Random Matrix Theory Improvements on the Matched Subspace Classifier

Nicholas Asendorf

August 24, 2011

1 Problem Statement

We consider the classification problem where our observed data, y, may be one of two classes. We may either observe signal in the presence of noise, or simply noise itself. Our setup is as follows:

$$y = \begin{cases} z & y \in H_0 \\ U_1 x + z & y \in H_1 \end{cases} \tag{1}$$

where $z \sim \mathcal{N}(0, I)$, $U_1 \in \mathbb{C}^{n \times k}$ is unknown with orthonormal columns, $x \sim \mathcal{N}(0, \Sigma_1)$ with $\Sigma_1 = \mathbf{diag}(\sigma_1^2, \ldots, \sigma_k^2)$ with σ_i^2 unknown. We also assume that x and z are independent.

We are given labeled training data $y_1, \ldots, y_m$, with $y_i \in H_1$ for $i = 1, \ldots, m$. We will use this training data to form estimates $\hat{U}_1$, $\hat{\Sigma}_1$ of our unknown parameters U_1, Σ_1.

We consider the processed data $w = \hat{U}_1^H y \in \mathbb{C}^n$. We are also given unlabeled testing data $y_1, \ldots, y_r$. Our goal is to determine a classifier, $g(w) \to \{0, 1\}$ which solves the following problem for our testing data:

$$\begin{array}{ll} \text{maximize} & P_D = P\left(g(w) = 1 | w \in H_1\right) \\ \text{subject to} & P_F = P\left(g(w) = 1 | w \in H_0\right) = \alpha \end{array} \tag{2}$$

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

2 Parameter Estimation

We have two unknown parameters, U_1, Σ_1. Using our training data, $\{y_1, \ldots, y_m\}$, we make estimate of these parameters. To do so, we form the matrix $Y = [y_1, \ldots, y_m]$ by stacking the training data as columns in a matrix. Define $S_1 = \frac{1}{m} Y Y^H$ as the sample covariance of our training data. By properties of Gaussian random variables, under H_1, $y_i \sim \mathcal{N}(0, U_1 \Sigma_1 U_1^H + I)$. Taking $U_2 = U_1^\perp$ to be the orthogonal complement of U_1 we may write this covariance as

$$\begin{aligned} U_1 \Sigma U_1^H + I &= \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U_1^H \\ U_2^H \end{bmatrix} + \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} U_1^H \\ U_2^H \end{bmatrix} \\ &= \begin{bmatrix} U_1 & U_2 \end{bmatrix} \left(\begin{bmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} I_k & 0 \\ 0 & I_{n-k} \end{bmatrix} \right) \begin{bmatrix} U_1^H \\ U_2^H \end{bmatrix} \\ &= \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_1 + I_k & 0 \\ 0 & I_{n-k} \end{bmatrix} \begin{bmatrix} U_1^H \\ U_2^H \end{bmatrix} \end{aligned}$$

Clearly this is the in the form of an eigenvalue decomposition of our covariance matrix. Therefore if we take the eigenvalue decomposition of the sample covariance matrix, S, we can form an estimate of our subspace U_1 and our covariances σ_i^2. Defining the eigenvalue decomposition $S_1 = V\Lambda V^H$ where $\Lambda = \mathbf{diag}(\lambda_1, \ldots, \lambda_n)$ and $V = [v_1, \ldots, v_n]$ such that $\lambda_1 > \lambda_2 > \cdots > \lambda_n$ we have

$$\begin{aligned} \hat{U}_1 &= [v_1 \ldots v_k] \\ \hat{\sigma}_i^2 &= \lambda_i - 1 \text{ for } i = 1, \ldots, k \end{aligned} \tag{3}$$

We also define $\hat{\Sigma}_1 = \mathbf{diag}(\hat{\sigma}_1^2, \ldots, \hat{\sigma}_k^2)$.

3 Random Matrix Theory Estimates

If we are given a sample covariance matrix, $S = \frac{1}{m} YY^H$, where the columns of Y are drawn from $y \sim \mathcal{N}(0, \Psi)$ where $\Psi = \mathbf{diag}(\lambda_1, \ldots, \lambda_k, 1, \ldots, 1) \in \mathbf{R}^n$, theorem 3 of [3] tells us that

$$\hat{\lambda}_i \sim \mathcal{N}\left(\lambda_i\left(1 + \frac{c}{\lambda_i - 1}\right), 2\lambda_i^2\left(1 - \frac{c}{(\lambda_i - 1)^2}\right)\right) \quad \text{if } \lambda_i > 1 + \sqrt{c} \tag{4}$$

where $c = \frac{n}{m}$ and $\hat{\lambda}_i$ are the eigenvalues of S.

Because our sample covariance matrix S_1 takes this form, we may apply this theorem to our problem at hand:

$$\begin{aligned} \left(\hat{\sigma}_i^2 + 1\right) &\sim \mathcal{N}\left(\left(\sigma_i^2 + 1\right)\left(1 + \frac{c}{\sigma_i^2}\right), \frac{2\left(\sigma_i^2 + 1\right)^2}{n}\left(1 - \frac{c}{\sigma_i^4}\right)\right) && \text{if } \sigma_i^2 + 1 > 1 + \sqrt{c} \\ \hat{\sigma}_i^2 &\sim \mathcal{N}\left(\left(c + \sigma_i^2 + \frac{c}{\sigma_i^2}\right), \frac{2\left(\sigma_i^2 + 1\right)^2}{n}\left(1 - \frac{c}{\sigma_i^4}\right)\right) && \text{if } \sigma_i^2 > \sqrt{c} \end{aligned}$$

We may then make an estimate of σ_i^2 by maximizing the likelihood of $\hat{\sigma}_i^2$.

$$\begin{aligned} \hat{\sigma}_{i_{\text{rmt}}}^2 &= \operatorname*{argmax}_{\sigma_i^2} f_{\hat{\sigma}_i^2} \\ &= \operatorname*{argmax}_{\sigma_i^2} \left[\left(2\pi\left(\frac{2\left(\sigma_i^2 + 1\right)^2}{n}\left(1 - \frac{c}{\sigma_i^4}\right)\right)\right)^{-1/2} \exp\left\{\frac{-n}{4\left(\sigma_i^2 + 1\right)^2\left(1 - \frac{c}{\sigma_i^4}\right)}\left(\hat{\sigma}_i^2 - \left(\sigma_i^2 + c + \frac{c}{\sigma_i^2}\right)\right)\right\}\right] \end{aligned}$$

