Worksheet: dimensionality reduction

Joaquin Rapela

February 12, 2025

This worksheet is a modification of a worksheet from the 2023 edition of the NEUO019 Foundations of Neuroinfomatics course, taught by Prof. Kenneth Harris at UCL.

We are going to use the singular value decomposition (SVD) to uncover hidden structure in the firing rate of a population of neurons in a mouse motor cortex and striatum. We will use data from the International Brain Laboratory (IBL). To learn more about the IBL please go to `https://viz.internationalbrainlab.org`.

1. First, install package needed for this worksheet by typing the following in the command line. I recommend using conda environments. If you are doing so, before running the next command, activate your conda environment.

```
cd <repository_directory>/worksheets/04_dimensionalityReduction
git pull
pip install -r requirements.txt
```

Next use the script doEx1.py to plot the firing rates of a population of neurons, as in Fig. 1.

In this script we first import libraries and open a connection IBL public data server:

```
import numpy as np
import matplotlib.pyplot as plt
import one.api
import scipy
```

Next, we load information about the spikes and trials for one experiment. Each experiment has a unique experiment ID, which is a long string identifying the experiment. This particular one is an experiment made at New York University, where they recorded from the motor cortex and striatum. We load two data objects: dictionaries containing information about the spikes, and about the trials.

```
eID = 'ebe2efe3-e8a1-451a-8947-76ef42427cc9'
spikes = one.load_object(eID, 'spikes', 'alf/probe00/pykilosort')
trials = one.load_object(eID, 'trials')
```

We then group the spikes together into an array, using the function `np.histogramdd`. Also we Z-score the activity of each cell (set it to have mean 0 and std 1)

```
eID = 'ebe2efe3-e8a1-451a-8947-76ef42427cc9'
bin_size = 1
spike_time_bin = np.floor(spikes.times/bin_size).astype(int)
activity_array, _ = np.histogramdd((spike_time_bin,spikes.clusters), bins=( spike_time_bin.max(), spikes.cluste
activity_array = activity_array.T % after transposing the shape of activity_array is n_neurons x n_times
activity_arrayZ = stats.zscore(activity_array)
```

Finally we plot a pseudocolor image of the array, but just sorted in whatever order it came. The functions `getHeatmap` and `getHovertext` are provided in module utils.py.

