Scalable batch mode Optimal Experimental Design for Deep Networks

Melanie Ducoffe, Geoffrey Portelli & Frederic Precioso
Univ. Nice Sophia Antipolis,
I3S, UMR UNS-CNRS 7271
06900 Sophia Antipolis, France
{ducoffe, portelli, precioso}@i3s.unice.fr

Abstract

Current trend in machine learning is the use of deep neural networks due to their outstanding accuracy on various challenging problems. There are evidences that because of the high number of parameters involved deep learning will need many samples to be labeled compared to other shallower classifiers. In the context where gathering and especially annotating data to train those architectures becomes more and more expensive, active learning which automatically select samples to be labeled for faster convergence, is well motivated. Thus we took interest in a recent field called batch active learning to query more than one sample at a time. Such techniques will help limiting the number of calls of active learning. Unfortunately most of the litterature about batch active learning deals with the computation of the Fisher matrix which is intractable for deep networks. Following the Optimal Experimental Design fashion, we adapt and scale for deep networks a batch active method called A-optimality which has been previously proposed for logistic regression. Our method relies on a computationally tractable approximation of the Fisher matrix based on a recent work on a kronecker product representation. We show that our scalable batch active learning technique designed for deep networks is relevant and promising. We also demonstrate the efficiency of our approach on two benchmarks datasets, MNIST and SVHN.

1 Introduction

Gathering huge amount of unlabeled data becomes more and more easy, however labeling them may be expensive especially when the data in question requires human experts as annotators. Active learning is a type of iterative supervised learning in which the goal is to successively and automatically query the labels of as few as possible unlabeled data in order to reduce the generalization error.

To handle the scalability of active learning for deep architectures we step away from traditional PAC method and focus on a more general setting: Maximum Likelihood Estimation (MLE).

A wide variety of active MLE estimation, notably under the framework of Optimal Experimental Design (OED) has been proposed [4]. All of them aim at selecting samples to be labeled by minimizing the variance of the estimator. Indeed this minimization is equivalent to the maximization of the information. However in a multivariate parameter setting, the variance corresponds to the inverse of a particular covariance matrix - called the Fisher matrix. Thus the *minimization* of this Fisher Matrix can be seen as an optimization through several scalar summary statistics such as the trace, the maximum eigenvalue or the determinant. Unfortunately, there is no consensus about which criterion rather than another should be applied regarding the active convergence rate or the solution quality; this is an open question.

29th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

Another issue lies in applying OED on neural networks, i.e managing the quadratic size of the Fisher matrix in terms of the number of parameters. With the growth of deep networks, such a constraint made OED statistics computationnaly unaffordable. However some recent works have suggested a compromise between an accurate representation of the Fisher matrix and its ressource cost. In particular R. Grosse in [10, 6] has proposed a representation of the Fisher matrix for deep networks with diagonal blocks matrix whose blocks are made of a kronecker product (KFAC, [10, 6]). The criterion balances wisely between tractability and quality of the approximation.

In this work we demonstrate the benefit of the KFAC representation to extend Fisher based active learning to deep networks. Especially we study the constraint, i.e. the quadratic size of the matrix, and whether or not the KFAC representation is suitable to lower the cost inherent to OED criterions. Especially the efficiency of a batch active method, called A-optimality is validated with experiments on MNIST and SVHN, using deep convolutional networks (CNNs). We also discuss about promising extensions to other OED framework.

2 Related work

Active learning is a framework to automatize the selection of instances to be labelled in a learning process. We consider the context of pool-based active learning where the learner selects its queries among a fixed unlabelled data set. For other variants (*query synthesis, selective sampling*) we refer the reader to the book of Burr Settles [12]. Many strategies have been developed to formulate the query process. We detail first the most general frameworks in the literature.

A general and intuitive approach is the idea of uncertainty sampling which queries the unlabelled instance on which the classifier is the least confident [9]. Extension has been made so to be more robust to the multivariate case and to encompass the uncertainty among several classes. If its implementation and its use is straightforward, finally the two main drawbacks of uncertainty sampling are its tendency to focus on specific classes on the input distribution and its bias towards outliers.