We may instead maximize the log likelihood and after simplification, when $\sigma_i^2 > \sqrt{c}$ our estimate becomes

$$\boxed{\hat{\sigma}_{i_{\text{rmt}}}^2 = \operatorname*{argmax}_{\sigma_i^2} \left[-\log\left(\sigma_i^2 + 1\right) + \frac{1}{2}\log(n) - \frac{1}{2}\log\left(1 - \frac{c}{\sigma_i^4}\right) - \frac{n}{4\left(\sigma_i^2 + 1\right)^2\left(1 - \frac{c}{\sigma_i^4}\right)}\left(\hat{\sigma}_i^2 - \left(\sigma_i^2 + c + \frac{c}{\sigma_i^2}\right)\right)\right]} \tag{5}$$

However, (5) depends on whether $\sigma_i^2 > \sqrt{c}$ which is unknown. Therefore, the first step is to determine the number which σ_i^2 are indeed larger than $\sqrt{c}$. To do so, we apply a statistical test proposed described in [2]

(Algorithm 2) which determines the number of factors present given the eigenvalues of the sample covariance matrix. We use a significance level of $\alpha = 0.1$. This algorithm returns q, the number of factors present. We then have the following modified estimate for σ_i

$$\hat{\sigma}^2_{i_{\mathrm{rmt}}} = \begin{cases} (5) & i \leq q \\ 0 & i > q \end{cases} \tag{6}$$

From Theorem 4 of [3], we also have that

$$| < v_i, \hat{v}_i > |^2 \rightarrow \begin{cases} 0 & \text{if } \lambda_i \leq 1 + \sqrt{c} \\ \frac{1 - \frac{c}{(\lambda - 1)^2}}{1 + \frac{c}{\lambda - 1}} & \text{if } \lambda_i > 1 + \sqrt{c} \end{cases} \tag{7}$$

where $\hat{v}_i$ is the eigenvector of the sample covariance matrix corresponding to the eigenvalue λ_i and v_i is the true underlying eigenvalue. Applying this theorem to our problem, we have

$$\begin{aligned} | < u_i, \hat{u}_i > |^2 \quad & \rightarrow \begin{cases} 0 & \text{if } \sigma_i^2 + 1 \leq 1 + \sqrt{c} \\ \frac{1 - \frac{c}{(\sigma_i^2 + 1 - 1)^2}}{1 + \frac{c}{\sigma_i^2 + 1 - 1}} & \text{if } \sigma_i^2 + 1 > 1 + \sqrt{c} \end{cases} \\ & \rightarrow \begin{cases} 0 & \text{if } \sigma_i^2 \leq \sqrt{c} \\ \frac{\frac{\sigma_i^4 - c}{\sigma_i^4}}{\frac{\sigma_i^2 + c}{\sigma_i^2}} & \text{if } \sigma_i^2 > \sqrt{c} \end{cases} \\ & \rightarrow \begin{cases} 0 & \text{if } \sigma_i^2 \leq \sqrt{c} \\ \frac{\sigma_i^4 - c}{\sigma_i^4 + \sigma_i^2 c} & \text{if } \sigma_i^2 > \sqrt{c} \end{cases} \end{aligned}$$

We then substitute our expression for σ_i^2 derived in (6)

$$\boxed{| < u_i, \hat{u}_i > |^2_{\mathrm{rmt}} \rightarrow \begin{cases} 0 & \text{if } \hat{\sigma}^2_{i_{\mathrm{rmt}}} \leq \sqrt{c} \\ \frac{\hat{\sigma}^4_{i_{\mathrm{rmt}}} - c}{\hat{\sigma}^4_{i_{\mathrm{rmt}}} + \hat{\sigma}^2_{i_{\mathrm{rmt}}} c} & \text{if } \hat{\sigma}^2_{i_{\mathrm{rmt}}} > \sqrt{c} \end{cases}} \tag{8}$$

4 Processed Matched Subspace Classifier

By properties of Gaussian random variables, under H_0, $y \sim \mathcal{N}(0, I)$ and under H_1, $y \sim \mathcal{N}(0, U_1 \Sigma_1 U_1^H + I)$. For our processed data, $w = \hat{U}_1^H$, using properties of Gaussian random variables, under H_0, $w \sim \mathcal{N}(0, I_k)$ and under H_1, $w \sim \mathcal{N}(0, \hat{U}_1^H U_1 \Sigma_1 U_1^H \hat{U}_1 + I)$.

We will consider 3 different classifiers of our data w. The first is an oracle classifier, which will assume that U_1 and Σ_1 are known. The purpose of this is to give an upper bound on a classifier's performance. The second classifier is a plug-in classifier which will approximate the oracle classifier by simply plugging in our estimates $\hat{U}_1$, $\hat{\Sigma}_1$ for our unknown U_1 and Σ_1. The third classifier uses the results of random matrix theory to form an approximation to the oracle classifier.

The Neyman Pearson Lemma [**?**] states that the likelihood ratio test (LRT) which solves (2) is the most powerful test of size α. Our LRT takes the form

$$\Lambda(w) = \frac{f(w|H_1)}{f(w|H_0)}$$

and our decision is

$$g(w) = \begin{cases} 0 & \Lambda(w) < \eta \\ 1 & \Lambda(w) > \eta \end{cases}$$

where $P(\Lambda(w) \leq \eta | H_0) = \alpha$.

