UPAL: Unbiased Pool Based Active Learning

Ravi Ganti, Alexander Gray
School of Computational Science & Engineering, Georgia Tech
gmravi2003@gatech.edu, agray@cc.gatech.edu

November 15, 2011

Abstract
In this paper we address the problem of pool based active learning, and provide an algorithm, called UPAL, that works by minimizing the unbiased estimator of the risk of a hypothesis in a given hypothesis space. For the space of linear classifiers and the squared loss we show that UPAL is equivalent to an exponentially weighted average forecaster. Exploiting some recent results regarding the spectra of random matrices allows us to establish consistency of UPAL when the true hypothesis is a linear hypothesis. Empirical comparison with an active learner implementation in Vowpal Wabbit, and a previously proposed pool based active learner implementation show good empirical performance and better scalability.

1 Introduction
In the problem of binary classification one has a distribution \( \mathcal{D} \) on the domain \( \mathcal{X} \times \mathcal{Y} \subseteq \mathbb{R}^d \times \{-1,+1\} \), and access to a sampling oracle, which provides us i.i.d. labeled samples \( S = \{(x_1,y_1), \ldots, (x_n,y_n)\} \). The task is to learn a classifier \( h \), which predicts well on unseen points. For certain problems the cost of obtaining labeled samples can be quite expensive. For instance consider the task of speech recognition. Labeling of speech utterances needs trained linguists, and can be a fairly tedious task. Similarly in information extraction, and in natural language processing one needs expert annotators to obtain labeled data, and gathering huge amounts of labeled data is not only tedious for the experts but also expensive. In such cases it is of interest to design learning algorithms, which need only a few labeled examples for training, and also guarantee good performance on unseen data.

Suppose we are given a labeling oracle \( \mathcal{O} \), which when queried with an unlabeled point \( x \) returns the label \( y \) of \( x \). Active learning algorithms query this oracle as few times as possible and learn a provably good hypothesis from these labeled samples. Broadly speaking active learning (AL) algorithms can be classified into three kinds, namely membership query (MQ) based algorithms, stream based algorithms and pool based algorithms. All these three kinds of AL algorithms query the oracle \( \mathcal{O} \) for the label of the point, but differ in the nature of the queries. In MQ based algorithms the active learner can query for the label of a point in the input space \( \mathcal{X} \), but this query might not necessarily be from the support of the marginal distribution \( \mathcal{D}_X \). With human annotators MQ algorithms might work poorly as was demonstrated by Lang and Baum in the case of handwritten digit recognition (1992), where the annotators were faced with the awkward situation of labeling semantically meaningless images. Stream based AL algorithms (Cohn et al. 1994; Chu et al. 2011) sample a point \( x \) from the marginal distribution \( \mathcal{D}_X \), and decide on the fly whether to query \( \mathcal{O} \) for the label of \( x \)? Stream based AL algorithms tend to be computationally efficient, and most appropriate when the underlying distribution changes with time. Pool based AL algorithms assume that one has access to a large pool \( \mathcal{P} = \{x_1, \ldots, x_n\} \) of unlabeled i.i.d. examples sampled from \( \mathcal{D}_Y \), and given budget constraints \( B \), the maximum number of points they are allowed to query, query the most informative set of points. Both pool based AL algorithms, and stream based AL algorithms overcome the problem of awkward queries, which MQ based algorithms face. However in our experiments we discovered that stream based AL algorithms tend to query more points than necessary, and have poorer learning rates when compared to pool based AL algorithms.
1.1 Contributions.

1. In this paper we propose a pool based active learning algorithm called UPAL, which given a hypothesis space $\mathcal{H}$, and a margin based loss function $\phi(\cdot)$ minimizes a provably unbiased estimator of the risk $\mathbb{E}[\phi(y_h(x))]$. While unbiased estimators of risk have been used in stream based AL algorithms, no such estimators have been introduced for pool based AL algorithms. We do this by using the idea of importance weights introduced for AL in Beygelzimer et al. (2009). Roughly speaking UPAL proceeds in rounds and in each round puts a probability distribution over the entire pool, and samples a point from the pool. It then queries for the label of the point. The probability distribution in each round is determined by the current active learner obtained by minimizing the importance weighted risk over $\mathcal{H}$. Specifically in this paper we shall be concerned with linear hypothesis spaces, i.e. $\mathcal{H} = \mathbb{R}^d$.

2. In theorem 2 (Section 2.1) we show that for the squared loss UPAL is equivalent to an exponentially weighted average (EWA) forecaster commonly used in the problem of learning with expert advice (Cesa-Bianchi and Lugosi, 2006). Precisely we show that if each hypothesis $h \in \mathcal{H}$ is considered to be an expert and the importance weighted loss on the currently labeled part of the pool is used as an estimator of the risk of $h \in \mathcal{H}$, then the hypothesis learned by UPAL is the same as an EWA forecaster. Hence UPAL can be seen as pruning the hypothesis space, in a soft manner, by placing a probability distribution that is determined by the importance weighted loss of each classifier on the currently labeled part of the pool.

3. In section 3 we prove consistency of UPAL with the squared loss, when the true underlying hypothesis is a linear hypothesis. Our proof employs some elegant results from random matrix theory regarding eigenvalues of sums of random matrices (Hsu et al., 2011a,b; Tropp, 2010). While it should be possible to improve the constants and exponent of dimensionality involved in $n_0, T_0, T_1, \delta$ used in theorem 3, our results qualitatively provide us the insight that the the label complexity with the squared loss will depend on the condition number, and the minimum eigenvalue of the covariance matrix $\Sigma$. This kind of insight, to our knowledge, has not been provided before in the literature of active learning.

4. In section 5 we provide a thorough empirical analysis of UPAL comparing it to the active learner implementation in Vowpal Wabbit (VW) (Langford et al., 2011), and a batch mode active learning algorithm, which we shall call as BMAL (Hoi et al., 2006). These experiments demonstrate the positive impact of importance weighting, and the better performance of UPAL over the VW implementation. We also empirically demonstrate the scalability of UPAL over BMAL on the MNIST dataset. When we are required to query a large number of points UPAL is upto 7 times faster than BMAL.

2 Algorithm Design

A good active learning algorithm needs to take into account the fact that the points it has queried might not reflect the true underlying marginal distribution. This problem is similar to the problem of dataset shift (Quinonero et al., 2008) where the train and test distributions are potentially different, and the learner needs to take into account this bias during the learning process. One approach to this problem is to use importance weights, where during the training process instead of weighing all the points equally the algorithm weighs the points differently. UPAL proceeds in rounds, where in each round $t$, we put a probability distribution $\{p_t^i\}_{i=1}^n$ on the entire pool $\mathcal{P}$, and sample one point from this distribution. If the sampled point was queried in one of the previous rounds $1, \ldots, t-1$ then its queried label from the previous round is reused, else the oracle $\mathcal{O}$ is queried for the label of the point. Denote by $Q_t^i \in \{0, 1\}$ a random variable that takes the value 1 if the point $x_i$ was queried for it’s label in round $t$ and 0 otherwise. In order to guarantee that our estimate of the error rate of a hypothesis $h \in \mathcal{H}$ is unbiased we use importance weighting, where a point $x_i \in \mathcal{P}$ in round $t$ gets an importance weight of $Q_t^i p_t^i$. Notice that by definition $\mathbb{E}[Q_t^i | p_t^i] = 1$. We formally prove that importance weighted risk is an unbiased estimator of the true risk. Let $D_n$ denote a product distribution
on \((x_1, y_1), \ldots, (x_n, y_n)\). Also denote by \(Q_{1:n}^t\) the collection of random variables \(Q_1^t, \ldots, Q_n^t, Q_{n+1}^t, \ldots, Q_n^t\). Let \(\langle \cdot, \cdot \rangle\) denote the inner product. We have the following result.

**Theorem 1.** Let \(\overline{L}_t(h) \overset{def}{=} \frac{1}{m} \sum_{i=1}^n \sum_{\tau=1}^t \frac{Q_{\tau}^t}{p_{i}^\tau} \phi(y_i(h, x_i))\), where \(p_{i}^\tau > 0\) for all \(\tau = 1, \ldots, t\). Then

\[
E_{Q_1^t, \ldots, Q_n^n, D_n} \overline{L}_t(h) = L(h).
\]

**Proof.**

\[
E_{Q_1^t, \ldots, Q_n^t, D_n} \overline{L}_t(h) = E_{Q_1^t, \ldots, Q_n^t, D_n} \frac{1}{mt} \sum_{i=1}^n \sum_{\tau=1}^t \frac{Q_{\tau}^t}{p_{i}^\tau} \phi(y_i(h, x_i)) = E_{Q_1^t, \ldots, Q_n^t, D_n} \frac{1}{mt} \sum_{i=1}^n \sum_{\tau=1}^t E_{Q_{\tau}^t|Q_{1:n}^{\tau-1}, D_n} \frac{Q_{\tau}^t}{p_{i}^\tau} \phi(y_i(h, x_i)) = \frac{1}{m} \sum_{i=1}^n \sum_{\tau=1}^t \phi(y_i(h, x_i)) = L(w). \quad \square
\]

The theorem guarantees that as long as the probability of querying any point in the pool in any round is non-zero \(\overline{L}_t(h)\), will be an unbiased estimator of \(L(h)\). How does one come up with a probability distribution on \(P\) in round \(t\)? To solve this problem we resort to probabilistic uncertainty sampling, where the point whose label is most uncertain as per the current hypothesis, \(h_{A,t-1}\), gets a higher probability mass. The current hypothesis is simply the minimizer of the importance weighted risk in \(H\), i.e. \(h_{A,t-1} = \arg\min_{h \in H} \overline{L}_{t-1}(h)\). For any point \(x_i \in P\), to calculate the uncertainty of the label \(y_i\) of \(x_i\), we first estimate \(\eta(x_i) \overset{def}{=} \mathbb{P}[y_i = 1|x_i]\) using \(h_{A,t-1}\), and then use the entropy of the label distribution of \(x_i\) to calculate the probability of querying \(x_i\). The estimate of \(\eta(\cdot)\) in round \(t\) depends both on the current active learner \(h_{A,t-1}\), and the loss function. In general it is not possible to estimate \(\eta(\cdot)\) with arbitrary convex loss functions. However it has been shown by Zhang [2004] that the squared, logistic and exponential losses tend to estimate the underlying conditional distribution \(\eta(\cdot)\). Steps 4, 11 of algorithm \(\square\) depend on the loss function \(\phi(\cdot)\) being used. If we use the logistic loss i.e \(\phi(yz) = \ln(1 + \exp(yz))\) then \(\eta_t(x) = \frac{1}{1 + \exp(-yhx)}\). In case of squared loss \(\eta_t(x) = \min\{\max\{0, w_{A,t-1}^T x\}, 1\}\). Since the loss function is convex, and the constraint set \(H\) is convex, the minimization problem in step 11 of the algorithm is a convex optimization problem.