Another branch, theoretically more motivated is PAC active learning, which assumes there exists an optimal hypothesis in a known set of classifiers that correctly labels every examples. Hence PAC active learning requires realizability and a correct bayesian prior on the hypothesis space (*a zero mistakes classifiers exists for such a problem and is in our space of hypothesis*). PAC active learning scenarios involve evaluating the informativeness of unlabelled instances. For example, Query-By-Committee consists in sampling a committee in the hypothesis space and measures the disagreement of unlabelled samples among the committee prediction [13].

From another theoretical point of view, Optimal Experimental Design is a field which takes interests into the Fisher information. Formally the Fisher information is the expectation on the input data of the partial derivative of the log likelihood function with respect to the parameters. This measure is interesting because, in a single parameter case, its inverse sets a lower bound on the variance of the model's parameter estimates; this result is known as the Cramer Rao bound [8]. In other words, to minimize the variance over its parameter estimates, an active learner should select data that maximize the Fisher information, or minimize the inverse. But for multivariate parameters, the Fisher information is a covariance matrix, so its maximization may go through several statistics. We cite the three most popular scenarios (*other variants exist but, less used by the community, are left unlisted for the sake of clarity*):

- A-optimality minimizes the trace of the inverse information matrix [3]
- D-optimality minimizes the determinant of the inverse information matrix [2]
- E optimality minimizes the maximum eigenvalue of the information matrix [5]
- ...

In our work, batches of actively selected samples have the same size as the minibatches but they could be decorrelated by considering importance sampling techniques. It thus makes sense to consider also batch active learning to fasten convergence instead of increasing the training set by one sample at-a-time. Furthermore, owing to the important number of parameters of deep networks, adding one sample at-a-time would require too many active learning iterations and so labelling requests to the oracle, leading to a computationnally inefficient approach.

When it comes to selecting a batch of queries, the most intuitive solution is to select top scoring samples. Top score selection is immediate in the process but prones to select highly correlated samples which may slow down the convergence especially for large batches of queries.

Recently other solutions have been proposed for choosing an appropriate subset of samples so as to minimize any significant loss in performance. They consider the problem of correlation as the minimization of the Kullback-Leibler (KL) divergence between the resampled distribution induced by the training data selected by active learning and the distribution from the whole unlabelled dataset. They define a lower bound of the KL which may be rewritten as a problem of submodular maximization [11]. Because the number of possible subsets is exponential in the number of available samples, these methods rely on properties related to submodular functions so to build a greedy algorithm. It has the advantage to approximate the optimal solution with a known fixed error. In [15], Wei et al have designed several submodular functions to answer at best the need of specific classifiers (Naive Bayes Classifier, Logistic Regression Classifier, Nearest Neighbor Classifier). They have mainly focused on uncertainty selection. However the main drawback of such a method is that it is not scalable to handle the information from non-shallow classifiers. Indeed, when it comes to the application on deep networks, we notice that taking the decision is based only on the fine tuning of the last layer, while not accounting for the information hold all along the path through the intermediate layers of the deep network. This could lead to two different samples, representing thus different parts of the input manifold, resulting into the same output while having been encoded by different part of the network.

In [17], Zhang & Oles proposed a batch mode extension of A-optimality. They studied active learning by looking for the best resampling of the input unlabelled data. They consider as the optimal resampling the one which minimizes the negative expected log likehood on a fixed parameter θ. It led them to formulate a criterion on the asymptotic expected log likelihood of the resampling Fisher matrix I_q, given q the resampled distribution on p the original distribution of the input data:

$$\mathbb{E}^n(\theta) = -\frac{1}{2n} Tr(I_q(\theta)^{-1} I_p(\theta))$$

where $\mathbb{E}^n$ is the expectation over n samples from q.