4.1 Oracle Classifier

Our LRT for our processed data w, is

$$\begin{aligned} \Lambda(w) \quad &= \frac{(2\pi)^{-k/2}|\hat{U}_1^H U_1 \Sigma_1 U_1^H \hat{U}_1 + I|^{-1/2} \exp\{-\frac{1}{2} w^H \left[\hat{U}_1^H U_1 \Sigma_1 U_1^H \hat{U}_1 + I\right]^{-1} w\}}{(2\pi)^{-k/2} \exp\{-\frac{1}{2} w^H w\}} \\ &= |\hat{U}_1^H U_1 \Sigma_1 U_1^H \hat{U}_1 + I|^{-1/2} \exp\{-\frac{1}{2} w^H \left[\left(\hat{U}_1^H U_1 \Sigma_1 U_1^H \hat{U}_1 + I\right)^{-1} - I\right] w\} \end{aligned} \tag{9}$$

where, defining $\eta = \frac{P(y \in H_0)}{P(y \in H_1)}$ our classifier is

$$g_{\text{oracle}}(w) = \begin{cases} 0 & \text{if } \Lambda(w) < \eta \\ 1 & \text{if } \Lambda(w) > \eta \end{cases} \tag{10}$$

We may apply the natural logarithm operator to both sides as it is a monotonic operation. Our statistic becomes

$$\boxed{\Lambda_{\text{oracle}}(w) = w^H \left[I - \left(\hat{U}_1^H U_1 \Sigma_1 U_1^H \hat{U}_1 + I\right)^{-1}\right] w} \tag{11}$$

and defining a threshold $\gamma = 2 \ln \left(\eta |\hat{U}_1^H U_1 \Sigma_1 U_1^H \hat{U}_1 + I|^{1/2}\right)$ we have the classifier

$$\boxed{g_{\text{oracle}}(w) = \begin{cases} 0 & \text{if } \Lambda_{\text{oracle}}(w) < \gamma \\ 1 & \text{if } \Lambda_{\text{oracle}}(w) > \gamma \end{cases}} \tag{12}$$

4.2 Plug-in Classifier

We assume that U_1 and Σ_1 are not known, and therefore we cannot compute (11) directly. One solution to this problem is to plug in our estimates $\hat{U}_1$ and $\hat{\Sigma}_1$ wherever U_1 and Σ_1 appear respectively. Using our estimates in (3) have the following plug-in classifier statistic:

$$\begin{aligned} \Lambda_{\text{plugin}}(w) \quad &= w^H \left(I - \left[\hat{U}_1^H \hat{U}_1 \hat{\Sigma}_1 \hat{U}_1^H \hat{U}_1 + I\right]^{-1}\right) w \\ &= w^H \left(I - \left(\hat{\Sigma}_1 + I\right)^{-1}\right) w \\ &= w^H \left(I - \mathbf{diag}\left(\hat{\sigma}_i^2 + 1\right)^{-1}\right) w \end{aligned} \tag{13}$$

This simplifies to

$$\boxed{\Lambda_{\text{plugin}}(w) = w^H \, \mathbf{diag}\left(\frac{\hat{\sigma}_i^2}{1+\hat{\sigma}_i^2}\right) w = \sum_{i=1}^{k} \frac{w_i^2 \hat{\sigma}_i^2}{\hat{\sigma}_i^2 + 1}} \tag{14}$$

and our classifier becomes

$$\boxed{g_{\text{plugin}}(w) = \begin{cases} 0 & \text{if } \Lambda_{\text{plugin}}(w) < \hat{\gamma} \\ 1 & \text{if } \Lambda_{\text{plugin}}(w) > \hat{\gamma} \end{cases}} \tag{15}$$

where $\hat{\gamma} = 2\log(\eta) + \sum_{i=1}^{k} \log\left(\hat{\sigma}_i^2 + 1\right)$.

4.3 Random Matrix Theory Classifier

To utilize our random matrix theory expressions derived in Section 3, we first make a diagonal approximation of (11)

$$\begin{aligned} \tilde{\Lambda}(w) \quad &= w^H \left[I - \left(\hat{U}_1^H U_1 \Sigma_1 U_1^H \hat{U}_1 + I \right)^{-1} \right] w \\ &\approx w^H \left(I - \left[\mathbf{diag}\left(| < u_i, \hat{u}_i > |^2 \sigma_i^2 \right) + I \right]^{-1} \right) w \\ &= w^H \left(\mathbf{diag}\left(\frac{| < u_i, \hat{u}_i > |^2 \sigma_i^2}{| < u_i, \hat{u}_i > |^2 \sigma_i^2 + 1} \right) \right) w \end{aligned} \tag{16}$$

However, σ_i^2 and $| < u_i, \hat{u}_i > |^2$ are unknown and we must use an estimate for them. However, instead of using $\hat{\sigma}_i^2$ and estimating $| < u_i, \hat{u}_i > |^2 = 1$ as the plug-in classifier does, we use expressions derived in Section 3 which considers the error in estimating the eigenvalues and eigenvectors of our sample covariance matrix.

Using (6) and (8) our random matrix theory statistic becomes

$$\boxed{\Lambda_{\text{rmt}}(w) = w^H \, \mathbf{diag}\left(\frac{| < u_i, \hat{u}_i > |_{\text{rmt}}^2 \hat{\sigma}_{i_{\text{rmt}}}^2}{| < u_i, \hat{u}_i > |_{\text{rmt}}^2 \hat{\sigma}_{i_{\text{rmt}}}^2 + 1} \right) w = \sum_{i=1}^{k} \frac{w_i^2 | < u_i, \hat{u}_i > |_{\text{rmt}}^2 \hat{\sigma}_{i_{\text{rmt}}}^2}{| < u_i, \hat{u}_i > |_{\text{rmt}}^2 \hat{\sigma}_{i_{\text{rmt}}}^2 + 1}} \tag{17}$$

and our classifier becomes

$$\boxed{g_{\text{rmt}}(w) = \begin{cases} 0 & \text{if } \Lambda_{\text{rmt}}(w) < \hat{\gamma}_{\text{rmt}} \\ 1 & \text{if } \Lambda_{\text{rmt}}(w) > \hat{\gamma}_{\text{rmt}} \end{cases}} \tag{18}$$

where $\hat{\gamma}_{\text{rmt}} = 2\log(\eta) + \sum_{i=1}^{k} \log\left(\hat{\sigma}_{i_{\text{rmt}}}^2 | < u_i, \hat{u}_i > |^2 + 1\right)$.

5 Theoretical ROC Curve

We wish to determine the theoretical ROC curve for our detectors (14) and (17). To do so, we must compute the probability of detection (P_D) and probability of false alarm (P_F) for each of our statistics. If $\Lambda(w)$ is our statistic of interest, we must compute

$$
\begin{aligned}
P_D &= P(\Lambda \geq y) \\
P_F &= P(\Lambda \geq y)
\end{aligned} \tag{19}
$$

for $-\infty < y < \infty$. To do this, we examine the form of our statistics (14) and (17).

