Machine Learning HWS21
Assignment 3: Singular Value Decomposition

Timur Michael Carstensen - 1722194

21.11.2021

Contents

1 Intuition on SVD

1.1 a)

Matrix M_1 is of rank 1 and the components of its SVD are: $\mathbf{U^T} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \end{bmatrix}$, $\mathbf{\Sigma} = \begin{bmatrix} 1 \end{bmatrix}$, $\mathbf{V} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \end{bmatrix}$.

Matrix M_2 is of rank 1 and the components of its SVD are: $\mathbf{U^T} = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 \end{bmatrix}$, $\mathbf{\Sigma} = \begin{bmatrix} 1 \end{bmatrix}$, $\mathbf{V} = \begin{bmatrix} 0 & 2 & 1 & 2 & 0 \end{bmatrix}$.

Matrix M_3 is of rank 1 and the components of its SVD are: $\mathbf{U^T} = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 \end{bmatrix}$, $\mathbf{\Sigma} = \begin{bmatrix} 1 \end{bmatrix}$, $\mathbf{V} = \begin{bmatrix} 0 & 1 & 1 & 1 \end{bmatrix}$.

Matrix M_4 is of rank 2 and the components of its SVD are: $\mathbf{U^T} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}$,

$\mathbf{\Sigma} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $\mathbf{V} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}$.

Matrix M_5 is of rank 3 and the components of its SVD are: $\mathbf{U^T} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix}$,

$\mathbf{\Sigma} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$, $\mathbf{V} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}$.

Matrix M_6 is of rank 2 and the components of its SVD are: $\mathbf{U^T} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \end{bmatrix}$,

$\mathbf{\Sigma} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $\mathbf{V} = \begin{bmatrix} 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$.

1.2 b)

cf. code for the computation.

No, I was not correct. What I computed were matrix decompositions, not SVDs. Hence, I failed on task 1a.

1.3 c)

cf. code for the rest of the rank 1 approximations. In Figures 1 and 4, one can see the rank 1 approximation of M_1 and the rank 1 to 3 approximations of M_5. It is intuitive that the rank 1 approximation of M_1 is exact while the same is not true for M_5. As the former is of rank 1, the size-1 truncated SVD is completly sufficient to fully reconstruct the matrix. However, as M_5 is of rank 3, a rank 1 approximation is not sufficient. Only the rank 3 approximation is able to fully reconstruct M_5.

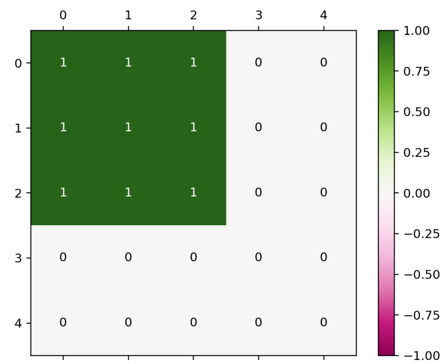


Figure 1: Rank 1 approximation of M_1

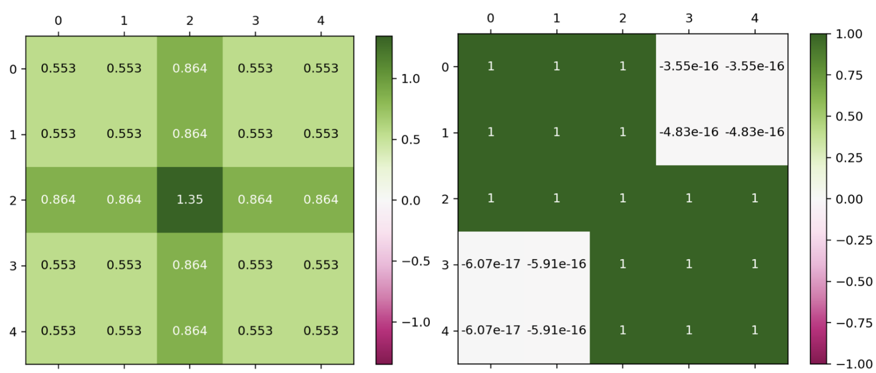


Figure 2: Rank 1 (left) and rank 3 (right) approximation of M_5

1.4 d)

As mentioned in subsection 1.1, M_6 is of rank 2 and therefore has 2 non-zero singular values. NumPy technically reports 5 non-zero singular values. However, the last three are very close to zero (apart from a rounding error). This is because we are taking roots, which leads to numerical stability issues.