As far as we restrict the context to log likelihood, the Cramer Rao bound implies that the MLE parameter Θ which minimizes $\mathbb{E}^n(\theta)$ is the asymptotic most efficient estimator of the optimal parameter among all estimators based on a resampling of the input distribution. To apply this result, they proposed to use a good empirical estimate $\hat{\Theta}$ of Θ and then replace the criterion Θ by its approximation in order to estimate the optimal resampled distribution q^*. They estimate $\hat{\Theta}$ by the trained parameters of their algorithm on the current labelled samples:

$$q^* = argmin_q Tr(I_q(\hat{\Theta})^{-1} I_p(\hat{\Theta}))$$

More samples can then be drawn so to re-estimate $\hat{\Theta}$ as well as the optimal distribution q^*. However, looking for a subset sampled from the optimal distribution q^* is not feasible as it is exponential in the number of unlabelled samples. Following this path, Hoi ([7]) used this previous criterion and approximated the solution by the maximum on a submodular function for logistic function (linear and non linear). Our contribution consists in applying Zhang & Oles criterion but by replacing the Fisher information by the KFAC approximation. Hence we obtain a direct greedy optimal search over this approximation. We dedicate section 3 to the description of our method and experiments it in section 5. Eventually we discuss how to extend our method on other OED criterions in section 4.

3 Batch active learning based on the KFAC Fisher

Let $\mathbf{p}$ and $\mathbf{q}$ be respectively the distribution of all unlabelled examples $\mathcal{U}$ and the distribution of unannotated samples selected to be labelled. Let Θ denote the parameters of the classification model. Let $I_{\mathbf{p}}(\Theta)$ and $I_{\mathbf{q}}(\Theta)$ denote respectively the Fisher information matrix of the classification model for the distribution $\mathbf{p}$ and $\mathbf{q}$. It has been proven by Zhang & Oles in [17] that the optimal subset so to reduce at best the uncertainty of classication is the one which minimizes the ratio between the two Fisher matrices, i.e.:

$$q^* = argmin_{(q|\ |q|=K)} Tr(I_q(\Theta)^{-1} I_p(\Theta)) \tag{1}$$

Two main drawbacks emerge from such an expression :

1. it requires to compute the inverse of the Fisher matrix for every potential subset **q** to be queried.

2. because of the non linarity of the inverse operator, we cannot deduce any recursive selection of the subset.

In their work [7], Hoi et al simplifies the Fisher matrix for the logistic function and upbound the criterion by the maximum of a submodular function. However, we cannot deduce any similar trick regarding deep networks. Our choice is to focus on a suboptimal solution by looking for the maximum of the trace of the inverse product. Our criterion, which we name KFAC_OED thus becomes:

$$q^{**} = argmax_{(q|\ |q|=K)} Tr(I_p(\Theta)^{-1} I_q(\Theta)) \tag{2}$$

with K the size of a batch.

When it comes to deep networks, we approximate the Fisher matrix on both distribution **p** and **q** by evaluating them on a subset of elements sampled from those distributions:

$$\begin{aligned} I_p(\Theta) &= I_{\mathcal{U}}(\Theta) \\ I_q(\Theta) &= \frac{1}{\mid q \mid} \sum_{x \in q} I_{\{x\}}(\Theta) \end{aligned} \tag{3}$$

The linearity of the trace helps into bringing a greedy selection to build the best subset:

$$q^{**} = argmax_{(q|\ |q|=K)} \frac{1}{K} \sum_{x \in q} Tr(I_{\mathcal{U}}(\Theta)^{-1} I_{\{x\}}(\Theta)) \tag{4}$$

Recently an approximation of the Fisher information for deep architectures has been proposed first for fully connected layer in [10], and then for convolutional layer as well in [6]. The block kronecker decomposition content is explained in [10, 6]

Based on their decomposition, we define the evaluation of blocks of the Fisher information at a certain point $x_i(\psi_{x_i,l}, \tau_{x_i,l})$ and an empirical estimation of the Fisher matrix on a set of data $\mathcal{D}$. A sum up of their decomposition is presented in equation (6) while the exact content of the kronecker blocks ψ and τ is only partially described in equation (5) for the sake of conciseness.