We first consider (14). Under H_0 we have that $w \sim \mathcal{N}(0, I)$ so $w_i \sim \mathcal{N}(0,1)$ are i.i.d for $i = 1, \ldots, k$. So $w_i^2 \sim \chi_1^2$ are i.i.d for $i = 1, \ldots, k$. So under H_0,

$$
\Lambda_{\text{plugin}}(w) = \sum_{i=1}^{k} \left(\frac{\sigma_i^2}{1+\sigma_i^2} \right) \chi_{1i}^2 \tag{20}
$$

That is, our statistic is a weighted sum of independent chi-square random variables with 1 degree of freedom.

Now, under H_1, we have that $w \sim \mathcal{N}(0, \hat{U}_1^H U_1 \Sigma_1 U_1^H \hat{U}_1 + I)$ so $w_i \approx \mathcal{N}(0, \sigma_i^2 |< u_i, \hat{u}_i >|^2 + 1)$ are i.i.d. Therefore,

$$
\frac{w_i^2}{\sigma_i^2 |< u_i, \hat{u}_i >|^2 + 1} \sim \chi_1^2 \tag{21}
$$

Therefore, under H_1,

$$
\Lambda_{\text{plugin}}(w) = \sum_{i=1}^{k} \left(\frac{\sigma_i^2 \left(\sigma_i^2 |< u_i, \hat{u}_i >|^2 + 1 \right)}{1+\sigma_i^2} \right) \chi_{1i}^2 \tag{22}
$$

which is also a weighted sum of independent chi-square random variables with 1 degree of freedom.

Turning to (17), under H_0 we have again that $w \sim \mathcal{N}(0, I)$ so $w_i \sim \mathcal{N}(0,1)$ are i.i.d for $i = 1, \ldots, k$. So $w_i^2 \sim \chi_1^2$ are i.i.d for $i = 1, \ldots, k$. So under H_0,

$$
\Lambda_{\text{rmt}}(w) = \sum_{i=1}^{k} \left(\frac{\sigma_i^2 |< u_i, \hat{u}_i >|_{\text{rmt}}^2}{1 + \sigma_i^2 |< u_i, \hat{u}_i >|_{\text{rmt}}^2} \right) \chi_{1i}^2 \tag{23}
$$

which is a weighted sum of independent chi-square random variables with 1 degree of freedom.

Now, under H_1, we again have that $w \sim \mathcal{N}(0, \hat{U}_1^H U_1 \Sigma_1 U_1^H \hat{U}_1 + I)$ so $w_i \approx \mathcal{N}(0, \sigma_i^2 |< u_i, \hat{u}_i >|^2 + 1)$ are i.i.d. Therefore,

$$
\frac{w_i^2}{\sigma_i^2 |< u_i, \hat{u}_i >|^2 + 1} \sim \chi_1^2 \tag{24}
$$

Therefore, under H_1,

$$
\Lambda_{\text{rmt}}(w) = \sum_{i=1}^{k} \left(\sigma_i^2 |< u_i, \hat{u}_i >|_{\text{rmt}}^2 \right) \chi_{1i}^2 \tag{25}
$$

which is also a weighted sum of independent chi-square random variables with 1 degree of freedom.

Therefore, in all 4 cases, our statistic is a weighted sum of chi-square random variables with 1 degree of freedom. That is

$$\Lambda = \sum_{i=1}^{k} \lambda_i \chi^2_{1i} \tag{26}$$

where λ_i is the appropriate weighting, unique to each statistic under each hypothesis.

To evaluate (19) we need to compute the CDF of a weighted sum of independent chi-square random variables. To do so, we utilize the generalized Lugannani-Rice formula proposed in [4]. We have the following approximation

$$P(\Lambda \geq y) \simeq 1 - \Gamma(\hat{\xi}) + \gamma(\hat{\xi}) \left\{ \frac{1}{\hat{u}_{\hat{\xi}}} - \frac{1}{w_{\hat{\xi}}} \right\} \tag{27}$$

where Γ and γ are the CDF and density, respectively, of a chi-squared random variable with α degrees of freedom where

$$\alpha = \frac{2[K'(0)]^2}{K''(0)} \tag{28}$$

where K is the CGF (cumulative generating function) of Λ. K takes the form

$$K(t) = \frac{1}{2} \sum_{i=1}^{k} -\log(1 - 2\lambda_i t) \tag{29}$$

We define G as the CGF of a chi-squared random variable with α degrees of freedom.

Using a line search, we determine $\hat{\xi}$ as

$$\hat{\xi} = \begin{cases} \xi_-(y) & \text{if } y < K'(0) \\ \xi_+(y) & \text{if } y > K'(0) \\ G'(0) & \text{if } y = K'(0) \end{cases} \tag{30}$$

where $\xi_-(y)$ and $\xi_+(y)$ are the solutions to

$$\frac{\alpha\{\log(\xi) - \log(\alpha) + 1\}}{2} - \frac{\xi}{2} = K(\hat{t}) = y\hat{t} \tag{31}$$

where $\hat{t} \in (m_-, m_+)$ is the unique solution to $K'(t) = y$, found by a line search, where

$$\begin{aligned} m_- &= -\infty \\ m_+ &= \frac{1}{2\max\{\lambda_i\}} \end{aligned} \tag{32}$$

To compute (19) we sweep y over $(0, \infty)$ and compute (27). This generates approximate points on the theoretical ROC curve.

6 Simulation Results

We now demonstrate the performance of the three classifiers derived in Section 4 through numerical simulations. To compare classifiers across all thresholds, γ, we generate a Receiver Operating Characteristic (ROC) curve for each classifier. ROC curves plot P_D vs. P_F for a classifier. Curves lying in the northwest regime are the best as they operate with a high probability of detection and a low probability of false-alarm.

To test our classifiers first generate a random U_1 by taking the first k left singular vectors of a random $n \times n$ matrix. Using the desired Σ_1 we generate m training points via (1). We then form our parameter estimates (3) and random matrix theory values (6) and (8) to be used in our classifiers.