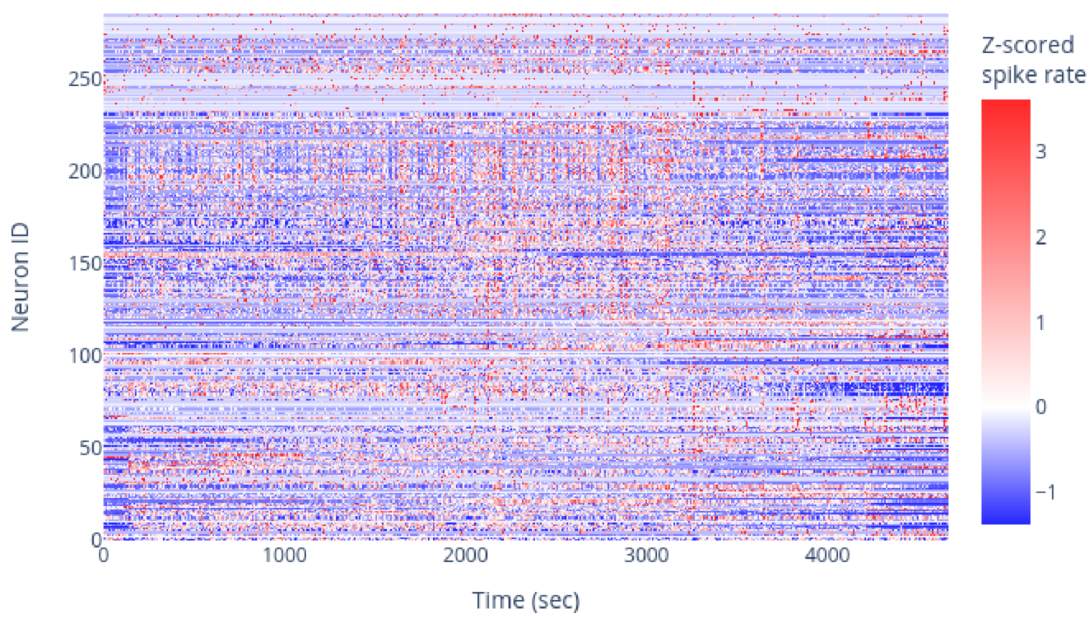

Figure 1: Z-scores of binned spikes times of all unsorted neurons.

```
eID = 'ebe2efe3-e8a1-451a-8947-76ef42427cc9'
zmin, zmax = np.percentile(activity_arrayZ, q=(1.0, 99.0))
hovertext = utils.getHovertext(
    times=times, clusters_ids=clusters_ids, z=activity_arrayZ.T,
    channels_for_clusters=clusters.channels,
    regions_for_channels=els[probe_id]["acronym"])
fig = utils.getHeatmap(xs=times, ys=clusters_ids, zs=activity_arrayZ.T,
                        hovertext=hovertext, zmin=zmin, zmax=zmax,
                        x_label=x_label, y_label=y_label,
                        colorbar_title=colorbar_title)
fig.write_image(fig_filename_pattern.format(bin_size, "original",  "png"))
fig.write_html(fig_filename_pattern.format(bin_size, "original",  "html"))
fig.show()
```

2. We want to use the SVD to find the temporal activity pattern that captures most of the variance in the population (i.e., the population firing pattern), and sort the neurons in Fig. 2 according to the similarity of their firing rates to the population firing pattern.

   Take a matrix $M \in \mathbb{R}^{m \times n}$, we can use the SVD to decompose M into as a summation of $r = \min(m, n)$ rank one matrices (a rank one matrix M is one that is the product of a column vector $\mathbf{u}$ times a row vector $\mathbf{v}^\intercal$; i.e., $M = \mathbf{u}\mathbf{v}^\intercal$). Eq. 1 gives the decomposition of M into a sum of r rank one matrices, and Eq. 2 gives the best rank one approximaton to M.

$$M = USV = \sum_{i=0}^{r-1} s_i U[:, i] V[:, i]^\intercal \tag{1}$$

$$\simeq s_0 U[:, 0] V[:, 0]^\intercal \tag{2}$$

   For the z-scored firing rates we are analyzing in this worksheet, the vector $V[:, 0]$ gives the population firing pattern. When $U[0, i]$ is positive (negative) and has a large absolute value, the z-scored firing rate of the ith neuron will be hightly correlated (anticorrelated) with the population firing pattern.

   Plot the z-scored firing rates, as in Fig. 2, but sorting the neurons according to the correlation of their firing pattern with population firing pattern. Add to this plot a vertical line at the time of the last response of the subject. You may want to complete the script doEx2.py. Fig. 2 plots the sorted spikes rates that I obtained.

3. Fig. 2 suggest that the response of the neurons in the population is related to the subjects response. To verify this, plot the population firing pattern and add to this plot vertical lines indicating subject response times. You may want to complete the script doEx3.py. Figure 3 shows the population firing pattern that I obtained.

4. (optional) As proved in the Eckart-Young-Mirsky theorem (Theorem 1), the truncated rank ν singular value decomposition of matrix M (Eq. 7), is the best rank ν approximation to M, among all possible rank ν matrices. In addition, the Frobenius error of this best approximation can be computed from the singular values of the SVD decomposition of matrix M, without having to build assemble the best approximation.

   For $\nu = 1, \ldots, 5$, verify the above claim by assembling the truncated SVD, M_ν, of the matrix, M, in Fig.2, computing the Frobenious norm of the difference betweeen M and M_ν, and comparing this norm with the the sum of the first ν singular values of matrix M. Plot M_ν and write in the title the Frobenius error (i.e., empirical error) and sum of the squared