By design UPAL might query points. An alternate strategy is to not allow querying of points. However the importance weighted risk may not be an unbiased estimator of the true risk in such a case. Hence in order to retain the unbiasedness property we allow querying in UPAL.

### 2.1 The case of squared loss

It is interesting to look at the behaviour of UPAL in the case of squared loss where \(\phi(yhx) = (1 - yhx)^2\). For the rest of the paper we shall denote by \(h_A\) the hypothesis returned by UPAL at the end of \(T\) rounds. We now show that the prediction of \(h_A\) on any \(x\) is simply the exponentially weighted average of predictions of all \(h\) in \(H\).

**Theorem 2.** Let

\[
z_i \overset{def}{=} \sum_{\tau=1}^T \frac{Q_{\tau}^t}{p_{i}^\tau} \quad v_i \overset{def}{=} \sum_{i=1}^n z_i y_i x_i \quad c \overset{def}{=} \sum_{i=1}^n z_i \quad \Sigma_z \overset{def}{=} \sum_{i=1}^n z_i x_i x_i^T
\]

Define \(w \in \mathbb{R}^d\) as

\[
w = \frac{\int_{\mathbb{R}^d} \exp(-\overline{L}_T(h))h \, dh}{\int_{\mathbb{R}^d} \exp(-\overline{L}_T(h)) \, dh}.
\]

Assuming \(\Sigma_z\) is invertible we have for any \(x_0 \in \mathbb{R}^d\), \(w^T x_0 = h_A^T x_0\).
**Algorithm 1** UPAL (Input: \( \mathcal{P} = \{x_1, \ldots, x_n\} \), Loss function \( \phi(\cdot) \), Budget \( B \), Labeling Oracle \( \mathcal{O} \))

1. Set num_unique_queries=0, \( h_{A,0} = 0 \), \( t = 1 \).

\[ \text{while} \ \text{num\_unique\_queries} \leq B \ \text{do} \]

2. Set \( Q_t^i = 0 \) for all \( i = 1, \ldots, n \).

3. for \( x_1, \ldots, x_n \in \mathcal{P} \) do

4.  Set \( \hat{p}_{t, \min} = \frac{1}{n^{1/4}} \).

5.  Calculate \( \hat{\eta}_t(x_i) = \mathbb{P}[y = +1 | x_i, h_{A,t-1}]. \)

6.  Assign \( p_{t, i} = \hat{p}_{t, \min} + (1 - n\hat{p}_{t, \min}) \sum_{j=1}^n \hat{\eta}_t(x_j) \ln(1/\hat{\eta}_t(x_j)) + (1 - \hat{\eta}_t(x_j)) \ln(1/(1 - \hat{\eta}_t(x_j))) \).

7.  Sample a point (say \( x_j \)) from \( p_{t} \).

8.  if \( x_j \) was queried previously then

9.    Reuse its previously queried label \( y_j \).

10. else

11.    Query oracle \( \mathcal{O} \) for its label \( y_j \).

12.    num_unique_queries ← num_unique_queries + 1.

13. end if

14. end for

15. Set \( Q_j^t = 1 \).

16. Solve the optimization problem: \( h_{A,t} = \arg\min_{h \in \mathcal{H}} \sum_{i=1}^n \sum_{\tau=1}^t Q_{\tau}^i \phi(y_i h^T x_i). \)

17. \( t ← t + 1 \).

18. end while

19. Return \( h_{A} \overset{\text{def}}{=} h_{A,t} \)

**Proof.** By elementary linear algebra one can establish that

\[ h_A = \Sigma_z^{-1} v_z \]  
(3)

\[ \hat{L}_T(h) = (h - \Sigma_z^{-1} v_z) \Sigma_z (h - \Sigma_z^{-1} v_z - z). \]  
(4)

Using standard integrals we get

\[ Z \overset{\text{def}}{=} \int_{\mathbb{R}^d} \exp(-\hat{L}_T(h)) \, dh = \exp(-c - v_T^T \Sigma_z^{-1} v_z) \sqrt{\pi^d} \sqrt{\det(\Sigma_z^{-1})}. \]  
(5)

In order to calculate \( w^T x_0 \), it is now enough to calculate the integral

\[ I \overset{\text{def}}{=} \int_{\mathbb{R}^d} \exp(-\hat{L}_T(h)) \, h^T x_0 \, dw. \]

To solve this integral we proceed as follows. Define \( I_1 = \int_{\mathbb{R}^d} \exp(-\hat{L}_T(h)) \, h^T x_0 \, dh \). By simple algebra we get

\[ I = \int_{\mathbb{R}^d} \exp(-w^T \Sigma_z w + 2w^T v_z - c) \, w^T x_0 \, dw \]  
(6)

\[ = \exp(-c - v_T^T \Sigma_z^{-1} v_z) I_1. \]  
(7)
Let \( a = h - \hat{\Sigma}_z^{-1}v_z \). We then get
\[
I_1 = \int_{\mathbb{R}^d} h^T x_0 \exp \left( -(h - \hat{\Sigma}_z^{-1}v_z) \hat{\Sigma}_z (h - \hat{\Sigma}_z^{-1}v_z) \right) \, dh
\]
\[
= \int_{\mathbb{R}^d} (a^T x_0 + v_z^T \hat{\Sigma}_z^{-1} x_0) \exp(-a^T \hat{\Sigma}_z a) \, da
\]
\[
= \int_{\mathbb{R}^d} (a^T x_0) \exp(-a^T \hat{\Sigma}_z a) \, da + \int_{\mathbb{R}^d} v_z^T \hat{\Sigma}_z^{-1} x_0 \exp(-a^T \hat{\Sigma}_z a) \, da.
\]
Clearly \( I_2 \) being the integrand of an odd function over the entire space calculates to 0. To calculate \( I_3 \) we shall substitute \( \hat{\Sigma}_z = SS^T \), where \( S \succ 0 \). Such a decomposition is possible since \( \hat{\Sigma}_z \succ 0 \). Now define \( z = S^T a \). We get
\[
I_3 = v_z^T \hat{\Sigma}_z^{-1} x_0 \int \exp(-z^T z) \det(S^{-1}) \, dz
\]
\[
= v_z^T \hat{\Sigma}_z^{-1} x_0 \det(S^{-1}) / \pi^d.
\]
Using equations (7), (8) we get
\[
I = (\sqrt{\pi})^d v_z^T \hat{\Sigma}_z^{-1} x_0 \det(S^{-1}) \exp(-c - v_z^T \hat{\Sigma}_z^{-1} v_z).
\]
Hence we get
\[
w^T x_0 = v_z^T \hat{\Sigma}_z^{-1} x_0 \frac{\det(S^{-1})}{\sqrt{\det(M^{-1})}} = v_z^T \hat{\Sigma}_z^{-1} x_0 = h_A^T x_0,
\]
where the penultimate equality follows from the fact that \( \det(\hat{\Sigma}_z^{-1}) = 1/\det(\hat{\Sigma}_z) = 1/(\det(SS^T)) = 1/(\det(S))^2 \), and the last equality follows from equation 3.

Theorem 2 is instructive. It tells us that assuming that the matrix \( \hat{\Sigma}_z \) is invertible, \( h_A \) is the same as an exponentially weighted average of all the hypothesis in \( \mathcal{H} \). Hence one can view UPAL as learning with expert advice, in the stochastic setting, where each individual hypothesis \( h \in \mathcal{H} \) is an expert, and the exponential of \( \hat{L}_T \) is used to weigh the hypothesis in \( \mathcal{H} \). Such forecasters have been commonly used in learning with expert advice. This also allows us to interpret UPAL as pruning the hypothesis space in a soft way via exponential weighting, where the hypothesis that has suffered more cumulative loss gets lesser weight.

### 3 Bounding the excess risk

It is natural to ask if UPAL is consistent? That is will UPAL do as well as the optimal hypothesis in \( \mathcal{H} \) as \( n \to \infty, T \to \infty \)? We answer this question in affirmative. We shall analyze the excess risk of the hypothesis returned by our active learner, denoted as \( h_A \), after \( T \) rounds when the loss function is the squared loss. The prime motivation for using squared loss over other loss functions is that squared losses yield closed form estimators, which can then be elegantly analyzed using results from random matrix theory \cite{Hsu,Tropp,vandeGeer}.