We then generate r testing points of each class via (1) and process them via $w = \hat{U}_1^H y$. We calculate our statistic for each testing point for each classifier via (11), (14) and (17).

Using algorithm 1 of [1] we calculate the ROC curve of each classifier by sweeping γ used in (12), (15) and (18)

We then repeat this process multiple times with a different random orthogonal U_1 to generate multiple ROC curves. Using algorithm 4 of [1] we average the ROC curves of each trial to produce a one final ROC curve for each of the three classifiers.

To compare multiple ROC curves against each other, we may utilize the AUC statistic which is the area under the ROC curve. We compute this statistic using algorithm 2 of [1]. An AUC value of 1 represents perfect detection, while an AUC value of 0.5 represents pure guessing.

We consider the $k = 2$ case. We wish to explore how varying Σ changes the behavior of our detectors. To do so, we sweep both σ_1 and σ_2 from 0.1 to 5 in 0.1 increments and calculate the theoretical and empirical AUC for each detector. This produces 2500 AUC points. The following results are for the c=1 case. We can visualize the AUC performance via a heatmap. Figures (**??**)-(3) show the empirical heatmaps generated while figures (4)-(5) show the theoretical prediction for the plugin and RMT detectors. Figures (6)-(7) show the improvement of the RMT over the plugin classifier, both empirically and theoretically. Figure (8) shows the accuracy of the empirical improvement of the RMT detector compared to the theoretical improvement. Figures (9)-(10) show the accuracy of the empirical plugin and RMT detectors to the theoretical detectors. Finally, figure (11) show the accuracy of the RMT detector to the oracle detector.