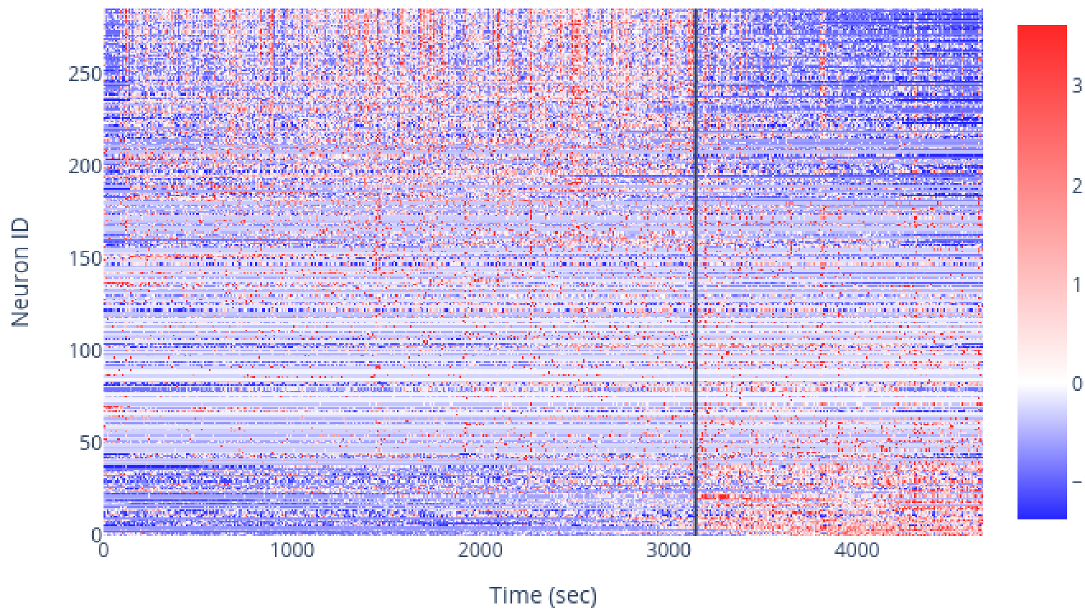

Figure 2: Z-scores of binned spikes times of all sorted neurons.

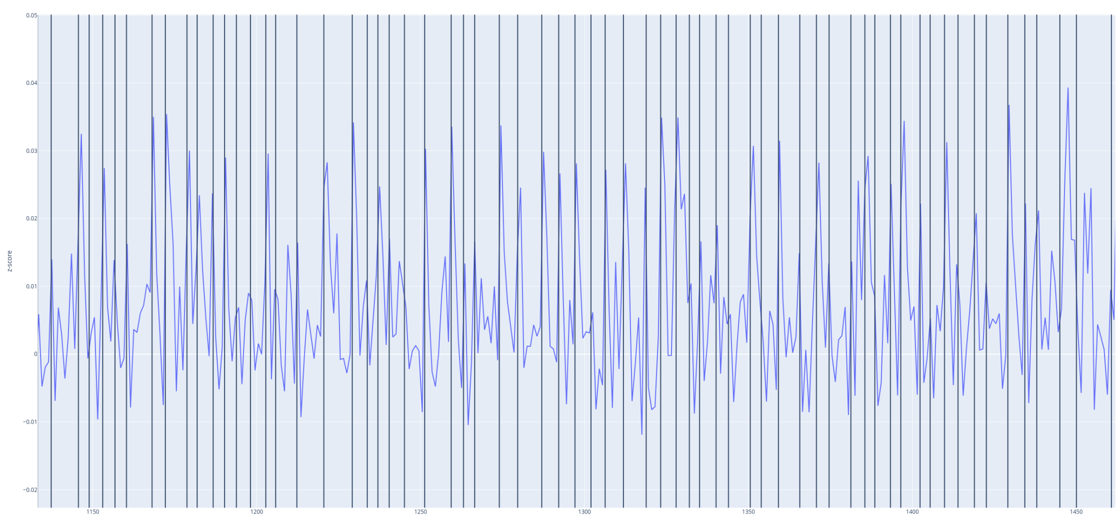

Figure 3: Segment of the population firing pattern (blue trace) and response times of the mice (vertical lines).