#### 3.1 Main result

**Theorem 3.** Let \((x_1, y_1), \ldots, (x_n, y_n)\) be sampled i.i.d from a distribution. Suppose assumptions A0-A3 hold. Let \( \delta \in (0, 1) \), and suppose \( n \geq n_{0, \delta}, T \geq \max\{T_{0, \delta}, T_{1, \delta}\} \). With probability at least \( 1 - 10\delta \) the excess risk of the active learner returned by UPAL after \( T \) rounds is
\[
L(h_A) - L(\beta) = O \left( \frac{1}{n} + \frac{n}{\sqrt{T}}(d + 2\sqrt{d \ln(1/\delta)} + 2 \ln(1/\delta)) \right).
\]
3.2 Assumptions, and Notation.

**A0** (Invertibility of Σ) The data covariance matrix Σ is invertible.

**A1** (Statistical leverage condition) There exists a finite $\gamma_0 \geq 1$ such that almost surely

$$||\Sigma^{-1/2}x|| \leq \gamma_0 \sqrt{d}.$$  

**A2** There exists a finite $\gamma_1 \geq 1$ such that $E[\exp(\alpha^T x)] \leq \exp\left(\frac{||\alpha||^2\gamma_1^2}{2}\right)$.

**A3** (Linear hypothesis) We shall assume that $y = \beta^T x + \xi(x)$, where $\xi(x) \in [-2, 2]$ is additive noise with $E[\xi(x)|x] = 0$.

Assumption A0 is necessary for the problem to be well defined. A1 has been used in recent literature to analyze linear regression under random design and is a Bernstein like condition [Rokhlin and Tygert 2008]. A2 can be seen as a softer form of boundedness condition on the support of the distribution. In particular if the data is bounded in a d-dimensional unit cube then it suffices to take $\gamma_1 = 1/2$. It may be possible to satisfy A3 by mapping data to kernel spaces. Though popularly used kernels such as Gaussian kernel map the data to infinite dimensional spaces, a finite dimensional approximation of such kernel mappings can be found by the use of random features [Rahimi and Recht 2007].

**Notation.**

1. $h_A$ is the active learner outputted by our active learning algorithm at the end of $T$ rounds.

2. 

$$\forall i = 1, \ldots, n : z_i \overset{\text{def}}{=} \sum_{t=1}^T Q_t^i p_t^i, \quad \hat{\Sigma}_z = \sum_{i=1}^n z_i x_i x_i^T$$

$$\psi_z = \sum_{i=1}^n z_i \xi(x_i) x_i$$

$$\Sigma = E[xx^T], \quad \hat{\Sigma} = \sum_{i=1}^n z_i x_i x_i^T$$

$$n_{0,\delta} \overset{\text{def}}{=} 7200d^2 \gamma_0^4 (d \ln(5) + \ln(10/\delta))$$

$$T_{0,\delta} \overset{\text{def}}{=} 12 + 512\sqrt{d}^8/3 \gamma_0^{16/3} \ln^{4/3}(d/\delta)$$

$$T_{0,\delta} \overset{\text{def}}{=} \gamma_1^{16/3} d^{8/3} \ln^{4/3}(d/\delta) \ln^{8/3}(n/\delta) \lambda_{\min}^{8/3}(\Sigma) + 4 \ln(d/\delta) \frac{\lambda_{\max}(\Sigma)}{\lambda_{\min}(\Sigma)},$$

where $\delta \in (0,1)$.

3.3 Overview of the proof

The excess risk of a hypothesis $h \in H$ is defined as $L(h) - L(\beta) = E_{x,y \sim D}(y - h^T x)^2 - (y - \beta^T x)^2$. Our aim is to provide high probability bounds for the excess risk, where the probability measure is w.r.t the sampled points $(x_1, y_1), \ldots, (x_n, y_n), Q_1^T, \ldots, Q_n^T$. The proof proceeds as follows.

1. In lemma 1 assuming that the matrices $\hat{\Sigma}_z, \hat{\Sigma}$ are invertible we upper bound the excess risk as the product $||\Sigma^{-1/2}\hat{\Sigma}_z^{-1}\Sigma^{-1/2}||^2 ||\Sigma^{-1/2}\hat{\Sigma}^{-1/2}||^2 ||\hat{\Sigma}_z^{-1/2}\psi_z||^2$. The prime motivation in doing so is that bounding such “squared norm” terms can be reduced to bounding the maximum eigenvalue of random matrices, which is a well studied problem in random matrix theory.
2. In lemma \([5]\) we provide an upper bound for \(\|\Sigma^{-1/2}\hat{\Sigma}^{1/2}\|^2\). To do this we use the simple fact that the matrix 2-norm of a positive semidefinite matrix is nothing but the maximum eigenvalue of the matrix. With this observation, and by exploiting the structure of the matrix \(\hat{\Sigma}\), the problem reduces to giving probabilistic upper bounds for maximum eigenvalue of a sum of random rank-1 matrices. Theorem \([5]\) provides us with a tool to prove such bounds.

3. In lemma \([6]\) we bound \(\|\Sigma^{-1/2}\hat{\Sigma}_z^{-1}\Sigma^{1/2}\|^2\). The proof is in the same spirit as in lemma \([5]\) however the resulting probability problem is that of bounding the maximum eigenvalue of a sum of random matrices, which are not necessarily rank-1. Theorem \([6]\) provides us with Bernstein type bounds to analyze the eigenvalues of sums of random matrices.

4. In lemma \([7]\) we bound the quantity \(\|\Sigma^{-1/2}\psi_z\|^2\). Notice that here we are bounding the squared norm of a random vector. Theorem \([8]\) provides us with a tool to analyze such quadratic forms under the assumption that the random vector has sub-Gaussian exponential moments behaviour.

5. Finally all the above steps were conditioned on the invertibility of the random matrices \(\hat{\Sigma}, \hat{\Sigma}_z\). We provide conditions on \(n, T\) (this explains why we defined the quantities \(n_{0,\delta}, T_{0,\delta}, T_{1,\delta}\) which guarantee the invertibility of \(\hat{\Sigma}, \hat{\Sigma}_z\). Such problems boil down to calculating lower bounds on the minimum eigenvalue of the random matrices in question, and to establish such lower bounds we once again use theorems \([5]\) \([6]\).

### 3.4 Full Proof

We shall now provide a way to bound the excess risk of our active learner hypothesis. Suppose \(h_A\) was the hypothesis represented by the active learner at the end of the \(T\) rounds. By the definition of our active learner and the definition of \(\beta\) we get

\[
h_A = \arg \min_{h \in \mathcal{H}} \sum_{i=1}^{n} \sum_{t=1}^{T} Q^t_i \left( y_i - h^T x_i \right)^2 = \sum_{i=1}^{n} z_i (y_i - h^T x_i)^2 = \hat{\Sigma}_z^{-1} \psi_z \tag{11}
\]

\[
\beta = \arg \min_{h \in \mathcal{H}} \mathbb{E}(y - \beta^T x)^2 = \Sigma^{-1} \mathbb{E}[yx]. \tag{12}
\]

**Lemma 1.** Assume \(\Sigma_z, \hat{\Sigma}\) are both invertible, and assumption \(A0\) applies. Then the excess risk of the classifier after \(T\) rounds of our active learning algorithm is given by

\[
L(h_A) - L(\beta) \leq \|\Sigma^{1/2}\hat{\Sigma}_z^{-1}\Sigma^{1/2}\|^2 \|\Sigma^{-1/2}\hat{\Sigma}^{1/2}\|^2 \|\Sigma^{-1/2}\psi_z\|^2. \tag{13}
\]

**Proof.**

\[
L(h_A) - L(\beta) = \mathbb{E}[(y - h_A^T x)^2 - (y - \beta^T x)^2]
\]

\[
= \mathbb{E}_{x, y} [h_A^T xx^T h_A - 2 y h_A^T x - \beta^T xx^T \beta + 2 y \beta^T x]
\]

\[
= h_A^T \Sigma h_A - 2 h_A^T \mathbb{E}[xy] - \beta^T \Sigma \beta + 2 \beta^T \Sigma \beta \quad [\text{Since } \Sigma \beta = \mathbb{E}[yx]]
\]

\[
= h_A^T \Sigma h_A - \beta^T \Sigma \beta - 2 h_A^T \Sigma \beta + 2 \beta^T \Sigma \beta
\]

\[
= h_A^T \Sigma h_A + \beta^T \Sigma \beta - 2 h_A^T \Sigma \beta
\]

\[
= \|\Sigma^{1/2}(h_A - \beta)\|^2. \tag{14}
\]

We shall next bound the quantity \(\|h_A - \beta\|\) which will be used to bound the excess risk in Equation \((14)\). To do this we shall use assumption \(A3\) along with the definitions of \(h_A, \beta\). We have the following chain of
inequalities.

\[ h_A = \hat{\Sigma}_z^{-1}v_z \]
\[ = \hat{\Sigma}_z^{-1}\sum_{i=1}^{n} z_i y_i x_i \]
\[ = \hat{\Sigma}_z^{-1}\sum_{i=1}^{n} z_i (\beta^T x_i + \xi(x_i))x_i \]
\[ = \hat{\Sigma}_z^{-1}\sum_{i=1}^{n} z_i x_i x_i^T \beta + z_i \xi(x_i)x_i \]
\[ = \beta + \hat{\Sigma}_z^{-1}\sum_{i=1}^{n} z_i \xi(x_i)x_i = \beta + \hat{\Sigma}_z^{-1}\psi_z. \]  

(15)

Using Equations 14, 15 we get the following series of inequalities for the excess risk bound

\[ L(h_A) - L(\beta) = ||\Sigma^{1/2}\hat{\Sigma}_z^{-1}\psi_z||^2 \]
\[ = ||\Sigma^{1/2}\hat{\Sigma}_z^{-1}\Sigma^{1/2}\hat{\Sigma}_z^{-1/2}\psi_z||^2 \]
\[ = ||\Sigma^{1/2}\hat{\Sigma}_z^{-1}\Sigma^{1/2}\Sigma^{-1/2}\Sigma^{1/2}\hat{\Sigma}_z^{-1/2}\psi_z||^2 \]
\[ \leq ||\Sigma^{1/2}\hat{\Sigma}_z^{-1}\Sigma^{1/2}||^2||\Sigma^{-1/2}\Sigma^{1/2}||^2||\Sigma^{-1/2}\psi_z||^2. \]  

(16)

The decomposition in lemma 1 assumes that both \( \hat{\Sigma}_z, \Sigma \) are invertible. Before we can establish conditions for the matrices \( \hat{\Sigma}_z, \Sigma \) to be invertible we need the following elementary result.