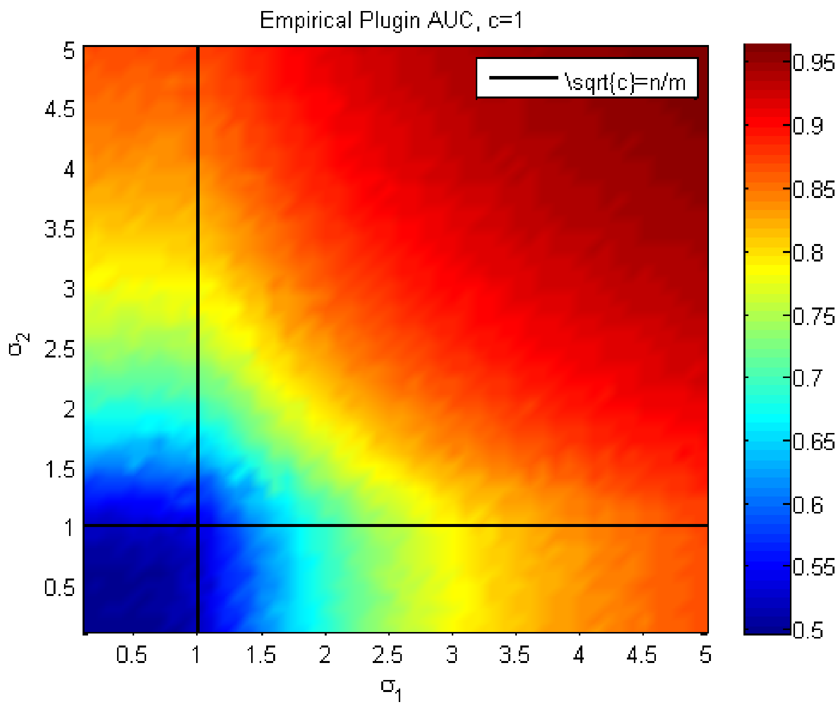

Figure 1: Empirical AUC of the plugin classifer. c=1

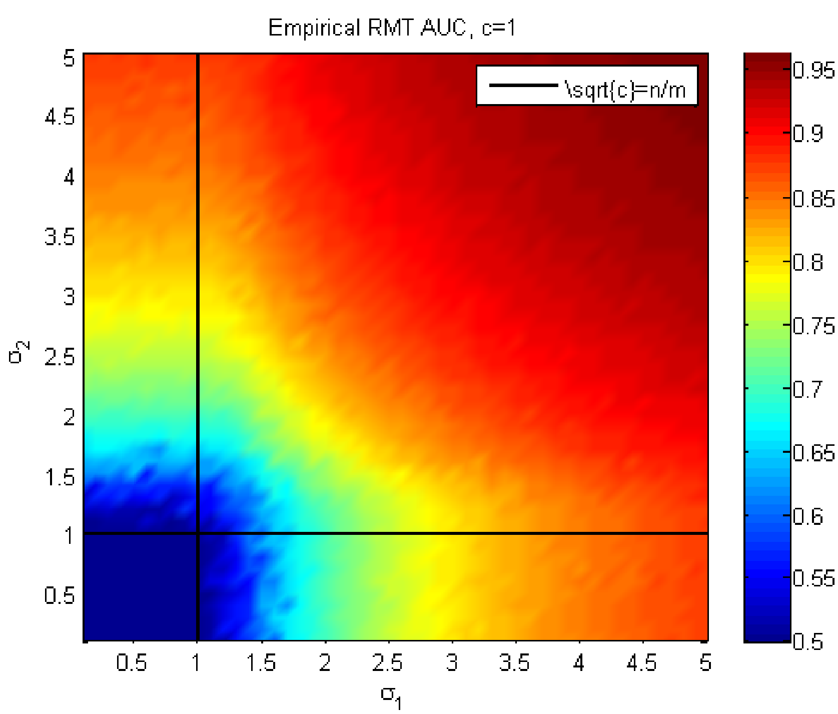

Figure 2: Empirical AUC of the random matrix theory classifier. c=1

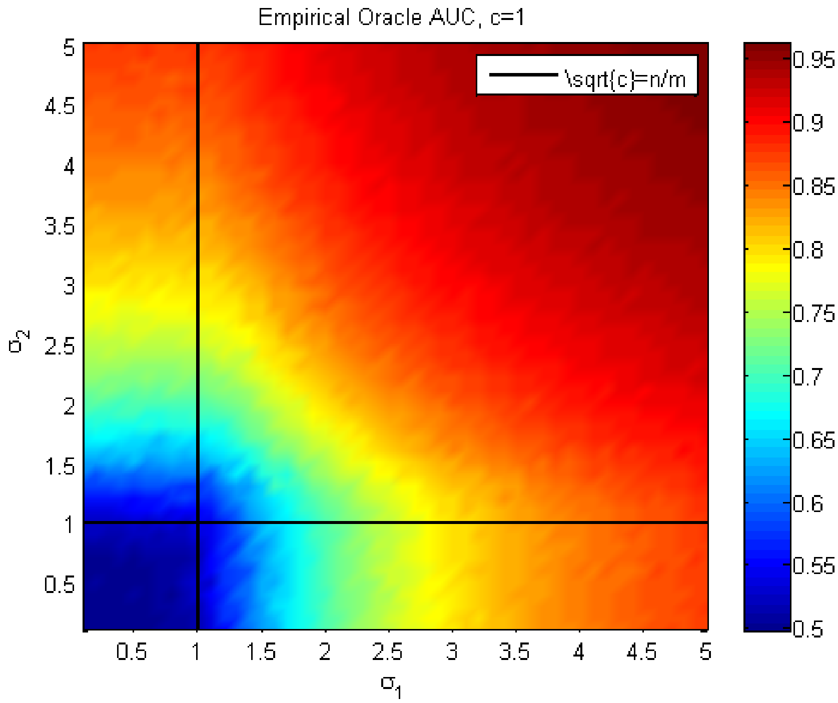

Figure 3: Empirical AUC of the oracle classifier. c=1

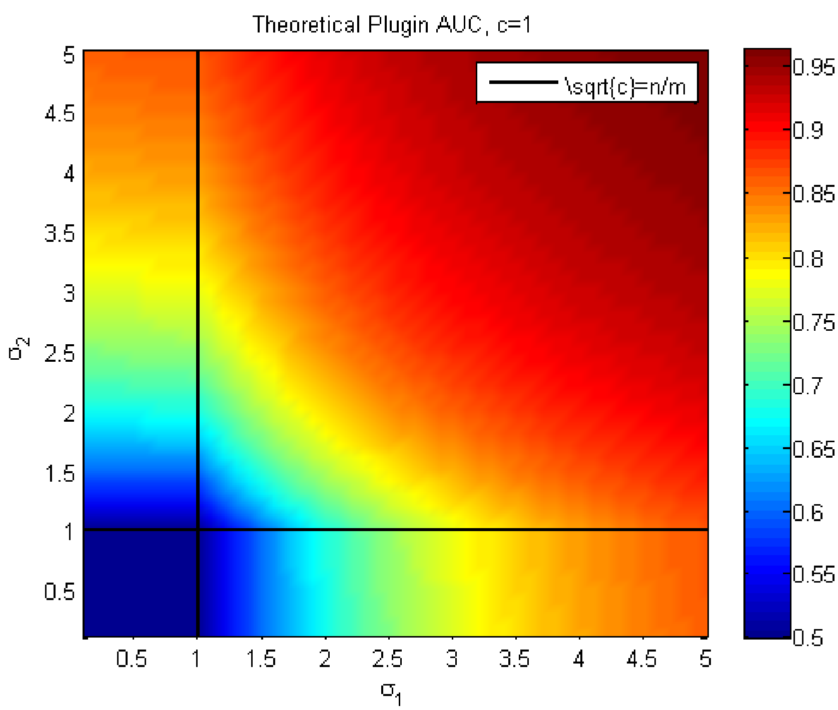

Figure 4: Theoretical AUC of the plugin classifier. c=1

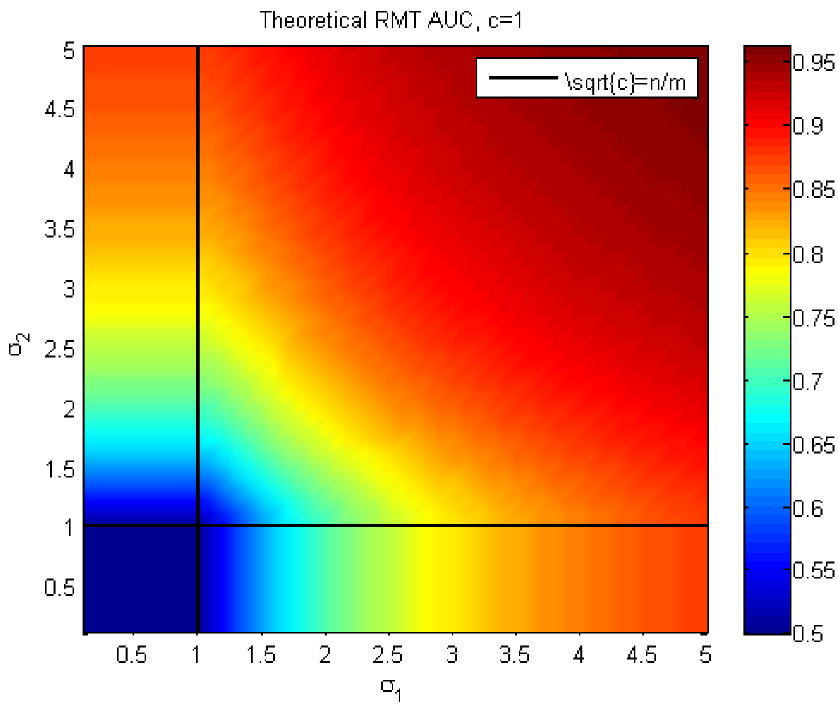