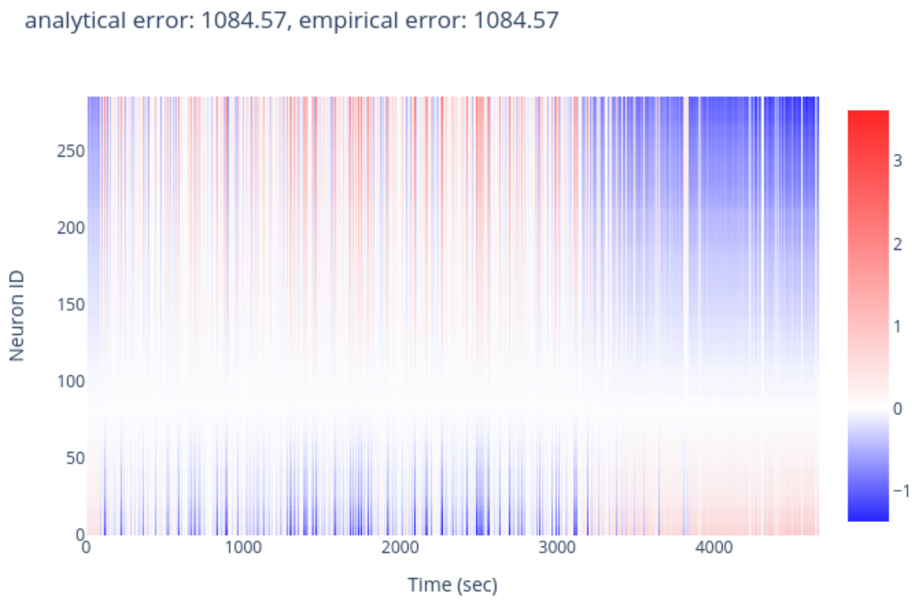

Figure 4: Low-rank approximation of the image in Figure 2 using a truncated SVD or rank ı. The title reports the empirical and analytical errors of the reconstruction. The empirical error is the Frobenius norm of the difference between the low-rank approximation and the image in Figure 2. The analytical error is computed from the singular values of the image in Figure 2 using Eq. 8.

first ν singular values (i.e., the analytical error), as shown in Fig. 4 for $\nu = 1$. You may want to complete the script doEx4.py. The truncated SVDs for ranks 1, 2 and 5, with their empirical and analyical approximation errors are shown inf Figs. 4, 5, and 6, respectively.

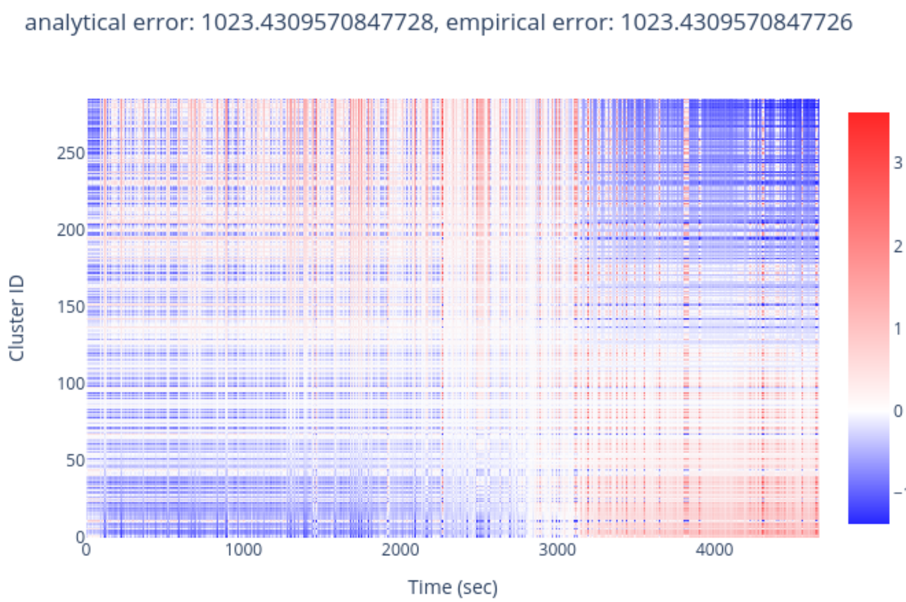

Figure 5: Low-rank approximation of the image in Figure 2 using a truncated SVD of rank 2. Same format as that in Figure 4.