**Proposition 1.** For any arbitrary \( \alpha \in \mathbb{R}^d \), under assumption A1 we have

\[ \mathbb{E}[\exp(\alpha^T \Sigma^{-1/2} x)] \leq 5 \exp\left(\frac{3d\gamma_0^2||\alpha||^2}{2}\right). \]  

(17)

**Proof.** From Cauchy-Schwarz inequality and A1 we get

\[-||\alpha||\gamma_0\sqrt{d} \leq -||\alpha|| ||\Sigma^{-1/2} x|| \leq \alpha^T \Sigma^{-1/2} x \leq ||\alpha|| ||\Sigma^{-1/2} x|| \leq ||\alpha||\gamma_0\sqrt{d}. \]  

(18)

Also \( \mathbb{E}[\alpha^T \Sigma^{-1/2} x] \leq ||\alpha||\gamma_0\sqrt{d} \). Using Hoeffding’s lemma we get

\[ \mathbb{E}[\exp(\alpha^T \Sigma^{-1/2} x)] \leq \exp\left(||\alpha||\gamma_0\sqrt{d} + \frac{||\alpha||^2 d\gamma_0^2}{2}\right) \]
\[ \leq \exp(3||\alpha||^2 d\gamma_0^2/2). \]

(19)

The following lemma will be useful in bounding the terms \( ||\Sigma^{1/2}\hat{\Sigma}_z^{-1}\Sigma^{1/2}||, ||\Sigma^{-1/2}\hat{\Sigma}_z^{-1/2}||^2 \).

**Lemma 2.** Let \( J \equiv \sum_{i=1}^{n} \Sigma^{-1/2} x_i x_i^T \Sigma^{-1/2} \). Let \( n \geq n_0, \delta \). Then the following inequalities hold separately with probability at least \( 1 - \delta \) each

\[ \lambda_{\text{max}}(J) \leq n + 6dn\gamma_0^2 \left[ \sqrt{\frac{32(d\ln(5) + \ln(10/\delta))}{n}} + \frac{2(d\ln(5) + \ln(10/\delta))}{n} \right] \leq 3n/2 \]  

(20)

\[ \lambda_{\text{min}}(J) \geq n - 6dn\gamma_0^2 \left[ \sqrt{\frac{32(d\ln(5) + \ln(10/\delta))}{n}} + \frac{2(d\ln(5) + \ln(10/\delta))}{n} \right] \geq n/2. \]  

(21)
Proof. Notice that $E[\Sigma^{-1/2} x_i x_i^T \Sigma^{-1/2}] = I$. From Proposition 1 we have $E[\exp(\alpha^T \Sigma^{-1/2} x)] \leq 5 \exp(3|\alpha|^2 d \gamma_0^2/2)$. By using theorem 5 we get $E[\exp(\alpha^T \Sigma^{-1/2} x)] \leq 5 \exp(3|\alpha|^2 d \gamma_0^2/2)$. Hence we get $\hat{\Sigma}$.

Proof. From A2 we have
\[
\lambda_{\max}(\frac{1}{n} \sum_{i=1}^{n} (\Sigma^{-1/2} x_i)(\Sigma^{-1/2} x_i)^T) \leq 1 + 6d \gamma_0^2 \left[ \frac{32(d \ln(5) + \ln(2/\delta))}{n} + \frac{2(d \ln(5) + \ln(2/\delta))}{n} \right]. \tag{22}
\]

Put $n \geq n_{0, \delta}$ to get the desired result. The lower bound on $\lambda_{\min}$ is also obtained in the same way.

Lemma 3. Let $n \geq n_{0, \delta}$. With probability atleast $1 - \delta$ separately we have $\hat{\Sigma} \succ 0$, $\lambda_{\min}(\hat{\Sigma}) \geq \frac{1}{2} \lambda_{\min}(\Sigma)$, $\lambda_{\max}(\hat{\Sigma}) \leq \frac{3}{2} \lambda_{\max}(\Sigma)$.

Proof. Using lemma 2 we get for $n \geq n_{0, \delta}$ with probability atleast $1 - \delta$, $\lambda_{\min}(J) \geq 1/2$ and with probability atleast $1 - \delta$, $\lambda_{\max}(\Sigma) \leq 3/2$. Finally since $\Sigma^{1/2} J \Sigma^{1/2} = \hat{\Sigma}$, and $J \succ 0$, $\hat{\Sigma} \succ 0$, we get $\hat{\Sigma} \succ 0$. Further we have the following upper bound with probability atleast $1 - \delta$:
\[
\lambda_{\max}(\hat{\Sigma}) = ||\Sigma^{1/2} J \Sigma^{1/2}|| \leq ||\Sigma^{1/2}||^2 ||J|| \leq ||\Sigma|| ||J|| = \lambda_{\max}(\Sigma) \lambda_{\max}(J) \leq \frac{3}{2} \lambda_{\max}(\Sigma),
\]

where in the last step we used the upper bound on $\lambda_{\max}(J)$ provided by lemma 2. Similarly we have the following lower bound with probability atleast $1 - \delta$:
\[
\lambda_{\min}(\hat{\Sigma}) = \frac{1}{\lambda_{\max}(\Sigma^{-1/2} J^{-1} \Sigma^{-1/2})} = \frac{1}{||\Sigma^{-1/2} J^{-1} \Sigma^{-1/2}||} \geq \frac{1}{||\Sigma^{-1}|| ||J^{-1}|| ||\Sigma^{-1/2}||} = \lambda_{\min}(\Sigma) \lambda_{\min}(J) \geq \frac{\lambda_{\min}(\Sigma)}{2},
\]

where in the last step we used the lower bound on $\lambda_{\min}(J)$ provided by lemma 2.

The following proposition will be useful in proving lemma 4.

Proposition 2. Let $\delta \in (0, 1)$. Under assumption A2, with probability atleast $1 - \delta$, $\sum_{i=1}^{n} ||x_i||^4 \leq 25\gamma_1^4 d^2 \ln^2(n/\delta)$

Proof. From A2 we have $E[\exp(\alpha^T x)] \leq \exp(\frac{|\alpha|^2 \gamma_1^2}{2})$. Now applying theorem 4 with $A = I_d$ we get
\[
P[||x_j||^2 \leq 2\gamma_1^2 + 2\gamma_1^2 \sqrt{d \ln(1/\delta)} + 2\gamma_1^2 \ln(1/\delta)] \geq 1 - \delta. \tag{33}
\]

The result now follows by the union bound.

Lemma 4. Let $\delta \in (0, 1)$. For $T \geq T_{0, \delta}$, with probability atleast $1 - 4\delta$ we have $\lambda_{\min}(\hat{\Sigma}_z) \geq \frac{nT \lambda_{\min}(\Sigma)}{4} > 0$. Hence $\hat{\Sigma}_z$ is invertible.
Proof. The proof uses theorem 6. Let \( M'_t \defeq \sum_{i=1}^n Q^T_t x_i x_i^T \), so that \( \hat{\Sigma} = \sum_t M'_t \). Now \( E_t M'_t = n \hat{\Sigma} \). Define \( R'_t \defeq n \hat{\Sigma} - M'_t \), so that \( E_t R'_t = 0 \). We shall apply theorem 6 to the random matrix \( \sum R'_t \). In order to do so we need upper bounds on \( \lambda_{\max}(R'_t) \) and \( \lambda_{\max} \left( \frac{1}{T} \sum_{t=1}^T E_t R'_t \right) \). Let \( n \geq n_{0,\delta} \). Using lemma 3 we get with probability at least \( 1 - \delta \)

\[
\lambda_{\max}(R'_t) = \lambda_{\max}(n \hat{\Sigma} - M'_t) \leq \lambda_{\max}(n \hat{\Sigma}) \leq \frac{3n \lambda_{\max}(\Sigma)}{2} \defeq b_2.
\]

\[
\lambda_{\max} \left( \frac{1}{T} \sum_{t=1}^T E_t R'_t^2 \right) = \frac{1}{T} \lambda_{\max} \left( \sum_{t=1}^T E_t (n \hat{\Sigma} - M'_t)^2 \right)
= \frac{1}{T} \lambda_{\max} \left( -n^2 T \hat{\Sigma}^2 \right) + \frac{1}{T} \lambda_{\max} \left( \sum_{t=1}^T \sum_{i=1}^n Q^T_t (x_i x_i^T)^2 \right)
\leq \frac{1}{T} \lambda_{\max} \left( \sum_{t=1}^T \sum_{i=1}^n \frac{1}{p_i} (x_i x_i^T)^2 \right) - n^2 \lambda_{\min}(\hat{\Sigma})
\leq n T^{1/4} \lambda_{\max} \left( \sum_{i=1}^n (x_i x_i^T)^2 \right)
\leq n T^{1/4} \sum_{i=1}^n \lambda_{\max}(x_i x_i^T)
= n T^{1/4} \sum_{i=1}^n ||x_i||^4
\leq 25 \gamma^4 d^2 n^2 T^{1/4} \ln^2 (n/\delta) \defeq \sigma^2_2.
\]

Equation (36) follows from Equation (35) by the definition of \( M'_t \) and the fact that at any given \( t \) only one point is queried i.e. \( Q^T_t Q_j^T = 0 \) for a given \( t \). Equation (37) follows from Equation (36) since \( E_t Q^T_t p_i = p_i \). Equation (38) follows from Equation (37) by Weyl’s inequality. Equation (39) follows from Equation (38) by substituting \( p_{\min} \) in place of \( p_i \). Equation (40) follows from Equation (39) by the use of Weyl’s inequality. Equation (41) follows from Equation (40) by using the fact that if \( p \) is a vector then \( \lambda_{\max}(p p^T) = ||p||^2 \). Equation (42) follows from Equation (41) by the use of proposition 2. Notice that this step is a stochastic inequality and holds with probability at least \( 1 - \delta \).