2 The SVD on Weather Data

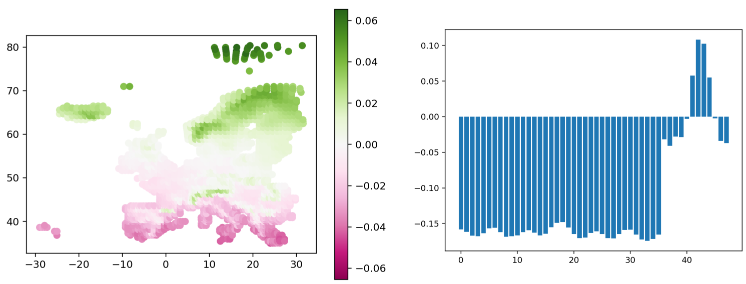
2.1 a)

Given that the climate data set contains columns (i.e. features) with differing data scales and that the SVD is sensitive in that regard, it is necessary to perform normalisation. We asume that all features are equally important.

2.2 b)

cf. code. The rank is 48.

2.3 c)

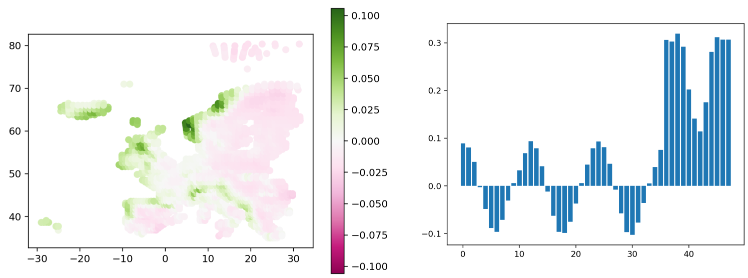


(a) (b)

Figure 3: (a) first left singular vector, (b) first right singular vector

As we can see in Figure 3, the first right singular vector (b) loads highly on temperature features (1-36) and only lightly on rain features. The first left singular vectors shows large values for data points up north and small ones for those in the south. Hence, we can interpret it as displaying the inverse temperatures. That is, data points with small values tend to have higher temperatures whereas those with large ones have rather low temperatures. This applies to minimum, maximum and average temperatures.

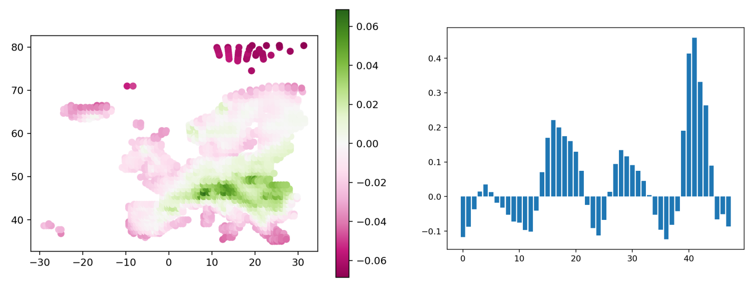
In Figure 4 we can see that coastal and mountainous regions generally have higher values in the 2nd left singular vector. The 2nd left singular vector loads highly (positively) on rain features and temperatures features are inverse to the seasons. That is, positive values in winter/autumn and negative ones in spring/summer. This can be interpreted as average rainfall. In (a), both coastal and mountainous regions have higher values. The former experiences more precipitation due to high evaporation and the former due



(a) (b)

Figure 4: (a) second left singular vector, (b) second right singular vector

to orographic lift. This is in line with the inverse temperatures, as one would expect more rain in autmn & winter than in spring & summer.