Given a fully connected layer l, s_l the input, a_l the activation and W_l the weights (and biases), the following relations holds:

$$\begin{aligned} \psi_l &= \mathbb{E}\left(a_{l-1} a_{l-1}^T\right) \\ \tau_l &= \mathbb{E}\left(\mathcal{D}s_l \mathcal{D}s_l^T\right) \end{aligned} \tag{5}$$

$$\begin{aligned} \mathbb{I}_{\mathcal{D}}(\theta) &= diag([\psi_{\mathcal{D},l}(\theta) \otimes \tau_{\mathcal{D},l}(\theta)]_{l=1}^{L}) \\ \psi_{\mathcal{D},l}(\theta) &= \frac{1}{\mid \mathcal{D} \mid} \sum_{x_i \in \mathcal{D}} \psi_{x_i,l}(\theta) \\ \tau_{\mathcal{D},l}(\theta) &= \frac{1}{\mid \mathcal{D} \mid} \sum_{x_i \in \mathcal{D}} \tau_{x_i,l}(\theta) \end{aligned} \tag{6}$$

The strength of this decomposition lies in the properties of block diagonal combined with those of the kronecker product. ψ and τ are respectively related to the covariance matrix of the activation and the covariance of the derivative given the input of a layer. Recent deep architectures tends to prevail the depth over the width (*the number of input and output neurons*) so this expression becomes really suitable and tractable. Using those relations, we obtain a fast and iterative evaluation of our criterion. Indeed we restrict the product matrix and inverse on diagonal blocks only, as described in equation (7):

$$\begin{aligned} q^{**} &= argmax_{(q|\ |q|=K)} C(q, \mathcal{U}) \\ \text{with} \quad C(q, \mathcal{U}) &= \frac{1}{K} \sum_{x \in q} \sum_{l} Tr(\psi_{(\mathcal{U},l)}^{-1} \psi_{(x,l)}) Tr(\tau_{(\mathcal{U},l)}^{-1} \tau_{x,l)}) \end{aligned} \tag{7}$$

Finally we estimate our subset $\mathcal{Q}$ by a greedy procedure: (i) to be more robust to outliers, we select first a subset $\mathcal{S} \subset \mathcal{U}$ which will be used as the set of possible queries; (ii) we recursively build $\mathcal{Q} \subset \mathcal{S}$ by picking the next sample $x_i \in \mathcal{S}$ which minimizes $C(\mathcal{Q} \cup \{x_i\}; \mathcal{U})$ among all remaining samples in $\mathcal{S} \setminus \mathcal{Q}$.
Pseudo code and illustration of the algorithm are provided in table (1).

Algorithm 1: Greedy selection of the final query $\mathcal{Q}$

Require : $\mathcal{U}$ set of initial unlabelled training examples
Require : $\mathcal{S}$ set of possibles queries
Require : N number of parameters
Require : L number of layers
Require : K number of samples to query

$i = 0;\ \mathcal{Q}^0 = \{\};\ \mathcal{S}^0 = \mathcal{S};\ \mathcal{C}^0 = 0$
# Compute the inverse Fisher information on the whole unlabelled data set
for *l in* $[|\ 1, L\ |]$ **do**
  $A_l = \psi_l(\mathcal{U}); B_l = \tau_l(\mathcal{U})$
  $A_l = A_l^{-1}; B_l = B_l^{-1}$ [a]
end
# coefficient for the greedy selection
for *x in* $\mathcal{S}$ **do**
  for *l in* $[|\ 1, L\ |]$ **do**
    $\mathcal{D}_{l,0}(x) = Tr(A_l \psi_l(\{x\}))$
    $\mathcal{D}_{l,1}(x) = Tr(B_l \tau_l(\{x\}))$
  end
end
# selection of $\mathcal{Q}$
for *k in* $[|\ 1, K\ |]$ **do**
  $x^* = argmax_{x \in \mathcal{S}^{k-1}}\ \mathcal{C}^{k-1} + \sum_l \mathcal{D}_{l,0}(x)\mathcal{D}_{l,1}(x)$
  $\mathcal{S}^k \leftarrow \mathcal{S}^{k-1} \setminus \{x^*\}$
  $\mathcal{Q}^k \leftarrow \mathcal{Q}^{k-1} \cup \{x^*\}$
  $\mathcal{C}^k \leftarrow \mathcal{C}^{k-1} + \sum_l \mathcal{D}_{l,0}(x^*)\mathcal{D}_{l,1}(x^*)$
end