Finally applying theorem 6 we have

\[
P \left[ \lambda_{\max} \left( \frac{1}{T} \sum_{t=1}^T R'_t \right) \leq \sqrt{\frac{2 \sigma^2_2 \ln(d/\delta)}{T}} + \frac{b_2 \ln(d/\delta)}{T} \right] \geq 1 - \delta
\]

\[
\Rightarrow P \left[ \lambda_{\max} \left( n \hat{\Sigma} - \frac{1}{T} \sum_{t=1}^T M'_t \right) \leq \sqrt{\frac{2 \sigma^2_2 \ln(d/\delta)}{T}} + \frac{b_2 \ln(d/\delta)}{T} \right] \geq 1 - \delta
\]

\[
\Rightarrow P \left[ \lambda_{\min} \left( n \hat{\Sigma} \right) - \frac{1}{T} \lambda_{\min} \left( \sum_{t=1}^T M'_t \right) \leq \sqrt{\frac{2 \sigma^2_2 \ln(d/\delta)}{T}} + \frac{b_2 \ln(d/\delta)}{T} \right] \geq 1 - \delta
\]
Substituting for $\sigma_2, b_2$, rearranging the inequalities, and using lemma 3 to lower bound $\lambda_{\min}(\hat{\Sigma})$ we get

\[
P \left[ \lambda_{\min}\left( \sum_{t=1}^{T} M_t' \right) \geq T \lambda_{\min}(n\hat{\Sigma}) - \sqrt{2T \sigma_2^2 \ln(d/\delta) - b_2 \ln(d/\delta)} \right] \geq 1 - \delta
\]

\[
\implies P \left[ \lambda_{\min}\left( \sum_{t=1}^{T} M_t' \right) \geq \frac{nT \lambda_{\min}(\Sigma)}{2} - \sqrt{2T \sigma_2^2 \ln(d/\delta) - b_2 \ln(d/\delta)} \right] \geq 1 - 2\delta
\]

\[
\implies P \left[ \lambda_{\min}\left( \sum_{t=1}^{T} M_t' \right) \geq \frac{nT \lambda_{\min}(\Sigma)}{2} - 5\sqrt{2T \sigma_2^2 \ln(d/\delta) \ln(n/\delta)} - \frac{n \ln(d/\delta) \lambda_{\max}(\Sigma)}{2} \right] \geq 1 - 4\delta
\]

For $T \geq T_{0,\delta}$ with probability at least $1 - 4\delta$, $\lambda_{\min}\sum_{t=1}^{T} M_t' = \lambda_{\min}(\hat{\Sigma}) \geq \frac{nT \lambda_{\min}(\Sigma)}{4}$. \qed

**Lemma 5.** For $n \geq n_{0,\delta}$ with probability at least $1 - \delta$ over the random sample $x_1, \ldots, x_n$

\[
||\Sigma^{1/2}\hat{\Sigma}^{1/2}||^2 \leq 3/2.
\] (46)

**Proof.**

\[
||\Sigma^{1/2}\hat{\Sigma}^{1/2}||^2 = ||\hat{\Sigma}^{1/2}\Sigma^{1/2}||^2
\] (47)

\[
= \lambda_{\max}(\Sigma^{1/2}\hat{\Sigma}\Sigma^{1/2})
\] (48)

\[
= \lambda_{\max}\left( \frac{1}{n} \sum_{i=1}^{n} (\Sigma^{1/2}x_i)(\Sigma^{1/2}x_i)^T \right)
\] (49)

\[
= \lambda_{\max}\left( \frac{J}{n} \right)
\] (50)

\[
\leq 3/2
\] (51)

where in the first equality we used the fact that $||A|| = ||A^T||$ for a square matrix $A$, and $||A||^2 = \lambda_{\max}(A^T A)$, and in the last step we used lemma 2.

**Lemma 6.** Suppose $\hat{\Sigma}$ is invertible. Given $\delta \in (0, 1)$, for $n \geq n_{0,\delta}$, and $T \geq \max\{T_{0,\delta}, T_{1,\delta}\}$ with probability at least $1 - 3\delta$ over the samples

\[
||\Sigma^{1/2}\hat{\Sigma}^{-1}\Sigma^{1/2}||^2 \leq \frac{400}{n^2T^2}.
\]

**Proof.** The proof of this lemma is very similar to the proof of lemma 4. From lemma 4 for $n \geq n_{0,\delta}, T \geq T_{0,\delta}$ with probability at least $1 - \delta$, $\lambda_{\min}(\hat{\Sigma}) > 0$. Using the assumption that $\Sigma > 0$, we get $\Sigma^{1/2}\hat{\Sigma}^{-1}\Sigma^{1/2} > 0$. Hence $||\Sigma^{1/2}\hat{\Sigma}^{-1}\Sigma^{1/2}|| = \lambda_{\max}(\Sigma^{1/2}\hat{\Sigma}^{-1}\Sigma^{1/2}) = \frac{1}{\lambda_{\min}(\Sigma^{1/2}\hat{\Sigma}^{-1}\Sigma^{1/2})}$. Hence it is enough to provide a lower bound on the smallest eigenvalue of the symmetric positive definite matrix $\Sigma^{1/2}\hat{\Sigma}^{-1}\Sigma^{1/2}$.

\[
\lambda_{\min}(\Sigma^{1/2}\hat{\Sigma}^{-1}\Sigma^{1/2}) = \lambda_{\min}\left( \sum_{i=1}^{n} z_i \Sigma^{-1/2} x_i x_i^T \Sigma^{-1/2} \right)
\]

\[
= \lambda_{\min}\left( \sum_{t=1}^{T} \sum_{i=1}^{n} \frac{q_t^i}{p_t^i} \Sigma^{-1/2} x_i x_i^T \Sigma^{-1/2} \right) \overset{\text{def}}{=} M_t
\]

\[
= \lambda_{\min}\left( \sum_{t=1}^{T} M_t \right)
\]
Define $R_t \overset{\text{def}}{=} J - M_t$. Clearly $\mathbb{E}_t[M_t] = J$, and hence $\mathbb{E}[R_t] = 0$. From Weyl’s inequality we have $\lambda_{\text{min}}(J) + \lambda_{\text{max}}\left(\frac{1}{T} \sum_{t=1}^{T} M_t\right) \leq \lambda_{\text{max}}\left(\frac{1}{T} \sum_{t=1}^{T} R_t\right)$. Now applying theorem 6 on $\sum R_t$ we get with probability at least $1 - \delta$,

$$\lambda_{\text{min}}(J) + \lambda_{\text{max}}\left(\frac{1}{T} \sum_{t=1}^{T} M_t\right) \leq \lambda_{\text{max}}\left(\frac{1}{T} \sum_{t=1}^{T} R_t\right) \leq \sqrt{\frac{2\sigma^2 \ln(d/\delta)}{T}} + \frac{b_1 \ln(d/\delta)}{3T}, \tag{52}$$

where

$$\lambda_{\text{max}}\left(\frac{1}{T} \sum_{t=1}^{T} J - M_t\right) \leq b_1 \tag{53}$$

$$\lambda_{\text{max}}\left(\frac{1}{T} \sum_{t=1}^{T} \mathbb{E}_t(J - M_t)^2\right) \leq \sigma_1^2 \tag{54}$$

Rearranging Equation (52) and using the fact that $\lambda_{\text{max}}(-A) = -\lambda_{\text{min}}(A)$ we get with probability at least $1 - \delta$,

$$\lambda_{\text{min}}\left(\sum_{t=1}^{T} M_t\right) \geq T\lambda_{\text{min}}(J) - \sqrt{2T\sigma_1^2 \ln(d/\delta)} - \frac{b_1 \ln(d/\delta)}{3}. \tag{55}$$

Using Weyl’s inequality (Horn and Johnson 1990) we have $\lambda_{\text{max}}\left(\frac{1}{T} \sum_{t=1}^{T} J - M_t\right) \leq \lambda_{\text{max}}(J) \leq \frac{3n}{T}$ with probability at least $1 - \delta$, where in the last step we used lemma (2). Let $b_1 \overset{\text{def}}{=} \frac{3n}{T}$. To calculate $\sigma_1^2$ we proceed as follows.