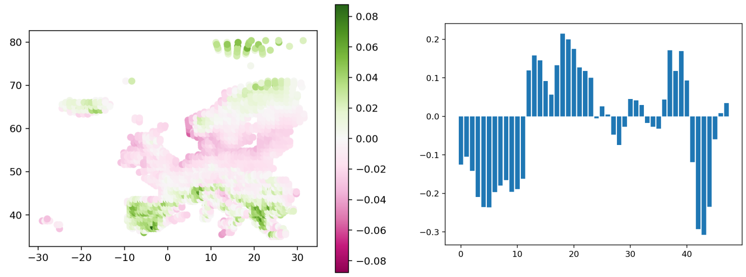


(a) (b)

Figure 5: (a) third left singular vector, (b) third right singular vector

In Figure 5 one can see that data points with high loadings in the 3rd left singular vector experience very low minimum and very high maximum temperatures as per the corresponding right singular vector. Also, there are significant differences regarding precipitation between summer and winter months. Hence, the 3rd left singular vector can be interpreted as the variance in temperatures across the year (similar to the example from the ML lecture on the SVD). Examples with high loadings experience large variation and those with lower values have a more steadfast weather over the year.

Figure 6 can be interpreted as the elevation when compared to a topological map of Europe. That is, data points with high values in the fourth left singular vector correspond to areas of high elevation whereas those with low values to ones closer to sea level. For example, the alps and the pyrenees have high values in (a). Together with (b), we can interpret this as areas with high elevation experiencing, on average, low minimum, high maximum tem-



(a) (b)

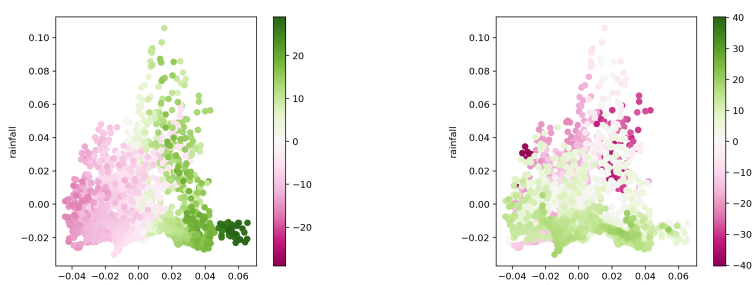
Figure 6: (a) fourth left singular vector, (b) fourth right singular vector

peratures (i.e continental climate) and alot of rain in the summer months.

The 5th left singular vector is not interpretable anymore (and hence not shown here), which is not a surprise as we will find out later when deciding on the "best" size-k truncated SVD.

2.4 d)

As we interpreted the first and second left singular vectors as (inverse) average temperature and rainfall, we can observe a large variation of temperatures along the x-axis in Figure 7 (a) moving from north to south (green to red). The dataset is not as well separated for rainfall when doing the same. An interesting observation is that data points far up north, experience very low temperatures and little rainfall. This corresponds to polar weather which is, typically, cold and dry.



(a) (b)

Figure 7: (a) U_1 and U_2 north to south, (b) U_1 and U_2 east to west

When moving from east to west in Figure 7 (b), this time the dataset is not well separated for temperatures nor for rainfall, though the separation is still somewhat apparent for the latter. This makes sense as we would expect there to be more rainfall along the coastlines (i.e. in the west), than in the east.

For the other columns of U, I could not find a meaningful interpretation, which is why the other plots (cf. Jupyter Notebook) are not included here.

2.5 e)

(i) According to the Guttman-Kaiser criterion, the SVD size-k should be 37.

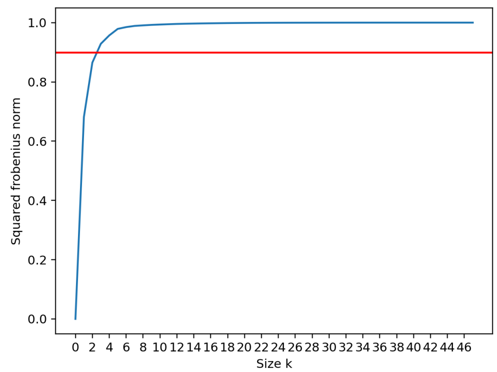


Figure 8: 90% of squared frobenius norm

(ii) As can be seen in Figure 8, one reaches 90% of the squared frobenius norm at $k = 3$.