[a] If blocks are too big, an approximation of the inverse by a Woodburry-Nystrom method is processed [16]

Table 1: Pseudo code of KFAC_OED

3.1 Computing the Fisher matrix on unlabelled data

The Fisher matrix implies to compute the score and the gradient on unlabelled samples. However, we cannot compute the score and the gradient without a prior knowledge on the groundtruth. In our case, we approximate the label by the prediction of the current network.

4 Extending the use of KFAC to other OED techniques: pros and cons

The greedy selection requires the operator applied on the Fisher matrix to be linear, or upbounded by a linear criterion. D-optimality is a popular framework owing to its theoretical explanation: D-optimality, by minimizing the determinant, is diminishing the volume of the version space. However, no inequalities hold between the determinant of the sum of two matrices with the sum of their determinant for non symmetric matrices. The kronecker product may help to reduce the computation of the determinant on smaller matrices, but it needs to be computed for every possible subset. Hence using our methodology, D-optimality is not scalable. An anaysis of the time complexity of D-optimality is proposed in the experiments.

E-optimality, another OED framework, is less present in the literature, except in [5] where it is used for robust design. E-optimality consists in minimizing the maximum of the eigenvalues of the observed Fisher information matrix on the current set of labeled data $\mathcal{L}$:

$$q^* = argmin_{(q,|q|=K)} \lambda_{max}(I_{\mathcal{L} \cup q}(\Theta))$$

It appears that the kronecker product properties and block diagonal configuration are highly convenient to upperbound the maximum eigenvalue:

1. the eigenvalues of a block diagonal matrix are the eigenvalues of the diagonal blocks

2. the eigenvalues of a kronecker product are all the possible products between the eigenvalues of the two matrices involved in the kronecker product

Because we consider the Fisher matrices as positive definite, we obtain the upperbound:

$$\lambda_{max}(I_{\mathcal{L}\cup q}(\Theta)) \leq \sum_{l} \lambda_{max}(\psi_{\mathcal{L}\cup q,l}(\Theta))\lambda_{max}(\tau_{\mathcal{L}\cup q,l}(\Theta)) \tag{8}$$

Since the eigenvalues are stricly positive for the Fisher matrix considered, the maximum of the eigenvalue may be upbounded by the trace. Hence a possible criterion for a greedy selection regarding E-optimality could be :

$$q^* = argmin_q \sum_{l} Tr(\psi_{\mathcal{L}\cup q,l}(\Theta))Tr(\tau_{\mathcal{L}\cup q,l}(\Theta)) \tag{9}$$

This assumption is left as an open perspective and has not been tested.

5 Experiments

5.1 Test error

We demonstrate the efficiency of our method on two benchmards datasets: MNIST (50000 greyscale images of handwritten digits) and SVHN with the extra set (over 600,000 pictures of house numbers cropped with ZCA whitening transform and global contrast normalization, classification on the central digit). We train a CNN whose architecture is detailled in table 2. We compare our approach to

hyper parameters	**MNIST**	**SVHN**
# filters	[20, 20]	[16, 16]
filter size	[(3,3), (3,3)]	[(5,5), (7,7)]
pooling size (no stride)	[(2,2), (2,2)]	[(2,2), (2,2)]
activation	Rectifier	Rectifier
# neurons in full layers	[200, 200, 50, 10]	[200, 10]

Table 2: Set of hyperparameters used respectively on the CNN for MNIST and SVHN

random selection and uncertainty selection with top score. For uncertainty selection with top score, the score of a sample is measured as the probability of its most probable label. The comparison is based on the accuracy reached for different query size.

