Discussion

In conclusion, all main goals of our project have been achieved. We wrote our own KNN algorithm and were able to accurately recognize digits at the very beginning of our project. Further work was mainly improvements and optimizations of this algorithm. We decided to use a KNN function from scikit-learn, which significantly increased the speed of our calculations. We also performed PCA, which not only accelerated the calculations, but also increased the accuracy by noise reduction. We have found, which values for parameters as number of neighbors for KNN and number of dimensions for PCA correspond to the peaks of accuracy in digit recognition. Although using only the first two PCs allowed us to predict the label of an image with more than 30% probability, the variance, explained by PC1 and PC2, hardly amounts to 10%. Therefor we had to discard the idea to plot our data in the representative 2D space. We successfully plotted reduced images after PCA using different numbers of dimensions along with blurring and gray transition. The PCA extracted the “essence” of the images, which made it possible to correctly recognize digits restored from the first 10 PCs using KNN and our subjective human eyes as well.

As one of the last parts of our main project, we calculated three different “metadigits”: mean, median and “with best neighbors”. We calculated the average distance from these different metadigits to the digits with the same label. The mean digit has the lowest average distance, while the “best neighbored” has the highest. These average distances give some information about how tight the digits with the same label are clustered. It helped us to understand the structure of our dataset better and to predict some complications, which were confirmed later. Although the “best neighbored” digit has the highest average distance, we have selected it as metadigit for further calculations in the part, in which we evaluated our own written digits. The reason is, that only this type of metadigit represents the real existing data point, while other are the mixture of many digits with the same label.

Finally, we have made the drawing canvas and identified which images were falsely recognized by our algorithm even with fully optimized parameters. Digits are more often falsely recognized due to the fact that they have many different writing-styles or are suffering from wrong centering. The best example for both cases is “7”, which sometimes has a middle touch and sometimes does not and is also very variable in the position of the highest horizontal line. The digits that are commonly mixed up, are determined by the structural similarity, like “4” and “9” or “9” and “7”.

All in all, our must-have project answered some crucial questions about digit recognition with KNN and figured out some important parameters together with some features in our dataset structure. These results were used in our more specific project, which we will gladly present on Thursday.