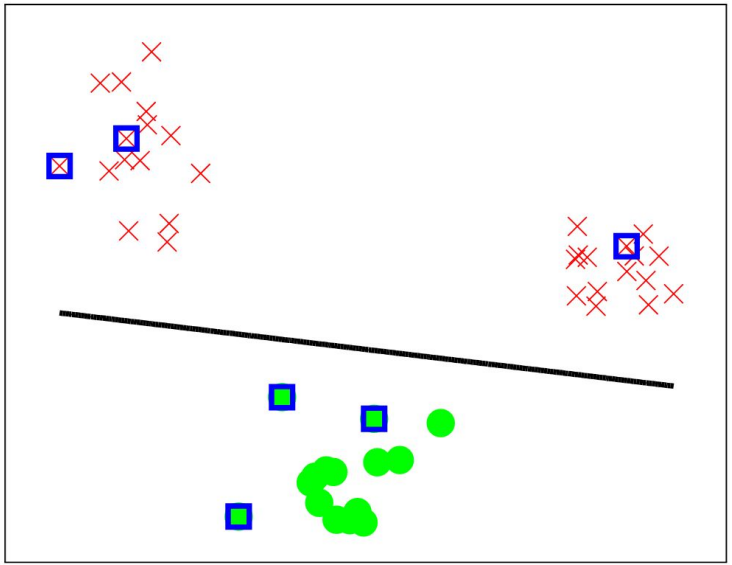


(b)

Active Learning with Hinted Support Vector Machine [Li 2012]

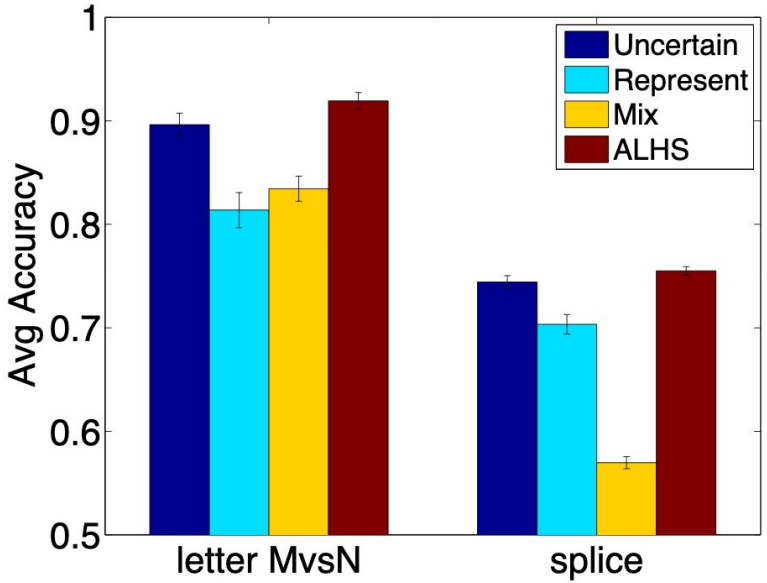


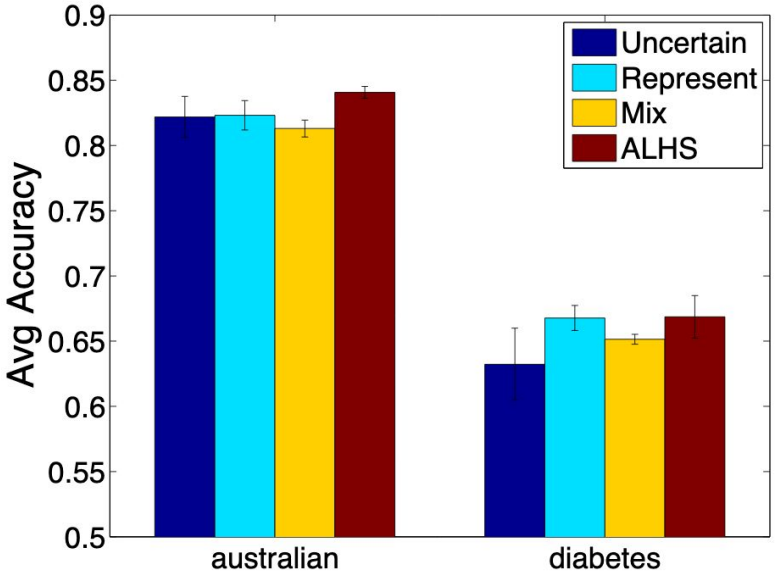
(a)



(b)

Active Learning with Hinted Support Vector Machine [Li 2012]





Exploring Representativeness and Informativeness for Active Learning [Du 2015]

Let S: $(i,j) \rightarrow \mathbb{R}$ be a similarity function between two samples and C be a certainty score vector. We compute the binary vector α that indicates which unlabeled samples should be queried.

We minimize the similarity between queried samples:

$$M_1(i,j) = \frac{1}{2}S(i,j)$$

The similarity between queries and labeled samples:

$$M_2(i) = \frac{n_t + 1}{n}\sum_{j=1}^{n_t} S(i,j)$$

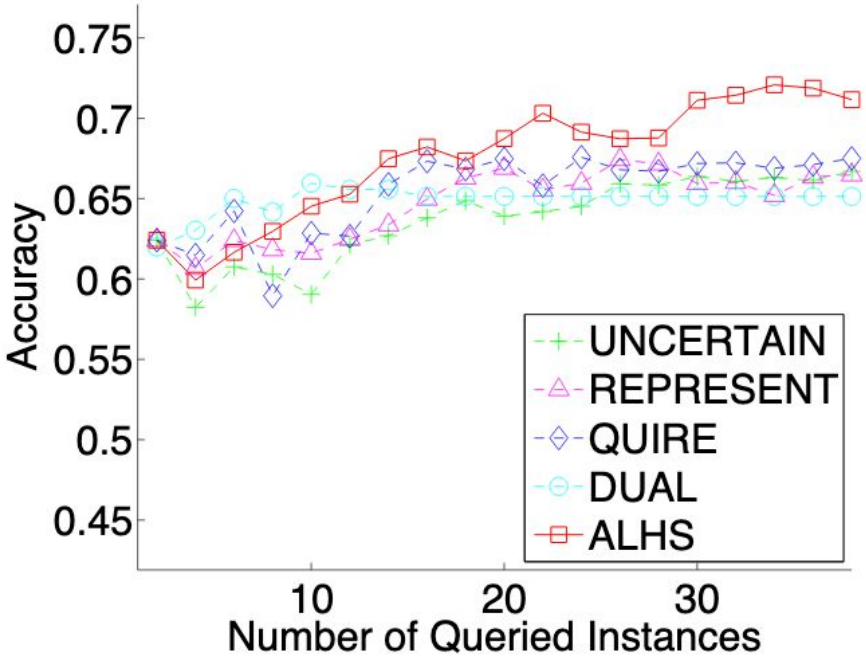
And we maximize the similarity between queries and unlabeled:

$$M_3(i) = \frac{u_t - 1}{n}\sum_{j=1}^{u_t} S(i,j)$$

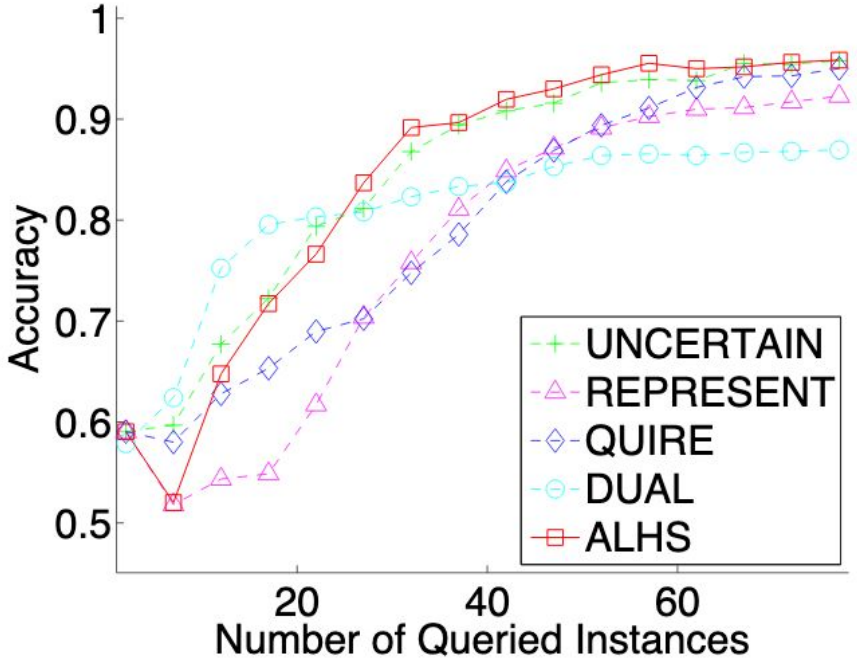
We then integrate this in a global minimization:

$$\min_{\alpha} \alpha^T M_1 \alpha + \alpha^T (M_2 - M_3) + \beta C$$
$$s.t. \quad \alpha^T 1^{u_t} = 1, \alpha_i \in [0,1]$$

Active Learning with Hinted Support Vector Machine [Li 2012]



(c) *diabetes*

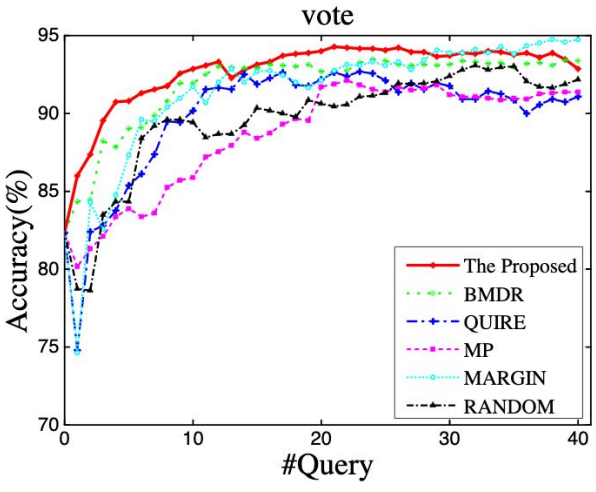


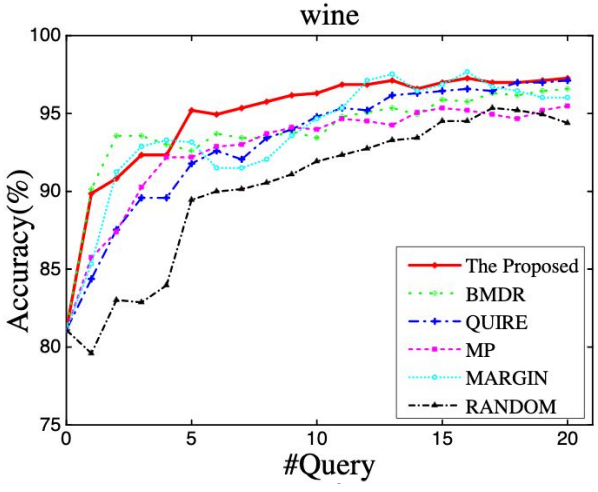
(d) *leterMvsN*

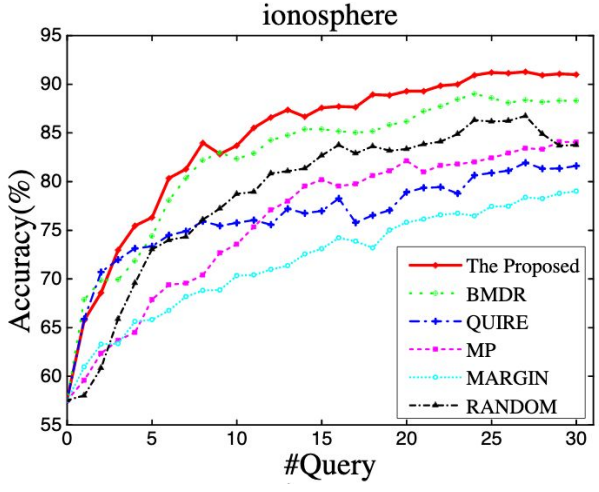
Active Learning with Hinted Support Vector Machine [Li 2012]

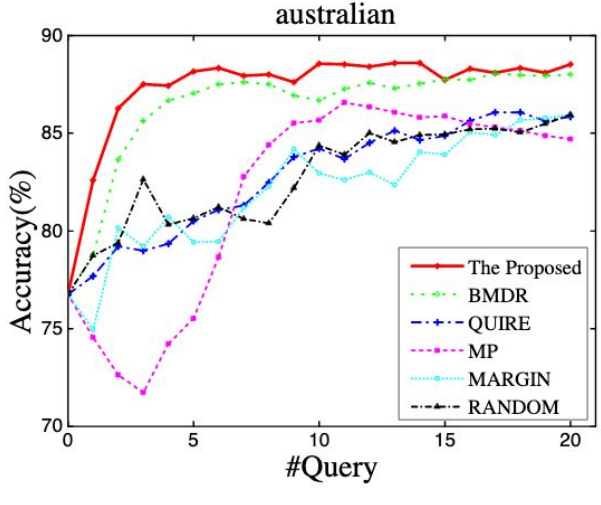
Table 1: Comparison on accuracy (mean ± se) after querying 5% of unlabeled pool