We present the test error corresponding to the best validation score and compare the tests error on MNIST and SVHN. Preliminary results are presented in figure 1

Our first observation is that both uncertainty selection and our KFAC_OED performs way better than random selection, for different size of queries.

In figure 1 we observe that if KFAC_OED achieves equivalent accuracy that uncertainty selection on MNIST, its performances slightly decreased on SVHN. Our assumption is that since our criterion is suboptimal it prevents against a wiser active selection scheme.

When it comes to the robustness on increasing query size, as expected, KFAC_OED is a way more stable behaviour. Indeed increasing the query size from 2 minibatches to 30 leads to similar test accuracy on the first runs (*see Figure 2*). An adaptative size of query may be a good compromise between accuracy and speed. However this hypothesis is left as an open perspective.

In terms of time complexity D-optimality has two main drawbacks compared to our KFAC_OED method:

1. The determinant is a cubic time operation given the size of the input matrix, while the trace is linear in time.

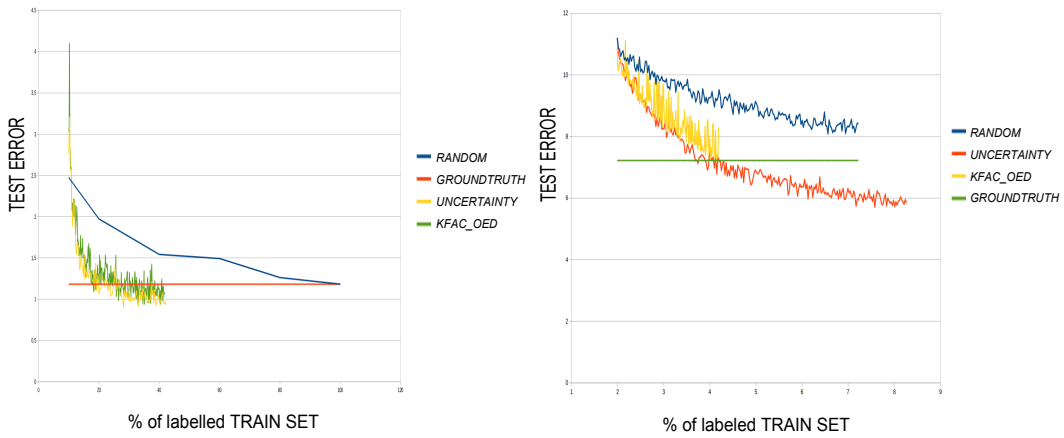

Figure 1: Test error for different percentage of labeled training set on MNIST (left) and SVHN (right)

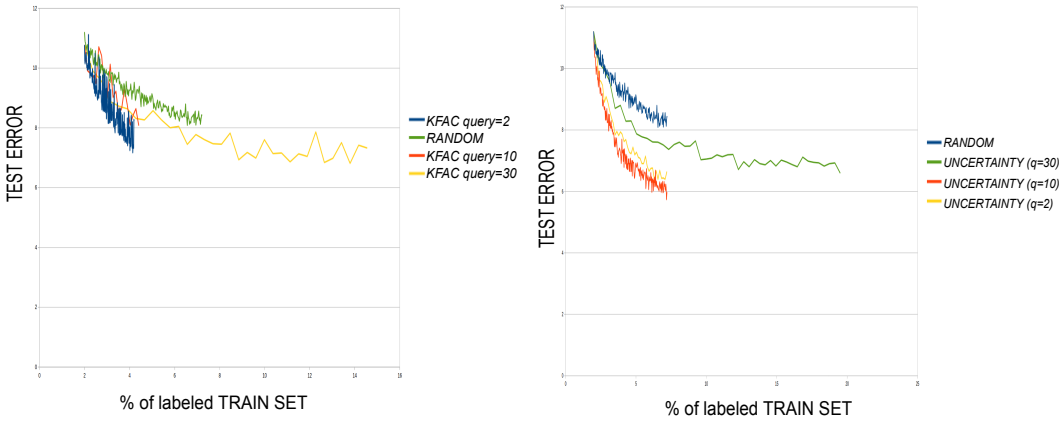

Figure 2: Test error for different query size (query 2, 10 and 30 minibatch size) on SVHN for KFAC_OED (left) and uncertainty (right)