$$\lambda_{\text{max}}\left(\frac{1}{T} \sum_{t=1}^{T} \mathbb{E}_t(J - M_t)^2\right) = \frac{1}{T} \lambda_{\text{max}}\left(\sum_{t=1}^{T} \mathbb{E}_t(M_t^2) - J^2\right) \tag{56}$$

$$\leq \frac{1}{T} \lambda_{\text{max}}\left(\sum_{t=1}^{T} \mathbb{E}_t M_t^2\right) \tag{57}$$

$$= \frac{1}{T} \lambda_{\text{max}}\left(\sum_{t=1}^{T} \mathbb{E}_t \left(\sum_{i=1}^{n} \frac{Q_i^t}{p_i^t} \Sigma_{-1/2} x_i x_i^T \Sigma_{-1/2}\right)^2\right) \tag{58}$$

$$= \frac{1}{T} \lambda_{\text{max}}\left(\sum_{t=1}^{T} \mathbb{E}_t \sum_{i=1}^{n} \frac{Q_i^t}{p_i^t} (\Sigma_{-1/2} x_i x_i^T \Sigma_{-1/2})^2\right) \tag{59}$$

$$= \frac{1}{T} \lambda_{\text{max}}\left(\sum_{i=1}^{n} \sum_{t=1}^{T} \frac{1}{p_i^t} (\Sigma_{-1/2} x_i x_i^T \Sigma_{-1/2})^2\right) \tag{60}$$

$$\leq \frac{1}{T} \sum_{i=1}^{n} \sum_{t=1}^{T} \frac{1}{p_i^t} ||\Sigma_{-1/2} x_i||^4 \tag{61}$$

$$\leq \frac{d^2 \gamma_0}{T} \sum_{i=1}^{n} \sum_{t=1}^{T} \frac{1}{p_i^t} \tag{62}$$

$$\leq \frac{nd^2 \gamma_0}{T} \sum_{t=1}^{T} \frac{1}{p_{\text{min}}} \tag{63}$$

$$\leq n^2 d^2 \gamma_0^4 T^{1/4} \overset{\text{def}}{=} \sigma_1^2. \tag{64}$$

Equation 57 follows from Equation 56 by using Weyl’s inequality and the fact that $J^2 \geq 0$. Equation 59 follows from Equation 58 since only one point is queried in every round and hence for any given $t, i \neq j$.
we have \( Q_i^t Q_i^t = 0 \), and hence all the cross terms disappear when we expand the square. Equation (60) follows from Equation (59) by using the fact that \( E_x Q_i = p_i \). Equation (61) follows from Equation (60) by Weyl’s inequality and the fact that the maximum eigenvalue of a rank-1 matrix of the form \( vv^T \) is \( \|v\|^2 \).

Equation (62) follows from Equation (61) by using assumption A1. Equation (63) follows from Equation (62) by our choice of \( p_{min}^t = \frac{1}{n^{\frac{1}{t}}} \). Substituting the values of \( \sigma^2, b_1 \) in (55), using Lemma 2 to lower bound \( \lambda_{min}(J) \), and applying union bound to sum up all the failure probabilities we get for \( n \geq n_{0,\delta}, T \geq \max\{T_{0,\delta}, T_{1,\delta}\} \) with probability at least \( 1 - 3\delta \),

\[
\lambda_{min} \left( \sum_{t=1}^T M_t \right) \geq T \lambda_{min}(J) - \sqrt{2T^5/4n^2d^2\gamma_0^4 \ln(d/\delta)} - 3n/2
\]

\[
\geq \frac{nT}{2} - \sqrt{2T^5/8n^2d^2 \gamma_0^2 \ln(d/\delta)} - 3n/2 \geq nT/4. \tag*{\square}
\]

The only missing piece in the proof is an upper bound for the quantity \( \|\hat{\Sigma}^{-1/2}\psi_z\|^2 \). The next lemma provides us with an upper bound for this quantity.

**Lemma 7.** Suppose \( \hat{\Sigma} \) is invertible. Let \( \delta \in (0, 1). \) With probability at least \( 1 - \delta \) we have

\[
\|\hat{\Sigma}^{-1/2}\psi_z\|^2 \leq (2nT^2 + 56n^3T\sqrt{T})(d + 2\sqrt{d \ln(1/\delta)} + 2 \ln(1/\delta)).
\]

**Proof.** Define the matrix \( A \in \mathbb{R}^{d \times n} \) as follows. Let the \( i^{th} \) column of \( A \) be the vector \( \frac{\hat{\Sigma}^{-1/2} x_i \hat{\Sigma}^{-1/2}}{\sqrt{n}} \), so that \( AA^T = \frac{1}{n} \hat{\Sigma}^{-1/2} x_i x_i^T \hat{\Sigma}^{-1/2} = I_d \). Now \( \|\hat{\Sigma}^{-1/2} \psi_z\|^2 = \|\sqrt{n}Ap\|^2 \), where \( p = (p_1, \ldots, p_n) \in \mathbb{R}^n \) and \( p_i = \xi(x_i)z_i \) for \( i = 1, \ldots, n \). Using the result for quadratic forms of subgaussian random vectors (Theorem 4) we get

\[
\|Ap\|^2 \leq \sigma^2 (\text{tr}(I_d) + 2\sqrt{\text{tr}(I_d) \ln(1/\delta)} + 2\|I_d\| \ln(1/\delta)) = \sigma^2 (d + 2\sqrt{d \ln(1/\delta)} + 2 \ln(1/\delta)), \tag{65}
\]

where for any arbitrary vector \( \alpha \), \( \mathbb{E}[\exp(\alpha^T p)] \leq \exp(\|\alpha\|^2 \sigma^2) \).

Hence all that is left to be done is prove that \( \alpha^T p \) has sub-Gaussian exponential moments. Let

\[
D_t \defeq \sum_{i=1}^n \frac{\alpha_i \xi(x_i) Q_i^t}{p_i^t} - \alpha^T \xi \quad \forall t = 1, \ldots, T. \tag{66}
\]

With this definition we have the following series of equalities

\[
\mathbb{E}[\exp(\alpha^T p)] = \mathbb{E}[\exp(\sum D_t + T\alpha^T \xi)] = \mathbb{E} \left[ \exp(T\alpha^T \xi) \mathbb{E}[\exp(\sum D_t)|D_n] \right]. \tag{67}
\]

Conditioned on the data, the sequence \( D_1, \ldots, D_T \), forms a martingale difference sequence. Let \( \xi = [\xi(x_1), \ldots, \xi(x_n)] \). Notice that

\[
-\alpha^T \xi - \frac{2\|\alpha\|}{p_{min}^t} \leq D_t \leq -\alpha^T \xi + \frac{2\|\alpha\|}{p_{min}^t}. \tag{68}
\]

We shall now bound the probability of large deviations of \( D_t \) given history up until time \( t \). This allows us to put a bound on the large deviations of the martingale sum \( \sum_{t=1}^T D_t \). Let \( a \geq 0 \). Using Markov’s inequality we get

\[
P[D_t \geq a|Q_{1:n}^{1:t-1}, D_n] \leq \min_{\gamma > 0} \exp(-\gamma a) \mathbb{E}[\gamma D_t|Q_{1:n}^{1:t-1}, D_n] \leq \min_{\gamma > 0} \exp \left( \frac{2\gamma^2 \|\alpha\|^2}{p_{min}^t} - \gamma a \right) \leq \exp \left( \frac{-a^2}{8\|\alpha\|^2n^2\sqrt{t}} \right). \tag{71}
\]
In the second step we used Hoeffding’s lemma along with the boundedness property of $D_t$ shown in equation 68. The same upper bound can be shown for the quantity $P[D_t \leq a|Q^{1:t-1}, D_n]$. Applying lemma 7 we get with probability at least $1 - \delta$, conditioned on the data, we have

$$\frac{1}{T} \sum_{t=1}^{T} D_t \leq \frac{448||\alpha||^2 n^2 \ln(1/\delta)}{\sqrt{T}} \implies \sum_{t=1}^{T} D_t \leq 112||\alpha||^2 n^2 T^{3/2} \ln(1/\delta). \quad (72)$$

Hence $\sum_{t=1}^{T} D_t$, conditioned on data, has sub-Gaussian tails as shown above. This leads to the following conditional exponential moments bound

$$E[\exp(\sum_{t=1}^{T} D_t)|D_n] = \exp \left( 56||\alpha||^2 n^2 T \sqrt{T} \ln(1/\delta) \right). \quad (73)$$

Finally putting together equations 67, 73 we get

$$E[\exp(\alpha^T \xi)] \leq E[\exp(T \alpha^T \xi) \exp(56||\alpha||^2 n^2 T \sqrt{T})] \leq \exp(\{2T^2 + 56n^2 T \sqrt{T}\||\alpha||^2\}). \quad (74)$$

In the last step we exploited the fact that $-2 \leq \xi(x_i) \leq 2$, and hence by Hoeffding lemma $E[\exp(\alpha^T \xi)] \leq \exp(2||\alpha||^2)$. This leads us to the choice of $\sigma^2 = 2T^2 + 56n^2 T \sqrt{T}$. Substituting this value of $\sigma^2$ in equation 65 we get

$$||A \xi||^2 \leq (2T^2 + 56n^2 T \sqrt{T})(d + 2\sqrt{d \ln(1/\delta)} + 2 \ln(1/\delta)), \quad (75)$$

and hence with probability at least $1 - \delta$,

$$||\hat{\Sigma}^{-1/2} \psi_\xi||^2 = n||A \xi||^2 \leq (2nT^2 + 56n^3 T \sqrt{T})(d + 2\sqrt{d \ln(1/\delta)} + 2 \ln(1/\delta)) \quad (76)$$

We are now ready to prove our main result.