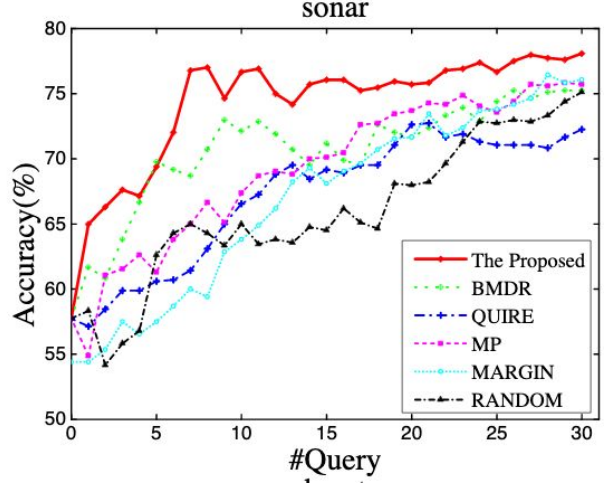
	Algorithms (%), the highest accuracy for each dataset is in boldface				
data	UNCERTAIN	REPRESENT	QUIRE	DUAL	ALHS
australian	82.188 ± 1.571	83.739 ± 0.548	82.319 ± 1.126	81.304 ± 0.647	$\mathbf{84.072 \pm 0.454}$
breast	96.334 ± 0.278	95.264 ± 0.439	$\mathbf{96.657 \pm 0.187}$	96.408 ± 0.196	96.525 ± 0.219
diabetes	63.229 ± 2.767	66.758 ± 0.505	66.771 ± 0.960	65.143 ± 0.381	$\mathbf{66.862 \pm 1.632}$
german	69.060 ± 0.497	67.240 ± 1.099	68.750 ± 0.605	69.620 ± 0.323	$\mathbf{69.750 \pm 0.349}$
letterMvsN	89.632 ± 1.103	83.463 ± 1.348	81.372 ± 1.693	83.437 ± 1.211	$\mathbf{91.919 \pm 0.812}$
letterVvsY	79.245 ± 1.176	63.523 ± 2.335	68.516 ± 2.132	76.213 ± 1.549	$\mathbf{79.381 \pm 1.174}$
segment	95.437 ± 0.367	94.390 ± 0.482	96.074 ± 0.224	86.078 ± 2.834	$\mathbf{96.095 \pm 0.204}$
splice	74.430 ± 0.606	69.117 ± 1.452	70.340 ± 0.942	56.969 ± 0.576	$\mathbf{75.506 \pm 0.403}$
wdbc	93.842 ± 3.137	95.616 ± 0.711	96.613 ± 0.230	96.056 ± 0.250	$\mathbf{96.921 \pm 0.200}$

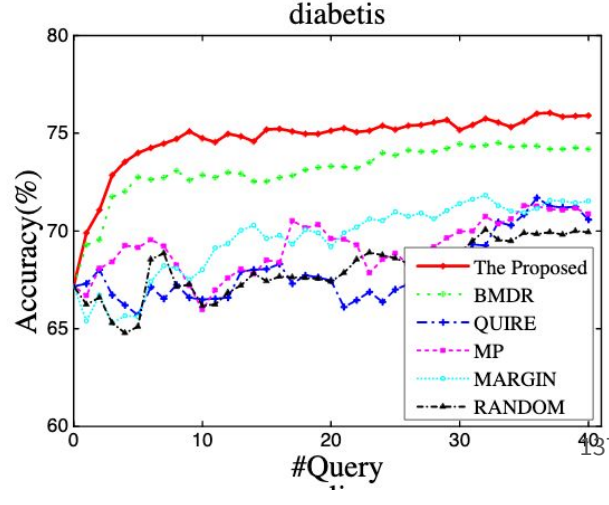

vote
Accuracy(%)
#Query
The Proposed
BMDR
QUIRE
MP
MARGIN
RANDOM


wine
Accuracy(%)
#Query
The Proposed
BMDR
QUIRE
MP
MARGIN
RANDOM


ionosphere
Accuracy(%)
#Query
The Proposed
BMDR
QUIRE
MP
MARGIN
RANDOM


australian
Accuracy(%)
#Query
The Proposed
BMDR
QUIRE
MP
MARGIN
RANDOM


sonar
Accuracy(%)
#Query
The Proposed
BMDR
QUIRE
MP
MARGIN
RANDOM

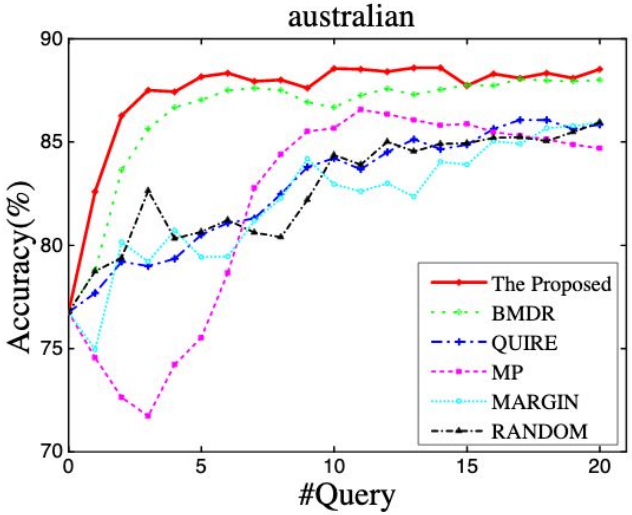

diabetis
Accuracy(%)
#Query
The Proposed
BMDR
QUIRE
MP
MARGIN
RANDOM

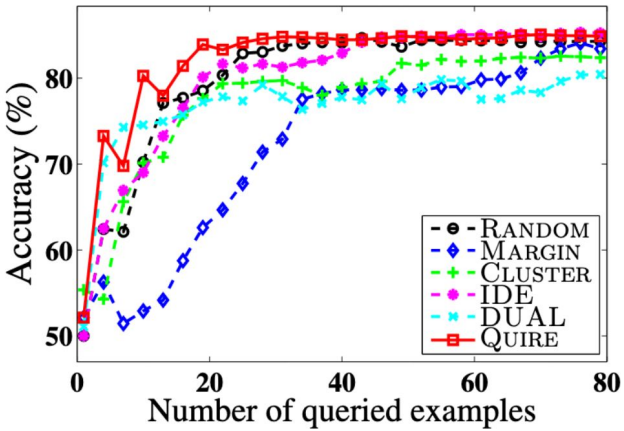
Comparing results between studies

What is the best option?

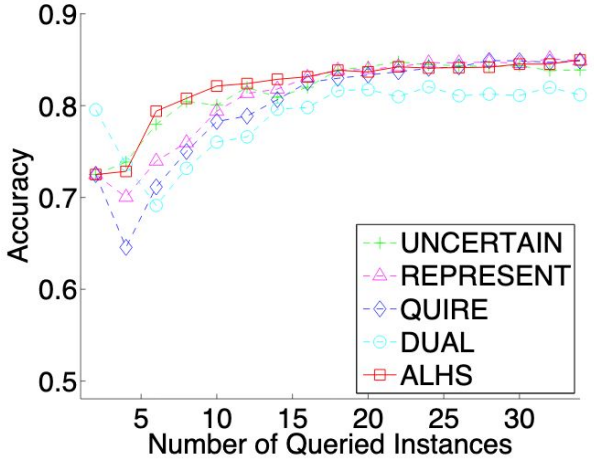
Top right is [Du 2015]

Bottom right is [Li 2012]