2. There is no recursive approach for D-optimality hence to build a batch of queries we have to test every possible subset of this size. This combinatorial issue is exponential in the number of possible queries $\mid \mathcal{S} \mid$.

6 Conclusion

KFAC_OED criterion is the first in the kind to scale batch mode Optimal Experimental Design to deep networks, especially Convolutional Neural Networks. It is computationnally fast, really efficient and robust to different size of query batches. On different databases, it achieves better test accuracy than random sampling, and have a stable behavior on increasing query batch size compared to uncertainty sampling.

Our works demonstrated the validity of batch mode active learning for deep networks and the promise of OED extensions using Fisher matrix approximations.

7 Acknowledgments

We would like to thank the developpers of the frameworks Theano, Blocks [1, 14]

References

[1] J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde-Farley, and Y. Bengio. Theano: A cpu and gpu math compiler in python. In *Proc. 9th Python in Science Conf*, pages 1–7, 2010.

[2] K. Chaloner and I. Verdinelli. Bayesian experimental design: A review. *Statistical Science*, pages 273–304, 1995.

[3] N. Chan. A-optimality for regression designs. *Journal of Mathematical Analysis and Applications*, 87(1):45–50, 1982.

[4] D. A. Cohn. Neural network exploration using optimal experiment design. 1994.

[5] P. Flaherty, A. Arkin, and M. I. Jordan. Robust design of biological experiments. In *Advances in neural information processing systems*, pages 363–370, 2005.

[6] R. Grosse and J. Martens. A kronecker-factored approximate fisher matrix for convolution layers. *arXiv preprint arXiv:1602.01407*, 2016.

[7] S. C. Hoi, R. Jin, J. Zhu, and M. R. Lyu. Batch mode active learning and its application to medical image classification. In *Proceedings of the 23rd international conference on Machine learning*, pages 417–424. ACM, 2006.

[8] A. Kagan. Another look at the cramer-rao inequality. *The American Statistician*, 55(3):211–212, 2001.

[9] D. D. Lewis and W. A. Gale. A sequential algorithm for training text classifiers. In *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 3–12. Springer-Verlag New York, Inc., 1994.

[10] J. Martens and R. Grosse. Optimizing neural networks with kronecker-factored approximate curvature. *arXiv preprint arXiv:1503.05671*, 2015.

[11] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions—i. *Mathematical Programming*, 14(1):265–294, 1978.

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

[13] H. S. Seung, M. Opper, and H. Sompolinsky. Query by committee. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 287–294. ACM, 1992.

[14] B. Van Merriënboer, D. Bahdanau, V. Dumoulin, D. Serdyuk, D. Warde-Farley, J. Chorowski, and Y. Bengio. Blocks and fuel: Frameworks for deep learning. *arXiv preprint arXiv:1506.00619*, 2015.

[15] K. Wei, R. Iyer, and J. Bilmes. Submodularity in data subset selection and active learning. In *Proceedings of the 32nd International Conference on Machine Learning, Lille, Fran*, pages 6–11, 2015.

[16] M. Woodburry. Inverting modified matrices, memorandum report 42. *Statistical Research Group, Princeton, NJ*, 1950.

[17] T. Zhang and F. Oles. The value of unlabeled data for classification problems. In *Proceedings of the Seventeenth International Conference on Machine Learning,(Langley, P., ed.)*, pages 1191–1198. Citeseer, 2000.