**Proof of theorem**

For $n \geq n_0, \delta$ and $T \geq \max\{T_0, \delta, T_1, \delta\}$ from lemma 3 4 both $\Sigma_z$, and $\hat{\Sigma}$ are invertible with probability at least $1 - \delta, 1 - 4\delta$ respectively. Conditioned on the invertibility of $\Sigma_z$, $\Sigma$ we get from lemmas 5 6 7 $||\Sigma^{-1/2} \Sigma^1/2||^2 \leq 3/2$ and $||\Sigma^1/2 \Sigma_z^{-1/2} \Sigma^1/2||^2 \leq 400/n^2 T^2$, and $||\Sigma^{-1/2} \psi_\xi||^2 \leq ||\Sigma^{-1/2} \psi_\xi||^2 \leq (2nT^2 + 56n^3 T^{3/2})(d + 2\sqrt{d \ln(1/\delta)} + 2 \ln(1/\delta))$ with probability at least $1 - \delta, 1 - 3\delta, 1 - \delta$ respectively. Using lemma 1 and the union bound to add up all the failure probabilities we get the desired result.

## 4 Related Work

A variety of pool based AL algorithms have been proposed in the literature employing various query strategies. However, none of them use unbiased estimates of the risk. One of the simplest strategy for AL is uncertainty sampling, where the active learner queries the point whose label it is most uncertain about. This strategy has been popular in text classification [Lewis and Gale 1994], and information extraction [Settles and Craven 2008]. Usually the uncertainty in the label is calculated using certain information-theoretic criteria such as entropy, or variance of the label distribution. While uncertainty sampling has mostly been used in a probabilistic setting, AL algorithms which learn non-probabilistic classifiers using uncertainty sampling have also been proposed. Tong et al. [2001] proposed an algorithm in this framework where they query the point closest to the current SVM hyperplane. Seung et al. [1992] introduced the query-by-committee (QBC) framework where a committee of potential models, which all agree on the currently labeled data is maintained and, the point where most committee members disagree is considered for querying. In order to design a committee in the QBC framework, algorithms such as query-by-boosting, and query-by-bagging in the discriminative setting [Abe and Mamitsuka 1998], sampling from a Dirichlet distribution over model parameters in the generative setting [McCallum and Nigam 1998] have been proposed. Other frameworks include querying the point, which causes the maximum expected reduction in error [Zhu et al. 2003] Guo and Greiner [2007], variance reducing query strategies such as the ones based on optimal design [Flaherty
et al., 2005; Zhang and Oles, 2000). A very thorough literature survey of different active learning algorithms has been done by Settles (2009). AL algorithms that are consistent and have provable label complexity have been proposed for the agnostic setting for the 0-1 loss in recent years (Dasgupta et al., 2007; Beygelzimer et al., 2009). The IWAL framework introduced in Beygelzimer et al. (2009) was the first AL algorithm with guarantees for general loss functions. However the authors were unable to provide non-trivial label complexity guarantees for the hinge loss, and the squared loss.

UPAL at least for squared losses can be seen as using a QBC based querying strategy where the committee is the entire hypothesis space, and the disagreement among the committee members is calculated using an exponential weighting scheme. However unlike previously proposed committees our committee is an infinite set, and the choice of the point to be queried is randomized.

5 Experimental results

We implemented UPAL, along with the standard passive learning (PL) algorithm, and a variant of UPAL called RAL (in short for random active learning), all using logistic loss, in matlab. The choice of logistic loss was motivated by the fact that BMAL was designed for logistic loss. Our matlab codes were vectorized to the maximum possible extent so as to be as efficient as possible. RAL is similar to UPAL, but in each round samples a point uniformly at random from the currently unqueried pool. However it does not use importance weights to calculate an estimate of the risk of the classifier. The purpose of implementing RAL was to demonstrate the potential effect of using unbiased estimators, and to check if the strategy of randomly querying points helps in active learning.

We also implemented a batch mode active learning algorithm introduced by Hoi et al. (2006) which, we shall call as BMAL. Hoi et al. in their paper showed superior empirical performance of BMAL over other competing pool based active learning algorithms, and this is the primary motivation for choosing BMAL as a competitor pool AL algorithm in this paper. BMAL like UPAL also proceeds in rounds and in each iteration selects \( k \) examples by minimizing the Fisher information ratio between the current unqueried pool and the queried pool. However a point once queried by BMAL is never requeried. In order to tackle the high computational complexity of optimally choosing a set of \( k \) points in each round, the authors suggested a monotonic submodular approximation to the original Fisher ratio objective, which is then optimized by a greedy algorithm. At the start of round \( t + 1 \) when, BMAL has already queried \( t \) points in the previous rounds, in order to decide which point to query next, BMAL has to calculate for each potential new query a dot product with all the remaining unqueried points. Such a calculation when done for all possible potential new queries takes \( O(n^2t) \) time. Hence if our budget is \( B \), then the total computational complexity of BMAL is \( O(n^2B^2) \). Note that this calculation does not take into account the complexity of solving an optimization problem in each round after having queried a point. In order to further reduce the computational complexity of BMAL in each round we further restrict our search, for the next query, to a small subsample of the current set of unqueried points. We set the value of \( p_{\min} \) in step 3 of algorithm 1 to \( \frac{1}{n^t} \). In order to avoid numerical problems we implemented a regularized version of UPAL where the term \( \lambda \|w\|^2 \) was added to the optimization problem shown in step 11 of Algorithm 1. The value of \( \lambda \) is allowed to change as per the current importance weight of the pool. The optimal value of \( C \) in VW\(^1\) was chosen via a 5 fold cross-validation, and by eyeballing for the value of \( C \) that gave the best cost-accuracy trade-off. We ran all our experiments on the MNIST dataset(3 Vs 5)\(^2\) and datasets from UCI repository namely Statlog, Abalone, Whitewine. Figure \(^1\) shows the performance of all the algorithms on the first 300 queried points. On the MNIST dataset, on an average, the performance of BMAL is very similar to UPAL, and there is a noticeable gap in the performance of BMAL and UPAL over PL, VW and RAL. Similar results were also seen in the case of Statlog dataset, though towards the end the performance of UPAL slightly worsens when compared to BMAL. However UPAL is still better than PL, VW, and RAL.

\(^1\)The parameters initial\( t, l \) were set to a default value of 10 for all of our experiments.

\(^2\)The dataset can be obtained from http://cs.nyu.edu/~roweis/data.html. We first performed PCA to reduce the dimensions to 25 from 784.
Figure 1: Empirical performance of passive and active learning algorithms. The x-axis represents the number of points queried, and the y-axis represents the test error of the classifier. The subsample size for approximate BMAL implementation was fixed at 300.

| Sample size | UPAL Time | UPAL Error | BMAL Time | BMAL Error |
|-------------|-----------|------------|-----------|------------|
| 1200        | 65        | 7.27       | 60        | 5.67       |
| 2400        | 100       | 6.25       | 152       | 6.05       |
| 4800        | 159       | 6.83       | 295       | 6.25       |
| 10000       | 478       | 5.85       | 643.17    | 5.85       |

Table 1: Comparison of UPAL and BMAL on MNIST data-set of varying training sizes, and with the budget being fixed at 300. The error rate is in percentage, and the time is in seconds.
Active learning is not always helpful and the success story of AL depends on the match between the marginal distribution and the hypothesis class. This is clearly reflected in Abalone where the performance of PL is better than UPAL at least in the initial stages and is never significantly worse. UPAL is uniformly better than BMAL, though the difference in error rates is not significant. However the performance of RAL, VW are significantly worse. Similar results were also seen in the case of Whitewine dataset, where PL outperforms all AL algorithms. UPAL is better than BMAL most of the times. Even here one can witness a huge gap in the performance of VW and RAL over PL, BMAL and UPAL.

One can conclude that VW though is computationally efficient has higher error rate for the same number of queries. The uniformly poor performance of RAL signifies that querying uniformly at random does not help. On the whole UPAL and BMAL perform equally well, and we show via our next set of experiments that UPAL has significantly better scalability, especially when one has a relatively large budget $B$.

### 5.1 Scalability results

Each round of UPAL takes $O(n)$ plus the time to solve the optimization problem shown in step 11 in Algorithm 1. A similar optimization problem is also solved in the BMAL problem. If the cost of solving this optimization problem in step $t$ is $c_{opt,t}$, then the complexity of UPAL is $O(nT + \sum_{t=1}^{T} c_{opt,t})$. While BMAL takes $O(n^2B^2 + \sum_{t=1}^{T} c'_{t,opt})$ where $c'_{t,opt}$ is the complexity of solving the optimization problem in BMAL in round $t$. For the approximate implementation of BMAL that we described if the subsample size is $|S|$, then the complexity is $O(|S|^2B^2 + \sum_{t=1}^{T} c'_{t,opt})$.

In our first set of experiments we fix the budget $B$ to 300, and calculate the test error and the combined training and testing time of both BMAL and UPAL for varying sizes of the training set. All the experiments were performed on the MNIST dataset. Table 1 shows that with increasing sample size UPAL tends to be more efficient than BMAL, though the gain in speed that we observed was at most a factor of 1.8.

In the second set of scalability experiments we fixed the training set size to 10000, and studied the effect of increasing budget. We found out that with increasing budget size the speedup of UPAL over BMAL increases. In particular when the budget was 2000, UPAL is approximately 7 times faster than BMAL. All our experiments were run on a dual core machine with 3 GB memory.