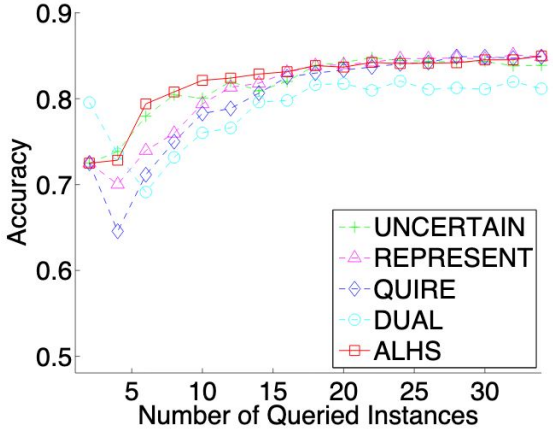


(a) *austra*

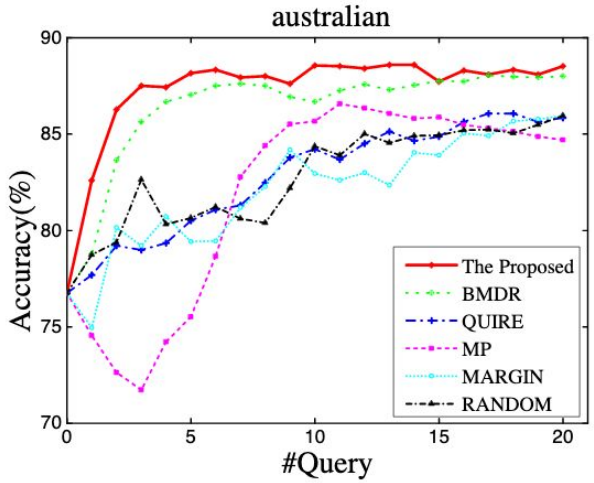


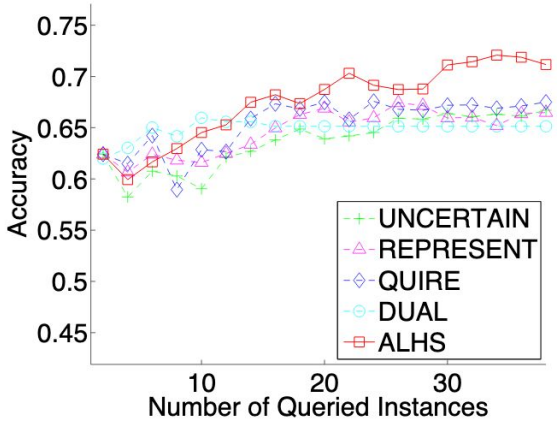
(a) *australian*

Comparing results between studies

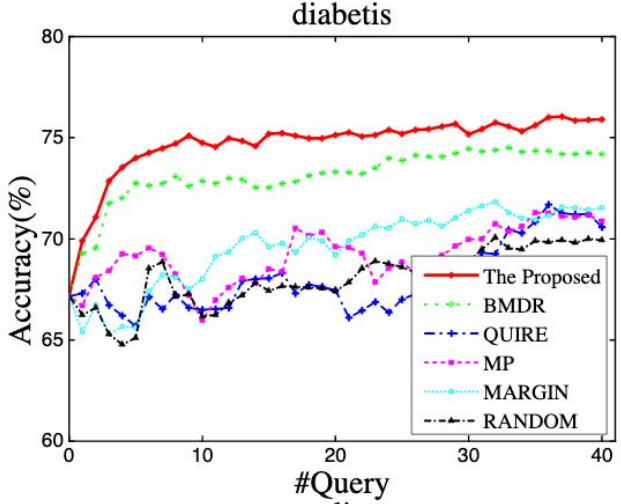


(a) *australian*



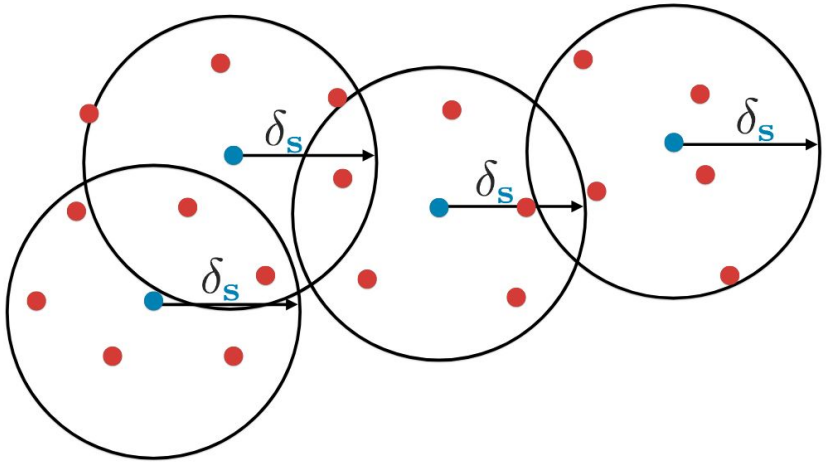


(c) *diabetes*



Elhamifar et al. 2018

Active Learning for Convolutional Neural Networks: A Core-Set Approach



Algorithm 1 k-Center-Greedy

Input: data $\mathbf{x}_i$, existing pool $\mathbf{s}^0$ and a budget b
Initialize $\mathbf{s} = \mathbf{s}^0$
repeat
  $u = \arg\max_{i \in [n] \setminus \mathbf{s}} \min_{j \in \mathbf{s}} \Delta(\mathbf{x}_i, \mathbf{x}_j)$
  $\mathbf{s} = \mathbf{s} \cup \{u\}$
until $|\mathbf{s}| = b + |\mathbf{s}^0|$
return $\mathbf{s} \setminus \mathbf{s}^0$

Elhamifar et al. 2018

Active Learning for Convolutional Neural Networks: A Core-Set Approach

Algorithm 2 Robust k-Center

Input: data $\mathbf{x}_i$, existing pool $\mathbf{s}^0$, budget b and outlier bound Ξ
Initialize $\mathbf{s}_g = \text{k-Center-Greedy}(\mathbf{x}_i, \mathbf{s}^0, b)$
$\delta_{2-OPT} = \max_j \min_{i \in \mathbf{s}_g} \Delta(\mathbf{x}_i, \mathbf{x}_j)$
$lb = \frac{\delta_{2-OPT}}{2}$, $ub = \delta_{2-OPT}$
repeat
  if $Feasible(b, \mathbf{s}^0, \frac{lb+ub}{2}, \Xi)$ **then**
    $ub = \max_{i,j|\Delta(\mathbf{x}_i,\mathbf{x}_j) \leq \frac{lb+ub}{2}} \Delta(\mathbf{x}_i, \mathbf{x}_j)$
  else
    $lb = \min_{i,j|\Delta(\mathbf{x}_i,\mathbf{x}_j) \geq \frac{lb+ub}{2}} \Delta(\mathbf{x}_i, \mathbf{x}_j)$
  end if
until $ub = lb$
return $\{i \; s.t. \; u_i = 1\}$

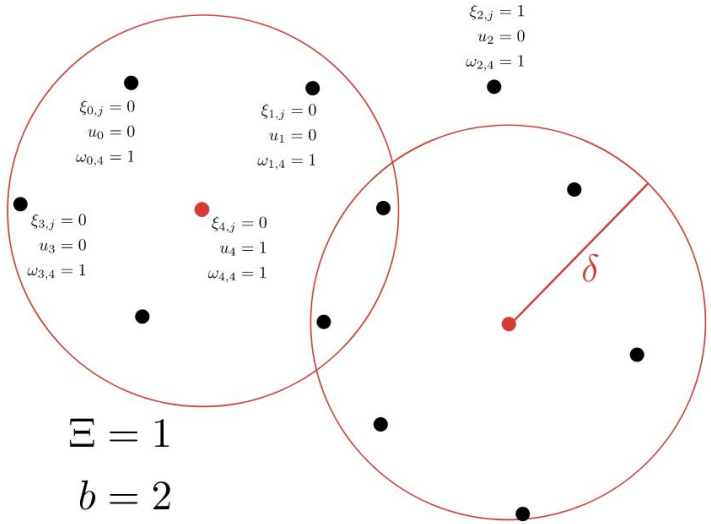


Figure 2: Visualizations of the variables. In this solution, the 4^{th} node is chosen as a center and nodes $0, 1, 3$ are in a δ ball around it. The 2^{nd} node is marked as an outlier.