Figure 5: Theoretical AUC of the random matrix theory classifier. c=1

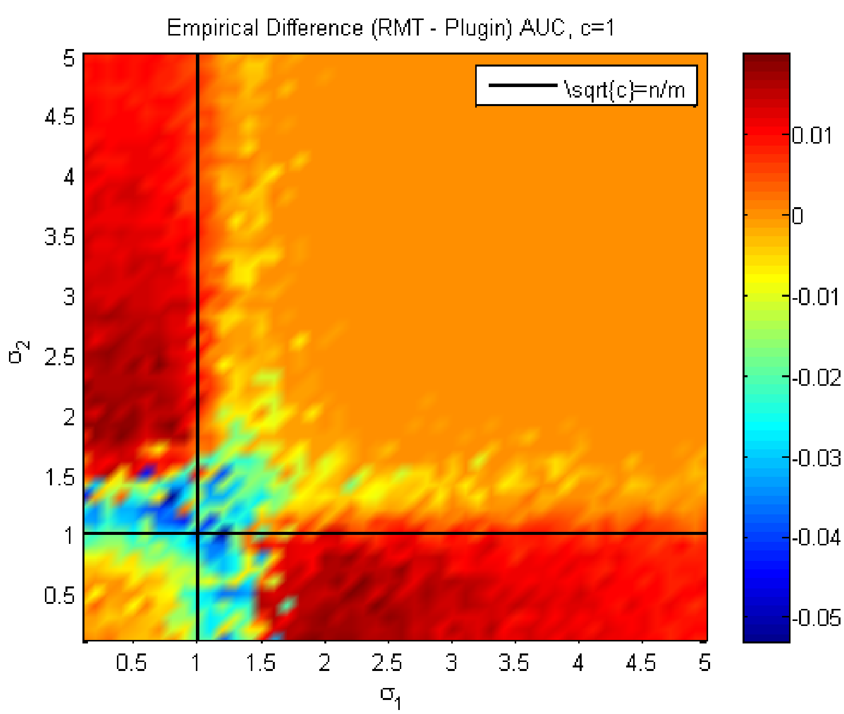

Figure 6: Empirical difference between the RMT and plugin classifier. c=1

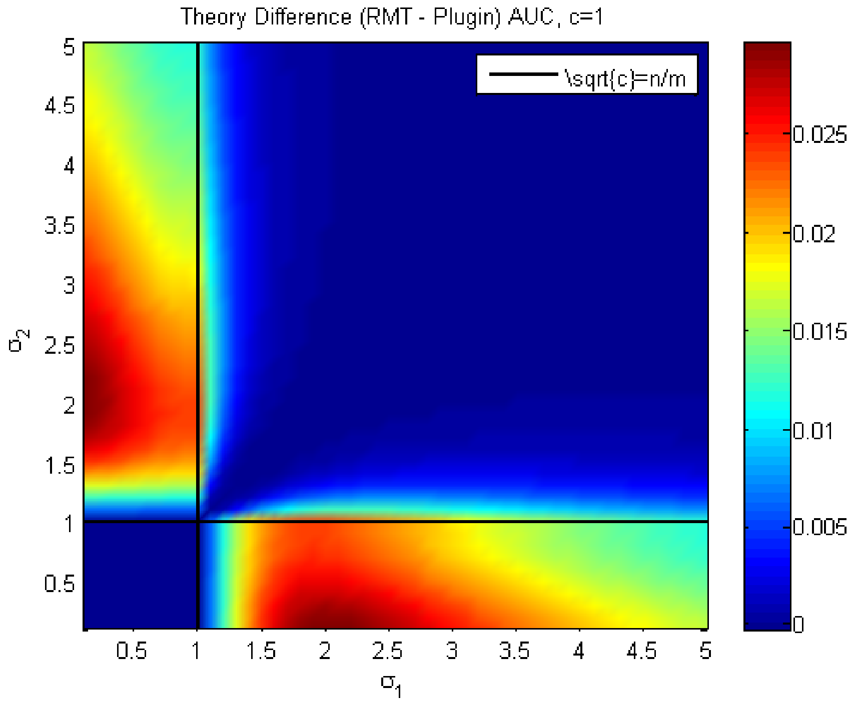

Figure 7: Theoretical difference between the RMT and plugin classifier. c=1

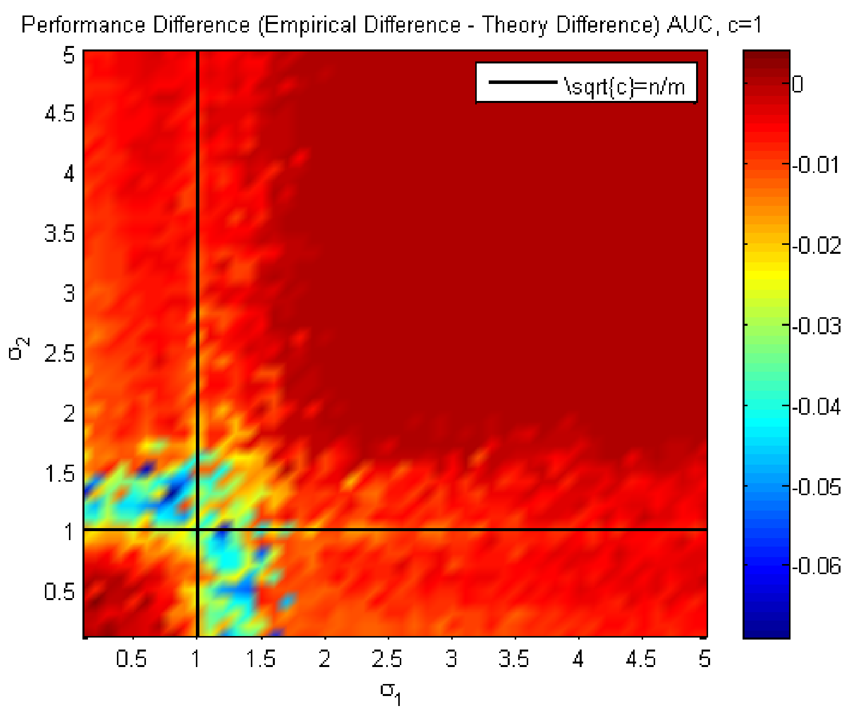

Figure 8: Difference between the empirical RMT improvement over the plugin classifer and the the theoretical improvment. c=1

