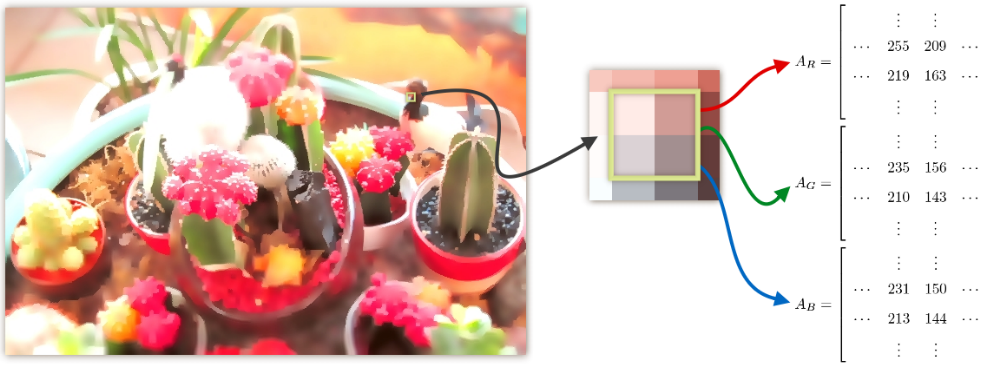
Project 2: Image Compression

1. In this exercise, we seek to compress an image while keeping most of its features intact. We will use SVD to do so. Consider the following cacti image of pixel dimensions $h \times w = 400 \times 600$:



For a 24-bit, RGB image, there are three channels—red, green, and blue—each with 8 bits (0 - 255). Thus, we construct three matrices A_R, A_G, $A_B \in \mathbb{R}^{h\times w}$, whose (i, j)-th entry is an integer between 0 and 255 denoting the intensity of that color in the (i, j)-th pixel. (From now on, we hide the subscripts R, G, and B for simplicity.) For each channel, we then find the SVD of its matrix, $A = U\Sigma V^T$, where $U \in \mathbb{R}^{h\times h}$, $\Sigma \in \mathbb{R}^{h\times w}$, and $V \in \mathbb{R}^{w\times w}$.

(a) WLOG, assume $h \leq w$, i.e. the image is a landscape shot (otherwise, turn the camera sideways). Suppose A has rank $r\,(\leq h)$. Show that $A = \sum_{i=1}^{r} \sigma_i \vec{u}_i \vec{v}_i^T$.

(b) For $1 \leq k \leq r$, partition

$$U = \left[\begin{array}{c|c} U_L & U_R \end{array}\right],\ \Sigma = \left[\begin{array}{c|c} \Sigma_{TL} & 0 \\ \hline 0 & \Sigma_{BR} \end{array}\right],\ V = \left[\begin{array}{c|c} V_L & V_R \end{array}\right],$$

so that we have $U_L \in \mathbb{R}^{h\times k}$, $\Sigma_{TL} \in \mathbb{R}^{k\times k}$, and $V_L \in \mathbb{R}^{w\times k}$.

Prove that $A_k \equiv U_L \Sigma_{TL} V_L^T$ is the best approximation to A among all matrices of rank k or less in the Frobenius-norm sense, i.e.