Metrics for Active Learning

Hard to classify samples

A batch of samples is selected

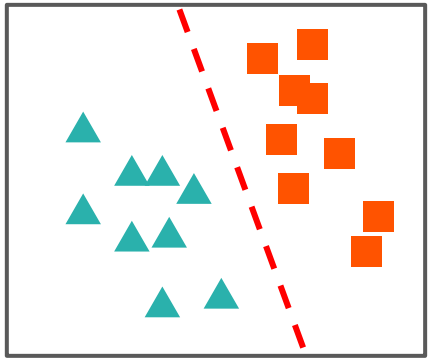
To evaluate them, we use a reference classifier trained on left out data

We predict on the selected batch

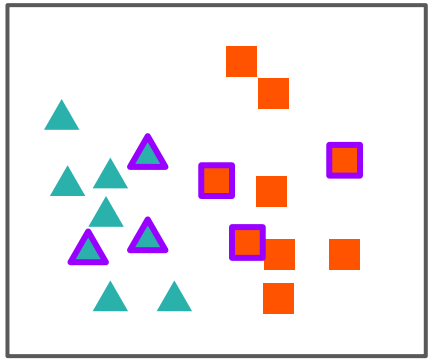
⇢ 66% accuracy

They are honey pots for uncertainty methods!

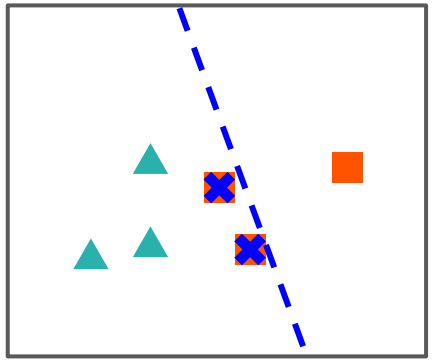
Test set



Train set



Selected batch



Elhamifar et al. 2018

Active Learning for Convolutional Neural Networks: A Core-Set Approach

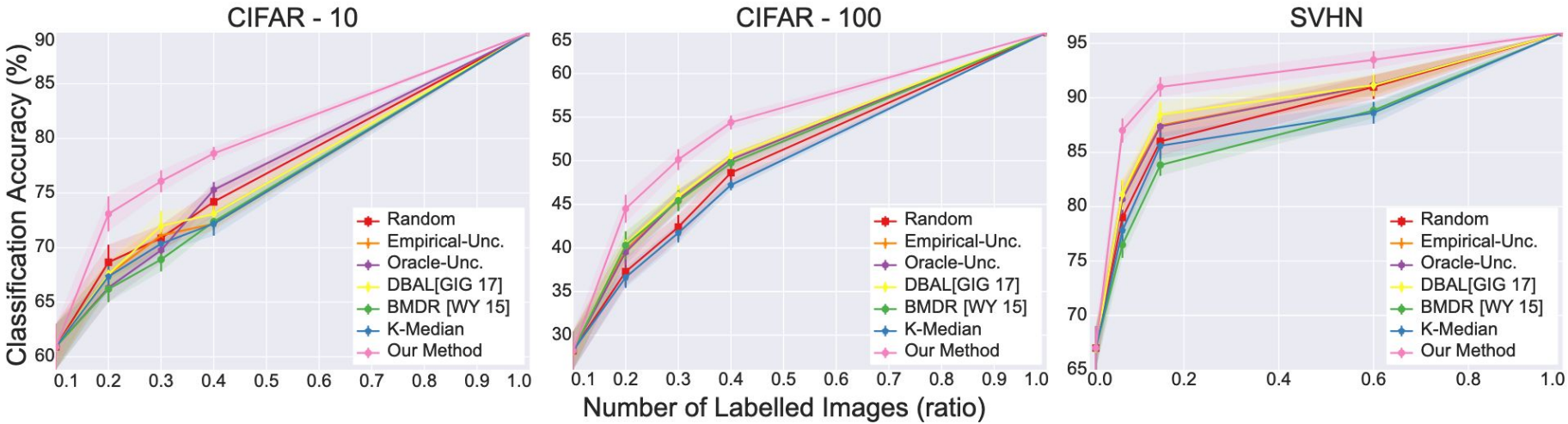


Figure 4: Results on Active Learning for Fully-Supervised Model (error bars are std-dev)

A realistic setting for Active Learning experiments

How can we compare query strategies?

Is there a way to do it in a real life setting?