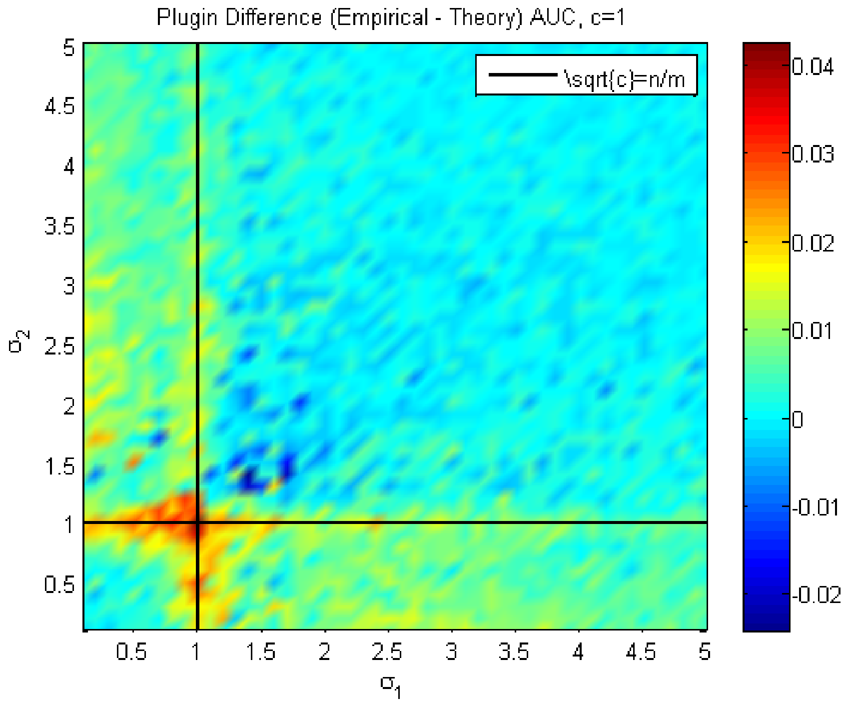

Figure 9: Difference between the empirical and theoretical plugin classsifier. c=1

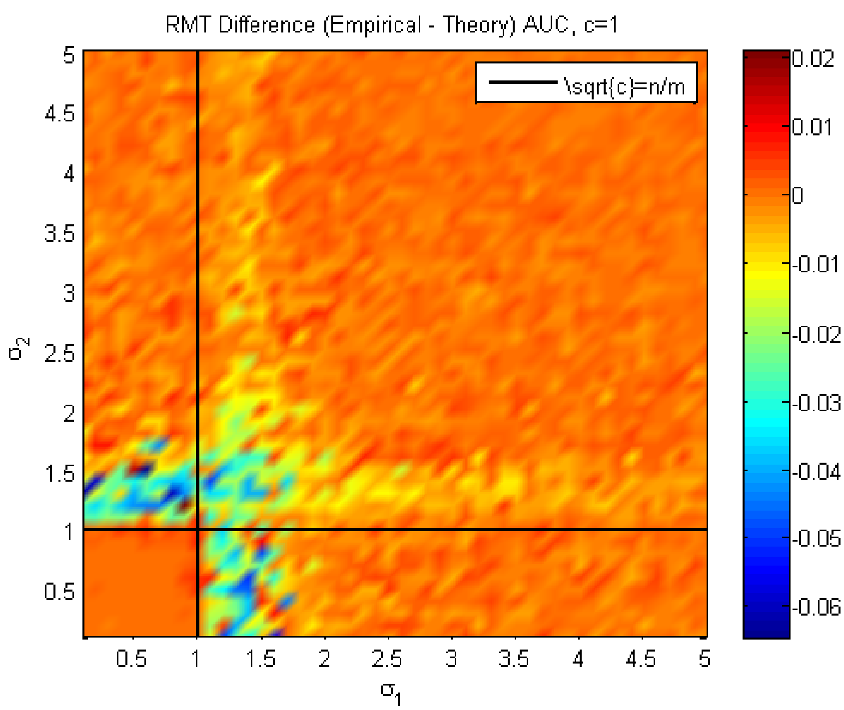

Figure 10: Difference between the empirical and theoretical RMT classifier. c=1

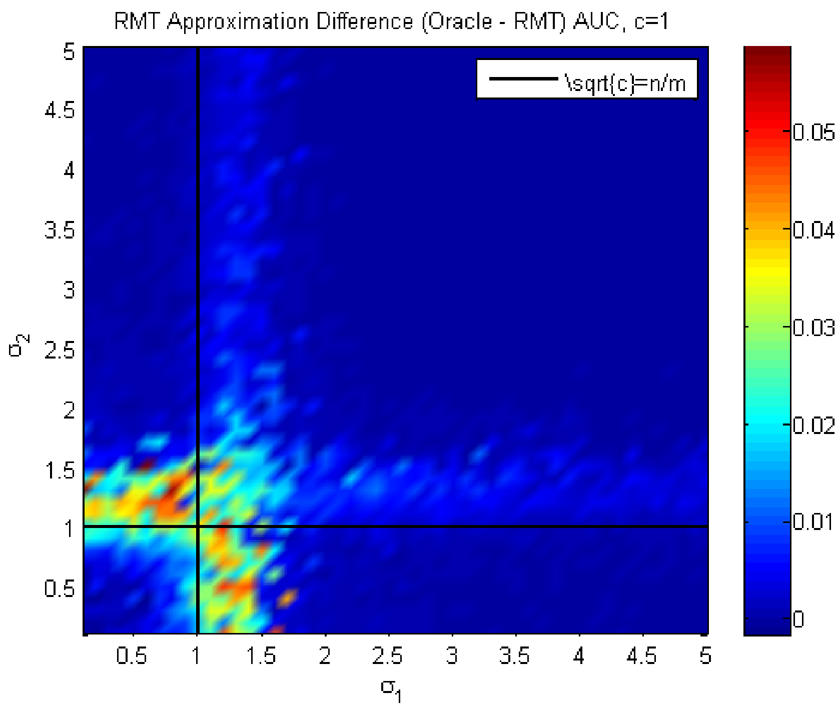

Figure 11: Difference between the empirical oracle and RMT classifier. c=1

References

[1] T. Fawcett. An introduction to roc analysis. *Pattern recognition letters*, 27(8):861–874, 2006.

[2] R.R. Nadakuditi and J.W. Silverstein. Fundamental limit of sample generalized eigenvalue based detection of signals in noise using relatively few signal-bearing and noise-only samples. *Selected Topics in Signal Processing, IEEE Journal of*, 4(3):468–480, 2010.

[3] D. Paul. Asymptotics of sample eigenstructure for a large dimensional spiked covariance model. *Statistica Sinica*, 17(4):1617, 2007.

[4] A.T.A. Wood, J.G. Booth, and R.W. Butler. Saddlepoint approximations to the cdf of some statistics with nonnormal limit distributions. *Journal of the American Statistical Association*, pages 680–686, 1993.