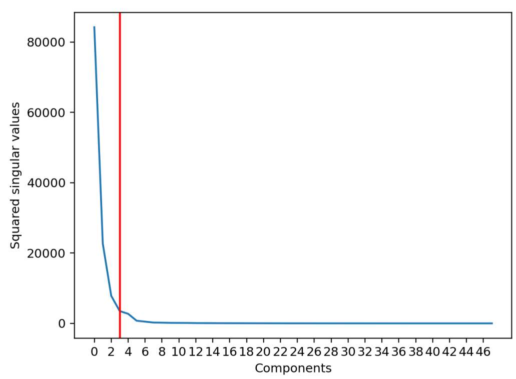


Figure 9: Scree plot

(iii) Using Catell's Scree test, the optimal size is $k = 4$.

(iv) According to the entropy-based method k is equal to 1.

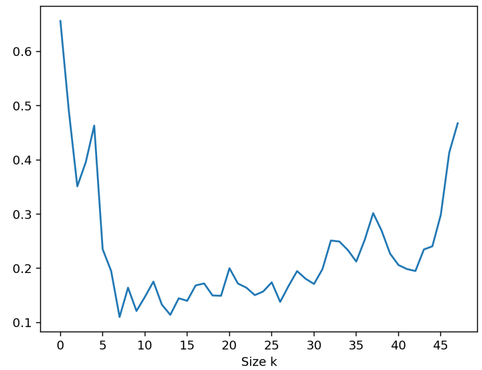


Figure 10: Random flip of signs

(v) As can be seen in Figure 10, the optimal k is 8.

As discussed in the lecture, all of the above methods have their issues. In the case of Guttman-Kaiser, the optimal k heavily depends on the scaling of the data. The 90% squared frobenius norm ”test” (i.e. captured energy) has the same problem. However, in this case it yields better results than Guttman-Kaiser which are also almost in line with the next (Catell’s Scree test). The entropy based method yields the smallest result with $k = 1$, which also seems a little too low. Finally, the random flip of signs test, when picking the minimum, yields a $k = 8$. Here, it is not really clear which k is the right one to pick.

All in all, none of these tests are perfect but in combination they do hint at a lower number for size k. In this particular case, I would choose captured energy. As we will see in subsection 3.3, $k = 3$ seems to be more or less optimal as the information gained beyond that is rather insignificant.

2.6 f)

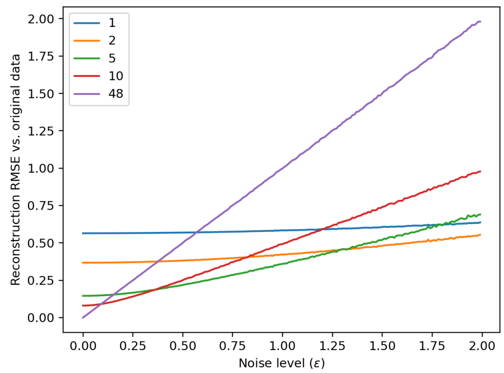


Figure 11: RMSE of the reconstruction truncated SVDs for varying levels of noise

To start with the most obvious, given that we use the full SVD and have no noise ($\epsilon = 0$), we obviously have a RMSE of zero. Something that is at first less obvious is that adding noise to the full reconstruction results in a higher RMSE than using a truncated SVD such as $k \in \{1, 2, 5, 10\}$. However, when recalling that one of the uses of the SVD is to denoise data, this makes a whole lot more sense. We can see that for the reconstruction with $k = 1$, the RMSE is constant for any noise level (ϵ). That is, we have "filtered" out the added noise by just using the first (and hence largest) singular value. When looking at the $k = 2$ reconstruction, we can already see that we are getting a worse RMSE for $\epsilon > 0.5$. In the end (i.e. at $\epsilon = 2$), $k = 2$ is the best reconstruction as measured by RMSE.

3 SVD and Clustering

3.1 a)

A very intuitive interpretation of the clustering are the different climate zones within the European Union. For example, boreal northern (green) and nemoral (blue) climate in the northern and continental (light blue) in the eastern part of the map (cf. Figure 12).