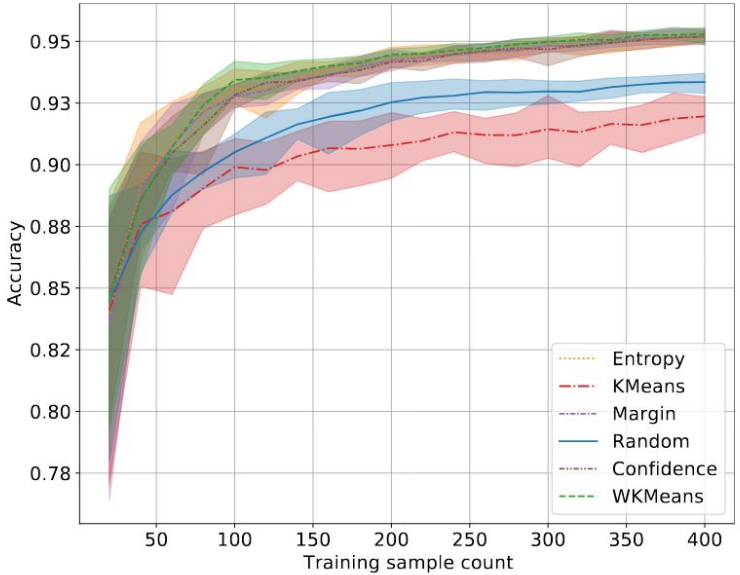


Fig. 1. Comparison of strategies on **NOMAO** dataset

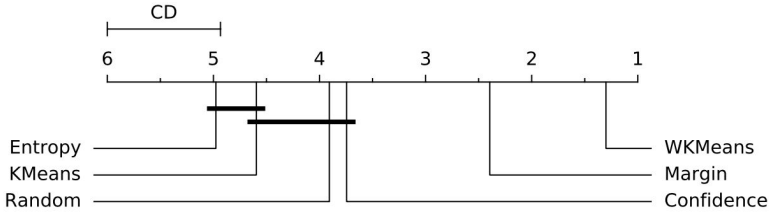


Fig. 3. Comparison of methods on hard tasks

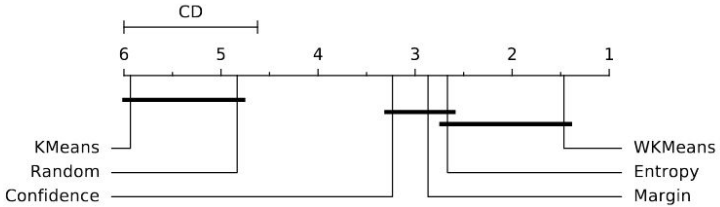


Fig. 2. Comparison of methods on easy tasks

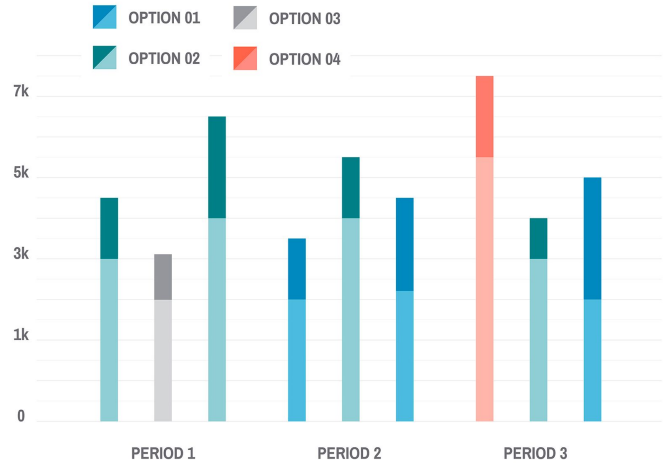
Introduction to Active Learning

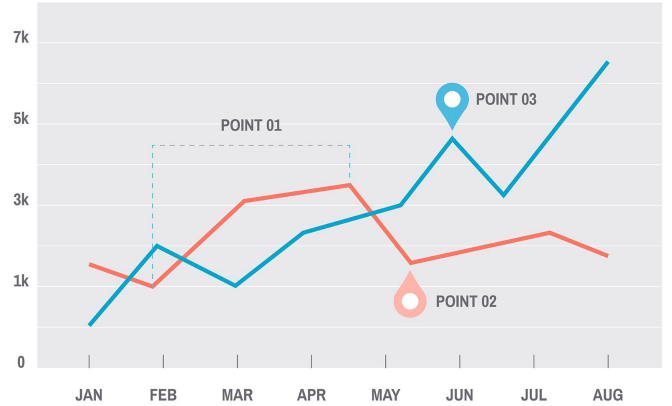
The need for better insights

What is the best method?

Moreover, when facing a new one-shot experiment, which method to choose?

We need real-time insights to take better decisions.





The Python Package cardinal

The Python Package cardinal

Active Learning Python package landscape

modAL

- Minimalist design
- Focus on ensemble approaches
- Provides an object ActiveLearner to run experiments with in-memory caching

AliPy

- Created in Nanjing University of Aeronautics and Astronautics
- Exclusive methods: Active Learning from Data, Self-paced Active Learning
- AlExperiment object ease experiments with disk caching but no replay

Libact

- Performance oriented
- Exclusive feature: Learning active learning by learning
- Utilities provided for experiment, such as noisy labeler, but the main loop is left to the user

Learn more in our blog post

The Python Package cardinal

Cardinal's philosophy

A package focused on experiments and metrics

- Tried and tested query strategies and metrics
- Numerous and detailed examples and experiments
- Ease research in active learning metrics by allowing experiment replay
- Human readable cache

So far in the package

- Classical query strategies
- Two-step query strategy from [Zhdanov 2019]
- Introduction to Active Learning and examples displaying the importance of exploration

The Python Package cardinal

Query strategy interface

BaseQuerySampler

```
init(batch_size)
fit(X, y=None)
select_samples(X)
```

ScoredQuerySampler

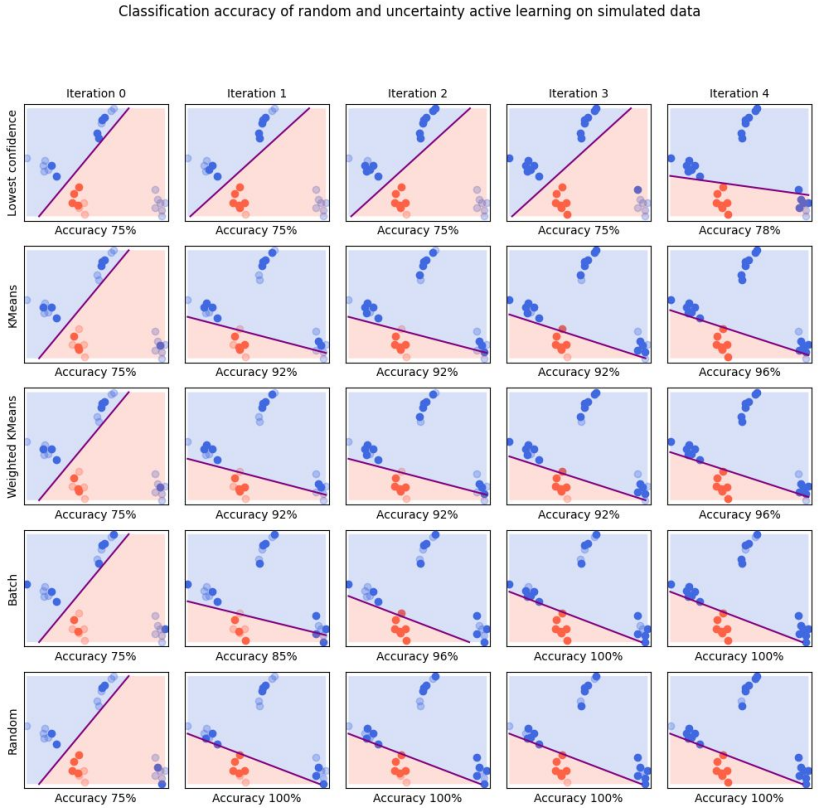
```
score_samples(X)
sample_scores_
```

MarginSampler
UncertaintySampler
EntropySampler

KCentroidSampler

```
init(batch_size)
score_samples(X, sample_weight)
```

KMeansSampler
MiniBatchKMeansSampler



The Python Package cardinal

Experimenting with metrics

Not final until version 1.0

```
X_train, y_train, X_test, y_test = get_dataset()

config = dict(method='margin')
clf = RandomForest()
sampler = MarginSampler(clf, batch_size=10)
batch_sizes = [10, 20, 30]

with ReplayCache('./cache', keys=config) as cache:
    selected = cache.variable('selected', [])
    for batch_size, prev_selected in cache.iter(batch_sizes, selected.previous()):
        clf.fit(X[prev_selected], y[prev_selected])
        cache.log_value('accuracy', clf.score(X_test, y_test))

        sampler.fit(X[prev_selected], y[prev_selected])
        selected.set(prev_selected + sampler.select_samples())

    cache.compute_metrics(contradictions,
                          selected.previous(),
                          selected.current())
```

The ReplayCache stores all variables of all iterations to replay the experiment. ResumeCache only runs once.

Cached variable values are stored across time in the experiment.

The logging tool logs values in a database for easy plotting.

Metrics can be computed afterward thanks to cached data. Formalism follows the same as the cached loop.