### 6 Conclusions and Discussion

In this paper we proposed the first unbiased pool based active learning algorithm, and showed its good empirical performance and its ability to scale both with higher budget constraints and large dataset sizes. Theoretically we proved that when the true hypothesis is a linear hypothesis, we are able to recover it with high probability. In our view an important extension of this work would be to establish tighter bounds on the excess risk. It should be possible to provide upper bounds on the excess risk in expectation which are much sharper than our current high probability bounds. Another theoretically interesting question is to calculate how many unique queries are made after $T$ rounds of UPAL. This problem is similar to calculating the number of non-empty bins in the balls-and-bins model commonly used in the field of randomized algorithms Motwani and Raghavan (1995), when there are $n$ bins and $T$ balls, with the different points in the pool being the bins, and the process of throwing a ball in each round being equivalent to querying a point in each round.
However since each round is, unlike standard balls-and-bins, dependent on the previous round we expect the analysis to be more involved than a standard balls-and-bins analysis.

References

N. Abe and H. Mamitsuka. Query learning strategies using boosting and bagging. In *ICML*, 1998.

E.B. Baum and K. Lang. Query learning can work poorly when a human oracle is used. In *ICML*, 1992.

A. Beygelzimer, S. Dasgupta, and J. Langford. Importance weighted active learning. In *ICML*, 2009.

N. Cesa-Bianchi and G. Lugosi. *Prediction, learning, and games*. Cambridge Univ Press, 2006.

W. Chu, M. Zinkevich, L. Li, A. Thomas, and B. Tseng. Unbiased online active learning in data streams. In *SIGKDD*, 2011.

D. Cohn, L. Atlas, and R. Ladner. Improving generalization with active learning. *Machine Learning*, 15(2), 1994.

S. Dasgupta, D. Hsu, and C. Monteleoni. A general agnostic active learning algorithm. *NIPS*, 2007.

Patrick Flaherty, Michael I. Jordan, and Adam P. Arkin. Robust design of biological experiments. In *Neural Information Processing Systems*, 2005.

Y. Guo and R. Greiner. Optimistic active learning using mutual information. In *IJCAI*, 2007.

S.C.H. Hoi, R. Jin, J. Zhu, and M.R. Lyu. Batch mode active learning and its application to medical image classification. In *ICML*, 2006.

R.A. Horn and C.R. Johnson. *Matrix analysis*. Cambridge Univ Press, 1990.

D. Hsu, S.M. Kakade, and T. Zhang. An analysis of random design linear regression. *Arxiv preprint arXiv:1106.2363*, 2011a.

D. Hsu, S.M. Kakade, and T. Zhang. Dimension-free tail inequalities for sums of random matrices. *Arxiv preprint arXiv:1104.1672*, 2011b.

J. Langford, L. Li, A. Strehl, D. Hsu, N. Karampatziakis, and M. Hoffman. Vowpal wabbit, 2011.

D.D. Lewis and W.A. Gale. A sequential algorithm for training text classifiers. In *SIGIR*, 1994.

AE Litvak, A. Pajor, M. Rudelson, and N. Tomczak-Jaegermann. Smallest singular value of random matrices and geometry of random polytopes. *Advances in Mathematics*, 195(2):491–523, 2005.

A.K. McCallum and K. Nigam. Employing EM and pool-based active learning for text classification. In *ICML*, 1998.

Rajeev Motwani and Prabhakar Raghavan. *Randomized Algorithms*. Cambridge University Press, 1st edition, August 1995.

J. Quinonero, M. Sugiaima, A. Schwaighofer, and N.D. Lawrence. Dataset shift in machine learning, 2008.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Neural Information Processing Systems*, 2007.

V. Rokhlin and M. Tygert. A fast randomized algorithm for overdetermined linear least-squares regression. *Proceedings of the National Academy of Sciences*, 105(36):13212, 2008.
B. Settles and M. Craven. An analysis of active learning strategies for sequence labeling tasks. In EMNLP, 2008.

Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

H.S. Seung, M. Opper, and H. Sompolinsky. Query by committee. In COLT, pages 287–294. ACM, 1992.

O. Shamir. A variant of azuma’s inequality for martingales with subgaussian tail. Arxiv preprint arXiv:1110.2392, 2011.

S. Tong and E. Chang. Support vector machine active learning for image retrieval. In Proceedings of the ninth ACM international conference on Multimedia, 2001.

J.A. Tropp. User-friendly tail bounds for sums of random matrices. Arxiv preprint arXiv:1004.4389, 2010.

Sara van de Geer. Empirical processes in m-estimation. 2000.

T. Zhang. Statistical behavior and consistency of classification methods based on convex risk minimization. Annals of Statistics, 32(1), 2004.

T. Zhang and F. Oles. The value of unlabeled data for classification problems. In ICML, 2000.

Xiaojin Zhu, John Lafferty, and Zoubin Ghahramani. Combining active learning and semi-supervised learning using gaussian fields and harmonic functions. In ICML, 2003.

A Some results from random matrix theory

**Theorem 4.** (Quadratic forms of subgaussian random vectors) Let \( A \in \mathbb{R}^{m \times n} \) be a matrix, and \( H = AA^T \), and \( r = (r_1, \ldots, r_n) \) be a random vector such that for some \( \sigma \geq 0 \),

\[
\mathbb{E}[\exp(\alpha^T r)] \leq \exp\left(\frac{||\alpha||^2 \sigma^2}{2}\right)
\]

for all \( \alpha \in \mathbb{R}^n \) almost surely. For all \( \delta \in (0, 1) \),

\[
\mathbb{P}\left[||Ar||^2 > \sigma^2 \text{tr}(H) + 2\sigma^2 \sqrt{\text{tr}(H^2)} \ln(1/\delta) + 2\sigma^2 ||H|| \ln(1/\delta)\right] \leq \delta.
\]

The above theorem was first proved without explicit constants by Litvak et al. (Litvak et al., 2005) Hsu et al (Hsu et al., 2011a) established a version of the above theorem with explicit constants.

**Theorem 5.** (Eigenvalue bounds of a sum of rank-1 matrices) Let \( r_1, \ldots, r_n \) be random vectors in \( \mathbb{R}^d \) such that, for some \( \gamma > 0 \),

\[
\mathbb{E}[r_i^T r_i^T | r_1, \ldots, r_{i-1}] = I
\]

\[
\mathbb{E}[\exp(\alpha^T r_i) | r_1, \ldots, r_{i-1}] \leq \exp(||\alpha||^2 \gamma/2) \forall \alpha \in \mathbb{R}^d.
\]

For all \( \delta \in (0, 1) \),

\[
\mathbb{P}\left[\lambda_{\max}\left(\frac{1}{n} \sum_{i=1}^n r_i r_i^T\right) > 1 + 2\epsilon_{\delta,n} \vee \lambda_{\min}\left(\frac{1}{n} \sum_{i=1}^n r_i r_i^T\right) < 1 - 2\epsilon_{\delta,n}\right] \leq \delta,
\]

where

\[
\epsilon_{\delta,n} = \gamma \left(\sqrt{\frac{32(d \ln(5) + \ln(2/\delta))}{n}} + \frac{2(d \ln(5) + \ln(2/\delta))}{n}\right).
\]
We shall use the above theorem in Lemma 3, and lemma 2.

**Theorem 6.** (Matrix Bernstein bound) Let $X_1, \ldots, X_n$ be symmetric valued random matrices. Suppose there exist $b, \sigma$ such that for all $i = 1, \ldots, n$

$$
E_i[X_i] = 0 \\
\lambda_{\max}(X_i) \leq \bar{b} \\
\lambda_{\max}\left(\frac{1}{n} \sum_{i=1}^{n} E_i[X_i^2]\right) \leq \bar{\sigma}^2.
$$

almost surely, then

$$
P\left[\lambda_{\max}\left(\frac{1}{n} \sum_{i=1}^{n} X_i\right) > \sqrt{\frac{2\bar{\sigma}^2 \ln(d/\delta)}{n}} + \frac{\bar{b} \ln(d/\delta)}{3n}\right] \leq \delta. \tag{77}
$$

A dimension free version of the above inequality was proved in Hsu et al. Such dimension free inequalities are especially useful in infinite dimension spaces. Since we are working in finite dimension spaces, we shall stick to the non-dimension free version.

**Theorem 7.** (Shamir, 2011) Let $(Z_1, F_1), \ldots, (Z_T, F_T)$ be a martingale difference sequence, and suppose there are constants $b \geq 1, c_t > 0$ such that for any $t$ and any $a > 0$,

$$
\max\{\mathbb{P}[Z_t \geq a | F_{t-1}], \mathbb{P}[Z_t \leq -a | F_{t-1}]\} \leq b \exp(-c_t a^2).
$$

Then for any $\delta > 0$, with probability atleast $1 - \delta$ we have

$$
\frac{1}{T} \sum_{t=1}^{T} Z_t \leq \sqrt{\frac{28b \ln(1/\delta)}{\sum_{t=1}^{T} c_t}}.
$$

The above result was first proved by Shamir. Shamir proved the result for the case when $c_1 = \ldots = c_T$. Essentially one can use the same proof with obvious changes to get the above result.

**Lemma 8** (Hoeffding’s lemma). (see Cesa-Bianchi and Lugosi, 2006, page 359) Let $X$ be a random variable with $a \leq X \leq b$. Then for any $s \in \mathbb{R}$

$$
\mathbb{E}[\exp(sX)] \leq \exp\left(s\mathbb{E}[X] + \frac{s^2(b-a)^2}{8}\right). \tag{78}
$$

**Theorem 8.** Let $A, B$ be positive semidefinite matrices. Then

$$
\lambda_{\max}(A) + \lambda_{\min}(B) \leq \lambda_{\max}(A + B) \leq \lambda_{\max}(A) + \lambda_{\max}(B).
$$

The above inequalities are called as Weyl’s inequalities (see Horn and Johnson, 1990, chap. 3)