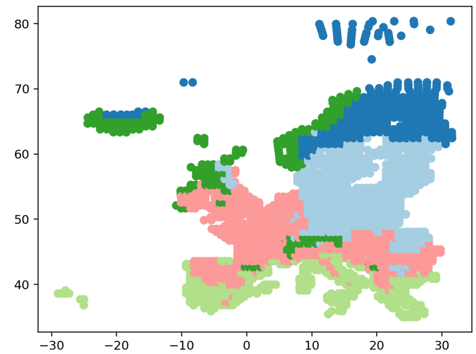


Figure 12: Clusters based on k-means

3.2 b)

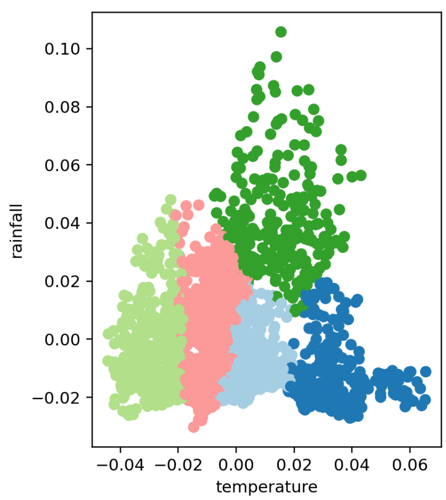


Figure 13: Plot of 1st and 2nd left singular vector colour coded by cluster

The clusters are well separated in the plot (cf. Figure 13). Generally, the clusters have relatively smooth edges and do not overlap. One exception is the cluster representing northern european coastal regions (green). It shows more variation in rainfall (as per the interpretation in subsection 2.3). Therefore, one could argue that the topmost data points in that cluster could represent their own (cluster). Same goes for the cluster that represents central Europe which has a few data points all the way toward the right on the x-axis.

3.3 c)

To compute the PCA scores of our data matrix $\mathbf{X}$ for the first k principal components, we can multiply the first k left singular vectors and singular values from the SVD of $\mathbf{X}$. This works, because we centered (normalised) our data before calculating the SVD. For $k = 1$, there are significant differences for the clustering, which, again, is intuitive when compared with our earlier findings. Similarly, for $k = 2$ and $k = 3$, it is not possible to spot any differences, especially in figures of this size. By using the size-k truncated SVD to calculate our PCA-scores, we perform dimensionality reduction and battle the curse of dimensionality. By using the principal components we are looking at the largest directions of variation of our data. That is, we reduce our data to $k \in \{1, 2, 3\}$ dimensions and perform k-means clustering on it. Combined with our findings from subsection 2.6, where we found out that using the truncated SVD is equivalent to performing noise reduction, this means that we could expect similar clustering performance when performing k-means on our dimensionality reduced data.

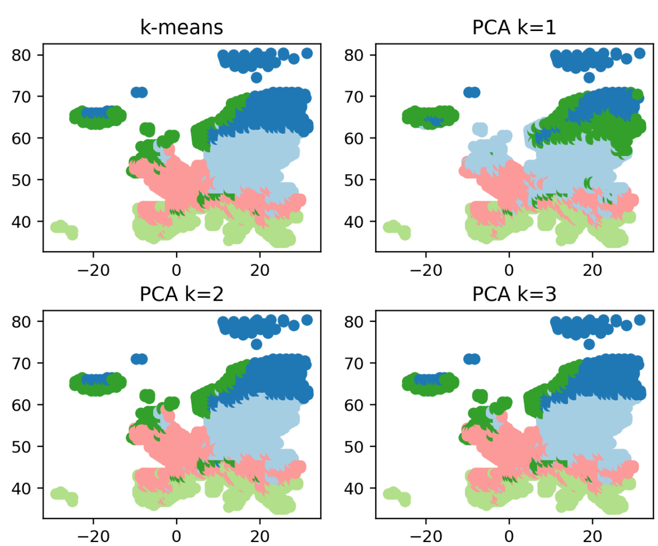


Figure 14: k-means and PCA based clusters for k 1 to 3