The Python Package cardinal

Caching system

On disk

```
$ tree cache/method/margin
cache/method/margin
└── iter
    ├── 0
    │   └── selected
    ├── 1
    │   └── selected
    ├── 10
    │   └── selected
    ├── 2
    │   └── selected
    ├── 3
    │   └── selected
    ├── 4
    │   └── selected
    ├── 5
    │   └── selected
    ⋮
```

In database

```
In [1]: import dataset

In [2]: import pandas as pd

In [3]: table = dataset.connect('sqlite:///cache.db')['accuracy']

In [4]: pd.DataFrame(list(table.all()))
Out[4]:
   id strategy  iter  accuracy
0   1   margin    0      0.51
1   2   margin    1      0.54
2   3   margin    2      0.61
3   4   margin    3      0.65
4   5   margin    4      0.70
5   6   margin    5      0.80
6   7   margin    6      0.84
7   8   margin    7      0.88
8   9   margin    8      0.89
9  10   margin    9      0.90
```

The Python Package cardinal

Diverse mini-batch Active Learning [Zhdanov 2019]

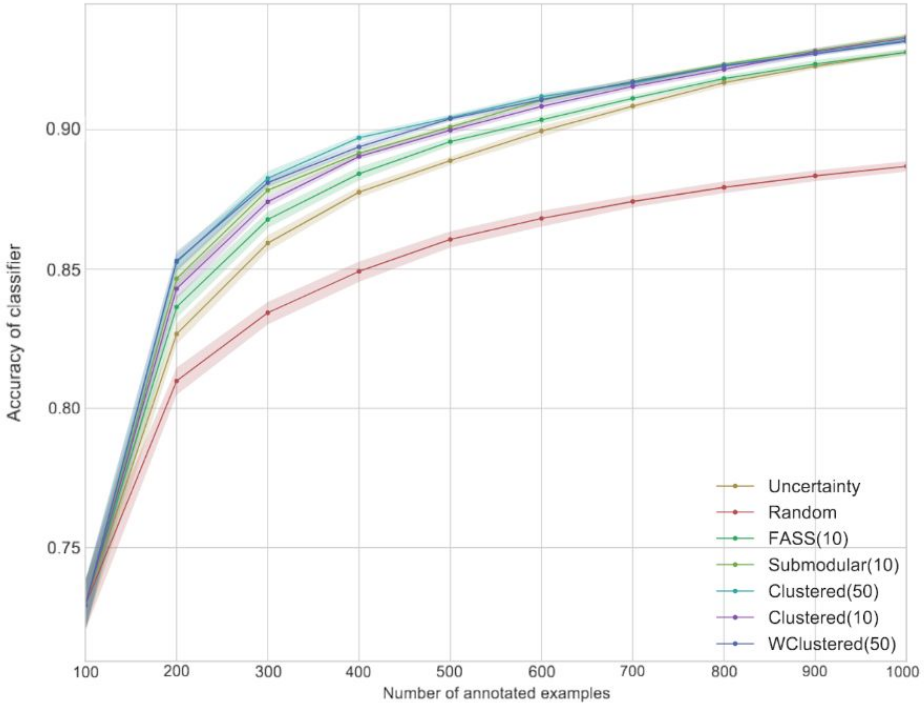
Input:

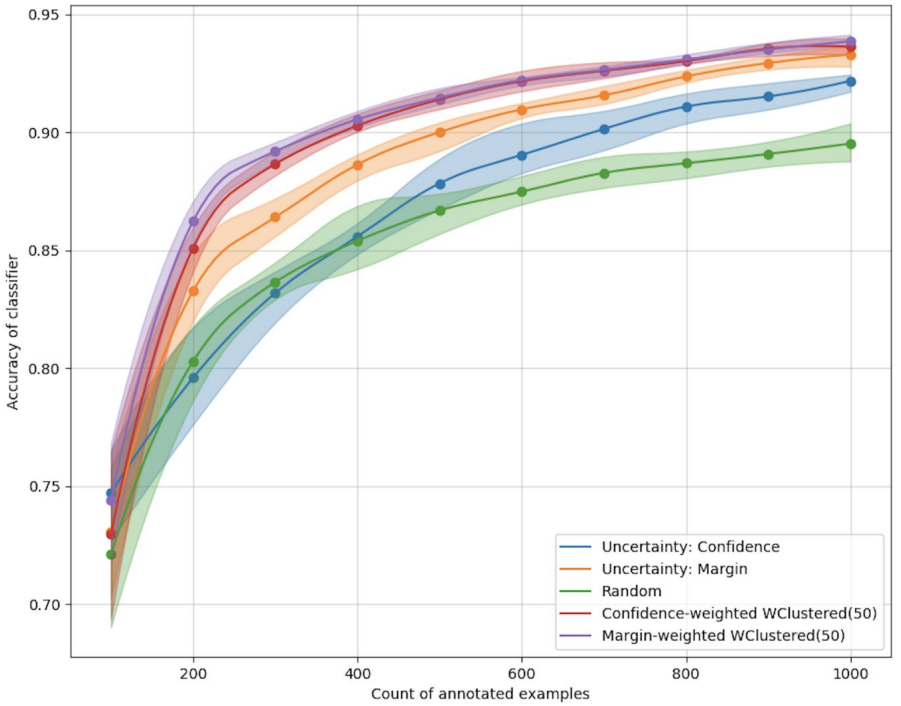
- dataset of examples
- budget B
- batch-size k
- pre-filter factor β

Select first k examples randomly, obtain labels for these examples
repeat
  Train classifier on all the examples selected so far
  Get informativeness for every unlabeled examples
  Prefilter to top βk informative examples
  Cluster βk examples to k clusters with (weighted) K-means
  Select k different examples closest to the cluster centers,
    obtain labels for these examples
until Budget B is exhausted

The Python Package cardinal

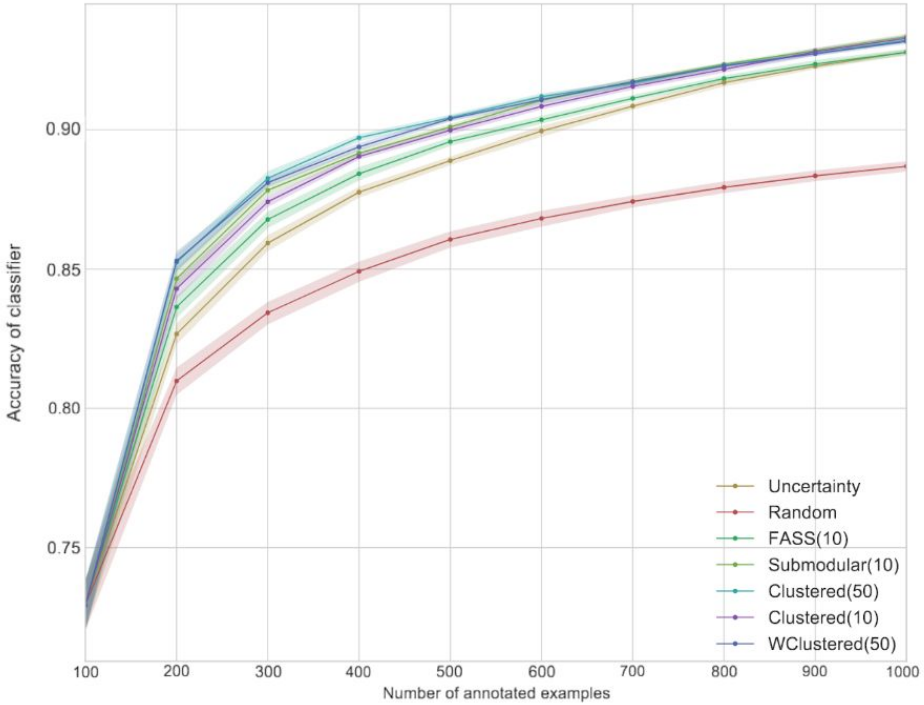
Diverse mini-batch Active Learning [Zhdanov 2019]

