# PROJECT OUTLINE

JUAN DURAZO, BRENDAN HORAN, ARTHUR MITRANO

## 1. Introduction

We study how noise propagates in the Golub-Kahan iterative bidiagonalization and eventually to the final solution. The majority of this study will be to replicate many of the results from the numerical experiments done in [1]. We may add or subtract some numerical experiments later on however. Below is a brief introduction to the problem.

We study the problem

$$Ax \approx b, \quad b = b^{exact} + b^{noise} \in \mathbb{R}^n \tag{1}$$

The test problem that will be used will be **shaw**, so in equation (1) above, A is the blurring matrix and $b^{exact}$ is the blurred image that is returned from the **shaw** function. We will then generate $b^{noise}$ and add it to $b^{exact}$ to create $b$. Using this information, we apply the Golub-Kahan iterative bidiagonalization(GKIB) using MATLAB code that we found online to perform our numerical experiments. From this code we will obtain a sequence of real numbers $\alpha_j, \beta_j$ and vectors $s_j, w_j$ for $k = 1, 2, ...k$. In the below experiments we will have the SVD decompositions of $A = U\Sigma V^T$ and $L_k = P_k \Theta_k Q_k^T$, where $L_k$ is a bidiagonal matrix containing each $\alpha_j$ and $\beta_j$.

## 2. Numerical Experiments

- In the paper, they assume that the components of $s_1$ in the directions of $u_1, u_2, ...u_k$ decay faster than the associated singular values. With this assumption they show that as a consequence of the orthogonalization process, $s_2, s_3, ...s_k$ will increasingly have dominant components in the directions of the vectors$u_1, u_2, ...u_k$ respectively. Also, the high frequency noise components in these vectors will gradually increase up until some $k = k_{noise}$. After this, the vectors $s_{k_{noise}+1}$ will have comparable components in practically all subspaces generated by singular vectors corresponding to $\sigma^2_{k_{noise}+1}, \sigma^2_{k_{noise}+2}, ...$, and this will reveal the noise. We will first verify that for the **shaw** problem, the above assumption indeed holds and that the advertised consequences occur.
- When the above happens, that is the step in the GKIB algorithm where we stop the iteration. Section 4 of this paper offers ways to estimate the noise level at this point. We will repeat this experiments for several values of noise level and see how well we can estimate it using this GKIB algorithm.

- Once we have the noise estimate, we can apply the NCP idea to solve the resulting problem that the GKIB problem gives us.