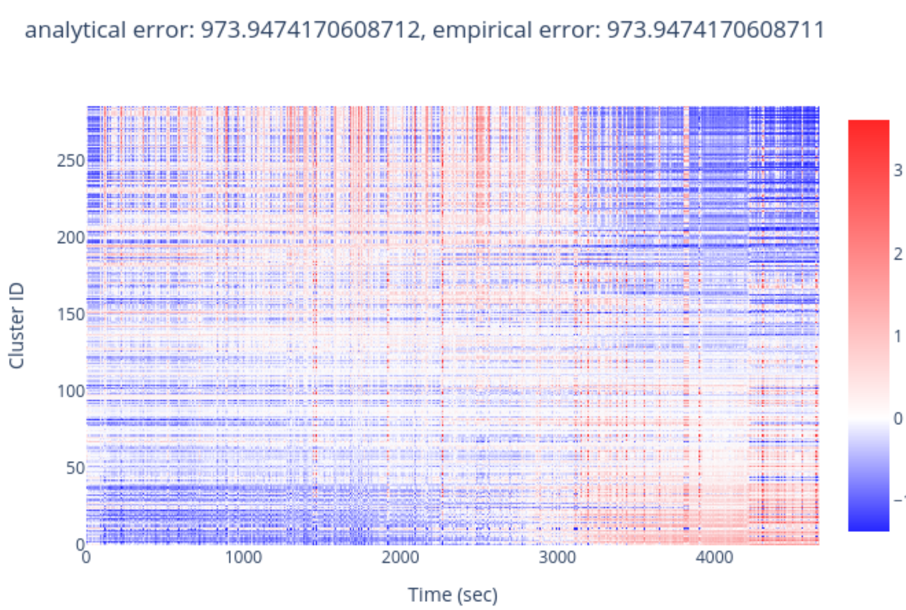

Figure 6: Low-rank approximation of the image in Figure 2 using a truncated SVD of rank 5. Same format as that in Figure 4.

Appendix A Notes on the SVD

Definition 1 (The SVD)**.** *Given $M \in \mathbb{C}^{m\times n}$, a singular value decomposition (SVD) of M is a factorisation:*

$$M = USV^*$$

where

$$\begin{aligned} U &\in \mathbb{C}^{m\times m} && \text{is unitary,} \\ V &\in \mathbb{C}^{n\times n} && \text{is unitary,} \\ S &\in \mathbb{C}^{m\times n} && \text{is diagonal.} \end{aligned}$$

In addition, it is assumed that the diagonal entries s_k of S are nonnegative and in nonincreasing order; that is, $s_1 \geq s_2 \geq \ldots \geq s_p \geq 0$, where $p = \min(m, n)$.

Definition 2 (Rank of a matrix)**.** *The column rank of a matrix is the dimension of the space spanned by its columns. Similarly, the row rank of a matrix is the dimension of the space spanned by its rows. The column rank of a matrix is always equal to its row rank. This is a corollary of the SVD. So we refer to this number simply as the rank of a matrix.*

The rank of a matrix can be interpreted as a measure of the complexity of the matrix. Matrices with lower rank are simpler than those with larger rank.

The SVD decomposes a matrix as a sum of rank-one (i.e., very simple) matrices.

$$M = \sum_{k=1}^{r} s_k \mathbf{u}_k \mathbf{v}_k^*$$

There are multiple other decompositions as sums of rank-one matrices. If $M \in \mathbb{C}^{m\times n}$, then it can be decomposed as a sum of m rank-one matrices given by its rows (i.e., $M = \sum_{i=1}^{m} \mathbf{e}_i \mathbf{m}_{i,\cdot}^*$, where $\mathbf{e}_i$ is the m-dimensional canonical unit vector, and $\mathbf{m}_{i,\cdot}$ is the ith row of M), or as a sum of n rank-one matrices given by its columns (i.e., $M = \sum_{j=1}^{n} \mathbf{m}_{\cdot,j} \mathbf{e}_j^*$, where $\mathbf{e}_j$ is the n-dimensional canonical unit vector, and $\mathbf{m}_{\cdot,j}$ is the jth column of M), or a sum of mn rank-one matrices each containing only one non-zero element (i.e., $M = \sum_{i=1}^{m} \sum_{j=1}^{n} m_{ij} E_{ij}$, where E_{ij} is the matrix with all entries equal to zero, except the ij entry that is one, and m_{ij} is the entry of M at position ij).

A unique characteristic of the SVD compared to these other decompositions is that, if the rank of a matrix is r, then its SVD yields optimal approximations of lower rank ν, for $\nu = 1, \ldots, r$, as shown by Theorem 1.

Definition 3 (Frobenius norm)**.** *The Frobenius norm of matrix $M \in \mathbb{C}^{m\times n}$ is*

$$\|M\|_F = \left(\sum_{i=1}^{m} \sum_{j=1}^{n} m_{ij}^2 \right)^{1/2}$$

Note that

$$\|M\|_F = \sqrt{tr(M^*M)} = \sqrt{tr(MM^*)} \tag{3}$$

Lemma 1 (Orthogonal matrices preserve the Frobenius norm)**.** *Let $M \in \mathbb{C}^{m\times n}$ and let $P \in \mathbb{C}^{m\times m}$ and $Q \in \mathbb{C}^{n\times n}$ be orthogonal matrices. Then*

$$\|PMQ\|_F = \|M\|_F$$

Proof.