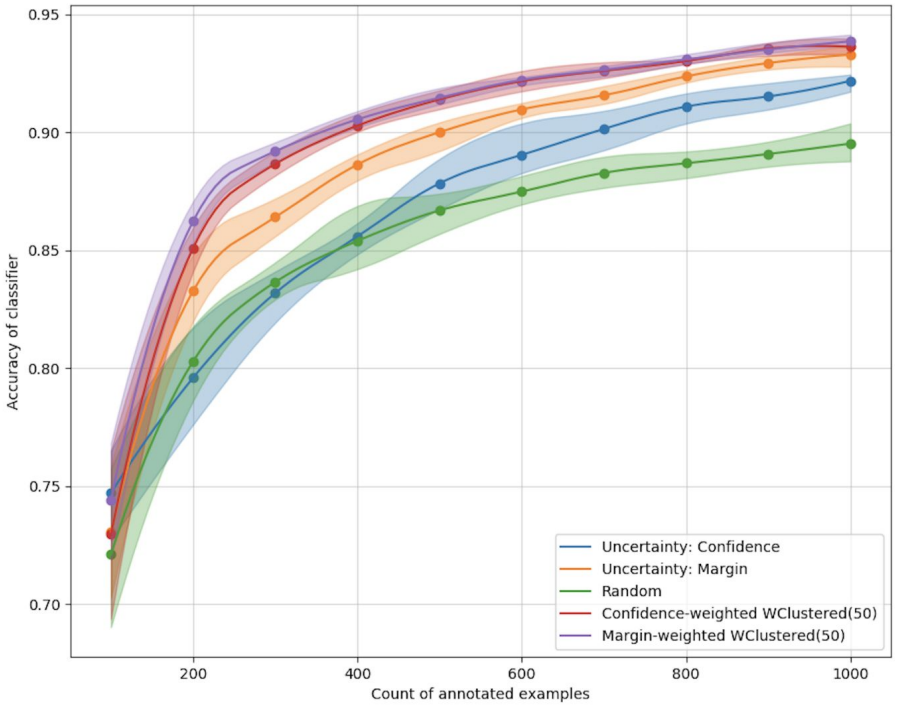




The Python Package cardinal

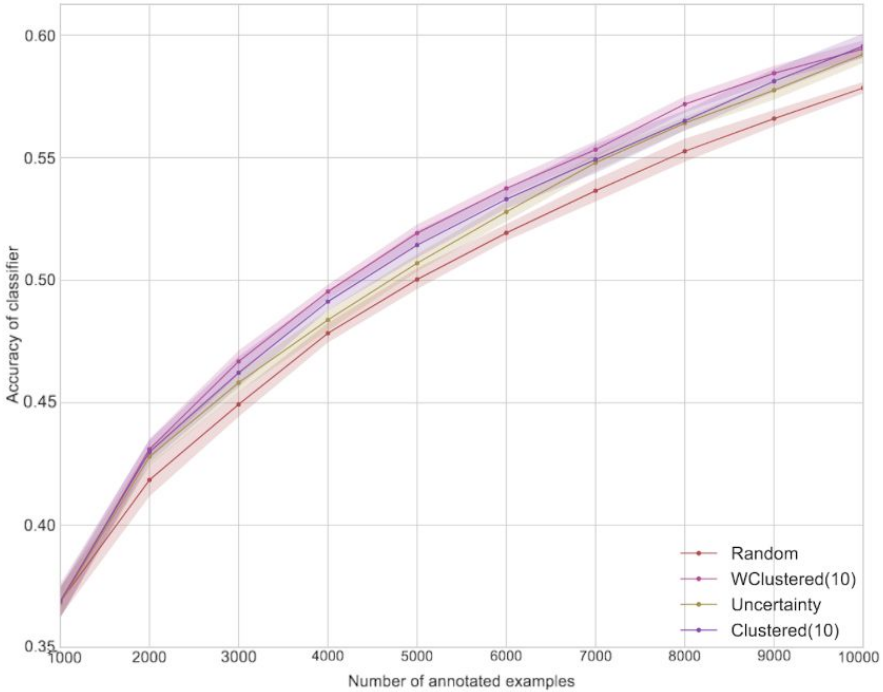
Diverse mini-batch Active Learning [Zhdanov 2019]

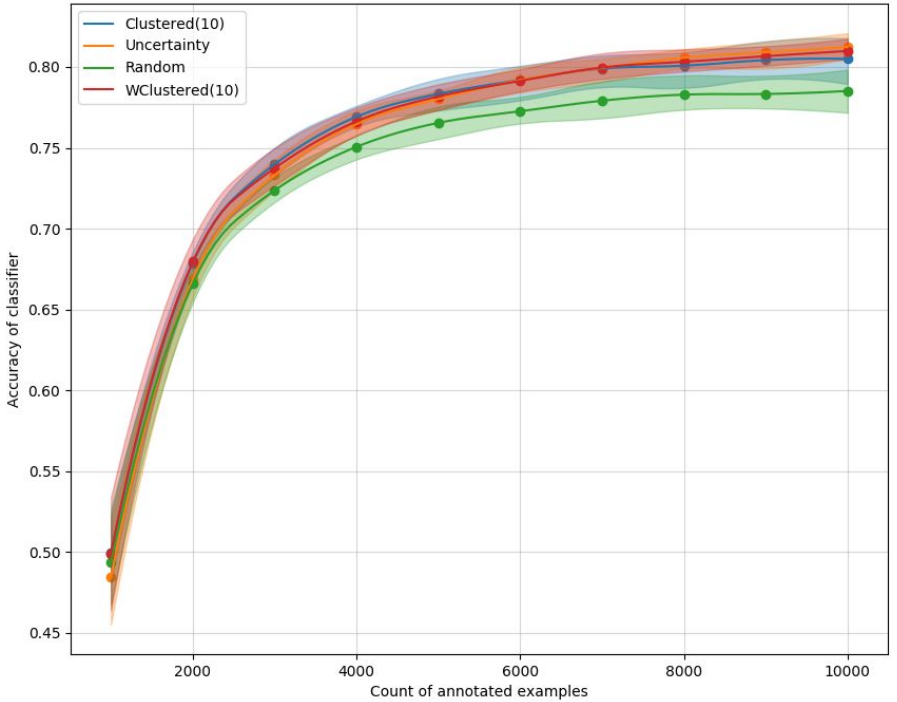




The Python Package cardinal

Diverse mini-batch Active Learning [Zhdanov 2019]





Learn more in our [blog post]

Computing metrics easily

Active Learning experiments

- are based on a for loop
- can be very costly

We want a package that

- make experiments easier to write
- allows to resume in case of error
- **allows to replay experiments to ease reproducibility and metrics research**

cardinal