Problem 1 (Hyperspectral Unmixing). Hyperspectral Imaging is similar to color photography, but each pixel acquires many bands of light intensity data from the spectrum, instead of just the three bands of the RGB color model.\footnote{Explanation from Wikipedia}

The input of hyperspectral unmixing contains $n \times n$ hyperspectral images. For each pixel in the image, the data contains a spectrum: intensity information on multiple wavelengths (represented as a vector in $\mathbb{R}^d$, see Figure\ref{fig: hyperspectral image}).

As we look at hyperspectral images from satellites, each pixel consists of a mixture of natural or construction materials (soil, vegetation, concrete, etc.). Different materials have different signature

Figure 1: Hyperspectral Image

\begin{figure}
\centering
\includegraphics[width=\textwidth]{hyperspectral_image}
\caption{Hyperspectral Image}
\end{figure}
spectra (which are vectors in \( \mathbb{R}^d \)). The spectrum of a pixel is the convex combination of the spectra of its constituting materials.

The goal of hyperspectral unmixing is to find the signature spectra of the materials, and the constituting materials for each pixel.

(a) [5 points] Explain why the hyperspectral unmixing problem can be viewed as an NMF problem \( M = AW \) (e.g. topic modeling can be viewed as NMF, because \( M \) is the word-by-document matrix, \( A \) is the word-by-topic matrix and \( W \) is the topic-by-word matrix). If there are \( k \) different materials, what are the dimensions of \( M, A, W \)?

(b) [5 points] Translate separability assumption into the context of hyperspectral unmixing.
(Hint: There are two possible translations because you can take the transpose of the matrices. However, the vectors for the signature spectra are strictly positive: all their entries are greater than 0.)

Problem 2 (Faster separable-NMF Algorithm). Let \( v_1, v_2, \ldots, v_k \in \mathbb{R}^d \) be points in \( d \)-dimensional space. Given points \( u_1, u_2, \ldots, u_n \) in the convex hull \( \text{conv}\{v_1, \ldots, v_k\} \), assume for each \( v_i \) there is a (unknown) \( r_i \in \{1, \ldots, n\} \) such that \( u_{r_i} = v_i \). Separable NMF is equivalent to finding the vertices \( v_i \)’s. For normalization assume all the \( v_i \)’s have nonnegative coordinates and \( |v_i|_1 = 1 \).

For a set of points \( \{v_1, \ldots, v_k\} \), define the affine hull \( \text{aff}\{v_1, \ldots, v_k\} \) to be the set \( \{ u : u = \sum_{i=1}^k c_i v_i, \sum_{i=1}^k c_i = 1 \} \). For example, the affine hull of two points is the line that passes through the two points. The distance between a point \( u \) and an affine hull \( \text{aff}(S) \) is defined to be the minimum \( \ell_2 \)-distance between \( u \) and any point in \( \text{aff}(S) \):

\[
\text{dist}(u, \text{aff}(S)) = \min_{v \in \text{aff}(S)} \|u - v\|_2.
\]

The following algorithm can be used to find the vertices efficiently:

```
Find \( p \) in \( \{1, \ldots, n\} \), such that \( \|u_p\|_2 \) is the largest among all \( u \)’s.
Let \( S = \{u_p\} \).
for \( i = 1 \) TO \( k - 1 \) do
    Find \( q \) in \( \{1, \ldots, n\} \), such that \( \text{dist}(u_q, \text{aff}(S)) \) is the largest among all \( u \)’s.
    Let \( S = S \cup \{u_q\} \).
end for
return set \( S \)
```

(a) [10 points] If \( \{v_1, \ldots, v_k\} \) form a simplex (that is, for any \( i \in \{1, \ldots, k\} \), \( v_i \notin \text{aff}(\{v_1, \ldots, v_k\} \setminus \{v_i\}) \), no \( v_i \) is in the affine hull of others), prove the set \( S \) returned by the algorithm contains all the vertices \( (v_i \in S \text{ for all } i = 1, \ldots, k) \).

Hint: The distance to an affine hull satisfy the strong convexity condition, for any \( u, v \in \mathbb{R}^d \), and any \( \alpha \in (0, 1) \), if \( u \notin \text{aff}(\{v\} \cup S) \) then

\[
\text{dist}(\alpha u + (1 - \alpha)v, \text{aff}(S)) < \alpha \text{dist}(u, \text{aff}(S)) + (1 - \alpha)\text{dist}(v, \text{aff}(S)).
\]

We also want to show that the algorithm is robust to noise. Recall a set \( \{v_1, \ldots, v_k\} \) is \( \alpha \)-\( \ell_1 \)-robust if for all \( i = 1, \ldots, k \)

\[
\text{dist}_{\ell_1}(v_i, \text{conv}(\{v_1, \ldots, v_k\} \setminus v_i)) \geq \alpha,
\]

We also want to show that the algorithm is robust to noise. Recall a set \( \{v_1, \ldots, v_k\} \) is \( \alpha \)-\( \ell_1 \)-robust if for all \( i = 1, \ldots, k \)
where $\text{dist}_{\ell_1}(u, \text{conv}(S))$ is defined as

$$\text{dist}_{\ell_1}(u, \text{conv}(S)) := \min_{v \in \text{conv}(S)} |u - v|_1.$$  

We define a set $\{v_1, \ldots, v_k\}$ to be $\alpha-\ell_2$-robust if

$$\text{dist}(v_i, \text{aff}(\{v_1, \ldots, v_k\} \setminus v_i)) \geq \alpha.$$  

(b) [5 points] Show that if a simplex is $\alpha-\ell_2$-robust, then it is also $\alpha-\ell_1$-robust. (Hint: For any vector $|v|_1/\sqrt{d} \leq \|v\|_2 \leq |v|_1$.

(c) [5 points] Show that the other direction is not true: in particular, there is a $0.1-\ell_1$-robust set that is not $\epsilon-\ell_2$-robust for any $\epsilon > 0$. (Hint: Affine hull is larger than convex hull.)

(d) [BONUS 10 points] Suppose $\{v_1, \ldots, v_k\}$ is $\alpha-\ell_2$-robust, the set $S$ contains a subset of vertices $S \subset \{v_1, \ldots, v_k\}$. Let $\hat{u}_i = u_i + \delta_i$ where $\|\delta_i\|_2 \leq \epsilon$ is a noise vector. Let $\hat{u}_q$ be the point that has largest $\ell_2$ distance to $\text{aff}(S)$ among all $\hat{u}$’s, show that there exists a $v_j \notin S$ such that $\|\hat{u}_q - v_j\|_2 \leq O(\epsilon/\alpha^2).$

(Hint: First project all the points to the orthogonal subspace of $\text{aff}(S)$, then expand $\|u\|_2 = \|\sum_{j=1}^k c_j v_j\|_2^2$ into convex combination of $k^2$ inner-products. Show that the cross-terms $\langle v_i, v_j \rangle$ are small, so $c_i c_j$ must also be small, and one of $c_j$ must be close to 1.)