$$\|PMQ\|_F = \sqrt{tr((PMQ)(PMQ)^*)} = \sqrt{tr(PMQQ^*M^*P^*)} = \sqrt{tr(PMM^*P^*)} \tag{4}$$

$$= \sqrt{tr(P^*PMM^*)} = \sqrt{tr(MM^*)} = \|M\|_F \tag{5}$$

Notes:

1. The first equality in Eq. 4 follows Eq. 3.
2. The second equality in Eq. 4 uses the fact that $(AB)^* = B^*A^*$.
3. The third equality in Eq. 4 holds because Q is orthogonal (i.e., $QQ^* = I$).
4. The first equality in Eq. 5 uses the cyclic property of the trace (i.e., tr(ABC)=tr(CAB)).
5. The first equality in Eq. 5 holds by the orthogonality of P.
6. The last equality in Eq. 5 again applies Eq. 3.

□

A direct consequence of Lemma 1 is that the Frobenius norm of any matrix $M = USV^*$ is

$$\|M\|_F = \|USV^*\|_F = \|S\|_F = \sqrt{\sum_{k=1}^{r} s_k^2}$$

Another consequence of Lemma 1 is the error in approximating a matrix M of rank r with its truncated SVD of rank ν (i.e., $M_\nu = \sum_{k=1}^{\nu} s_k\mathbf{u}_k\mathbf{v}_k^*$) is

$$\|M - M_\nu\|_F = \|\sum_{k=1}^{r} s_k\mathbf{u}_k\mathbf{v}_k^* - \sum_{k=1}^{\nu} s_k\mathbf{u}_k\mathbf{v}_k^*\|_F = \|\sum_{k=\nu+1}^{r} s_k\mathbf{u}_k\mathbf{v}_k^*\|_F = \sqrt{\sum_{k=\nu+1}^{r} s_k^2} \tag{6}$$

Theorem 1 (Eckart-Young-Mirsky)**.** *Let $M \in \mathbb{C}^{m\times n}$ be of rank r with singular value decomposition $M = \sum_{k=1}^{r} s_k\mathbf{u}_k\mathbf{v}_k^*$. For any ν with $0 \le \nu \le r$, define*

$$M_\nu = \sum_{k=1}^{\nu} s_k \mathbf{u}_k \mathbf{v}_k^* \tag{7}$$

Then

$$\|M - M_\nu\|_F = \inf_{\substack{\tilde{M}\in\mathbb{C}^{m\times n} \\ rank(\tilde{M})\le\nu}} \|M - \tilde{M}\|_F = \sqrt{\sum_{k=\nu+1}^{r} s_k^2} \tag{8}$$

Proof. We use the Weyl's inequality that relates the singular values of a sum of two matrices to the singular values of each of these matrices. Precisely, if $X, Y \in \mathbb{C}^{m\times n}$ and $s_i(X)$ is the ith singular value of X, then

$$s_{i+j-1}(X+Y) \le s_i(X) + s_j(Y) \tag{9}$$

Let $\tilde{M}$ be a matrix of rank at most ν. Applying Eq. 9 to $X = M - \tilde{M}$, $Y = \tilde{M}$ and $j - 1 = \nu$ we obtain

$$s_{i+\nu}(M) \le s_i(M - \tilde{M}) + s_{\nu+1}(\tilde{M}) = s_i(M - \tilde{M}) \tag{10}$$

The last equality in Eq. 10 holds because $\tilde{M}$ has rank less or equal to ν, and therefore its $\nu+1$ singular value is zero.

$$\|M - M_\nu\|_F^2 = \sum_{j=\nu+1}^{r} s_j^2(M) = \sum_{i=1}^{r-\nu} s_{i+\nu}^2(M) \le \sum_{i=1}^{r-\nu} s_i^2(M-\tilde{M}) \le \sum_{i=1}^{\min(m,n)} s_i^2(M-\tilde{M}) \tag{11}$$

$$= \|M - \tilde{M}\|_F^2 \tag{12}$$

Notes:

1. The first equality in Eq. 11 holds by Eq. 6.
2. The second equality in Eq. 11 used the change of variables $i = j - \nu$.
3. The first inequality in Eq. 11 used Eq. 10
4. The last inequality in Eq. 11 is true because $r - \nu \le \min(m, n)$ and adding squared singular values to the sum in the left hand side only increases this sum.
5. The equality in Eq. 12 again holds by Eq. 8 and by the fact that singular values of index larger than the rank of a matrix are zero.

The last equality in Eq. 8 follows from Eq. 6.

□