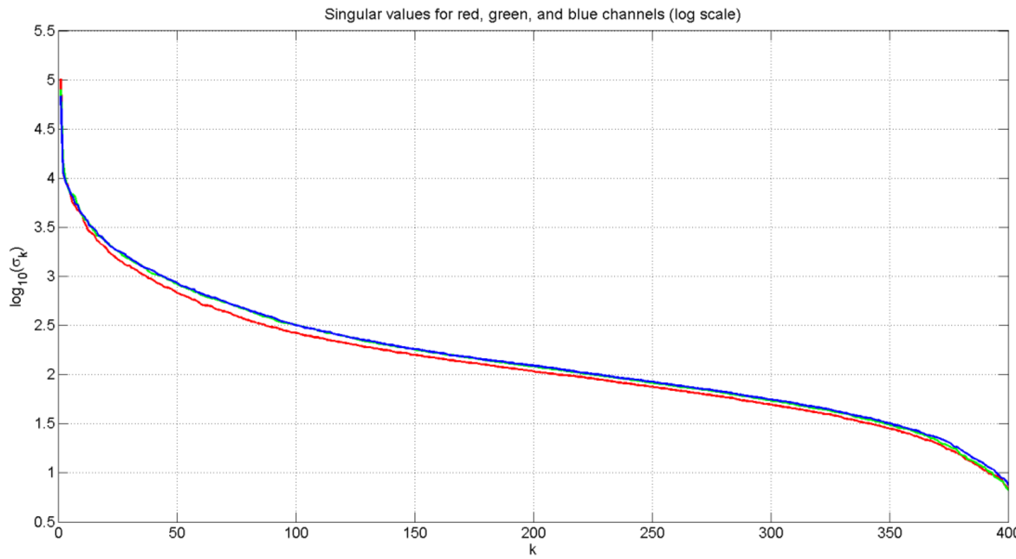
$$||A - A_k||_F = \min_{\substack{B\,\in\,\mathbb{R}^{h\times w} \\ \operatorname{rank}(B)\,\leq\,k}} ||A - B||_F = \sqrt{\sigma_{k+1}^2 + \cdots + \sigma_r^2}.$$

(c) From parts (a) and (b), we see that we can approximate each color channel matrix A with A_k, a sum of k rank-one matrices, and that A_k is the best approximation of A in the Frobenius-norm sense.

Now, A requires storing hw entries. In comparison, how many entries does $A_k = \sum_{i=1}^{k} \sigma_i \vec{u}_i \vec{v}_i^T$ require in terms of h, w, and k?

For the cacti example, evaluate the ratio in the number of entries that need to be stored for A_k to A for $k = 5, 10, 20, 40, 80$. Since this ratio shouldn't exceed 1, we have an upper bound on k. Express it in terms of h and w, and evaluate it for the cacti example.

(d) Ideally, the compression would require a small fraction in storage but still render an image of good quality. The upper bound found in part (c) is rather large and doesn't tell much what the ideal k is. Let us heuristically find k by examining the singular values of A.



The graph above shows the singular values (log scale) for each color channel for the cacti example. Based on the storage ratio and this graph, argue that $k = 80$ is a good choice. Run **project2.m** to check the fidelity of the compressed image for various k. For $k = 80$, evaluate the relative error $\dfrac{||A - A_k||_F}{||A||_F}$ for each color channel using the formula in part (b).

(e) We seek to generate an approximation $\tilde{A}_k \in \mathbb{R}^{h\times w}$ to A that is also of rank k or less, so that we can see that A_k indeed "looks better" than, say, $\tilde{A}_k$.

Follow the steps below to prove the following statement:

$$\text{"Let } A \in \mathbb{R}^{m\times p},\ B \in \mathbb{R}^{p\times n}. \text{ Then, } \operatorname{rank}(AB) \le \min\{\operatorname{rank}(A), \operatorname{rank}(B)\}."$$

(i) Show that $\operatorname{rank}(AB) \le \operatorname{rank}(B)$. [Relate $\operatorname{null}(B)$ to $\operatorname{null}(AB)$ and use the rank-nullity theorem.]

(ii) Show that $\operatorname{rank}(A^T) = \operatorname{rank}(A)$.

(iii) Use (i) and (ii) to show that $\operatorname{rank}(AB) \le \operatorname{rank}(A)$.

Thus, for each color channel, if we define $\tilde{A}_k := A_k E$, where $E \in \mathbb{R}^{w\times w}$ has a rank of at least k, then we would get

$$\operatorname{rank}(\tilde{A}_k) \le \operatorname{rank}(A_k) = k.$$

The error matrix E should be chosen so that the entries of $A_k E$ remain between 0 and 255. So let E be a bidiagonal matrix, with the diagonals all 0.5's and the superdiagonals all 0.45's. We easily see that the entries of $A_k E$ are still between 0 and 255, and that $\operatorname{rank}(E) = w \ge k$.

Run **project2.m** to compare A_k and $\tilde{A}_k$ against the original for $k = 80$.