

ECS 271 - SVD-Based Classification - Explanation and Results

Jared White

November 10th, 2025

SVD-Based Classification

One of the classification methods we tested to classify emotions based on landmark face data was a Singular Value Decomposition (SVD)-based method. This method reduces the dimensionality of our training data by using a Rank- k SVD and evaluates the classification performance based on reconstruction errors. We evaluate the effectiveness of this approach using entropy metrics and confusion matrices, highlighting its benefits and limitations.

Rank-k SVD

On our training data, we apply Rank- k SVD, where k is the number of components used to approximate the data. With the SVD, a data matrix X can be decomposed into three matrices:

$$X \approx U_k S_k V_k^T$$

where U_k and V_k^T contain the left and right singular vectors, and S_k contains the singular values corresponding to the first k components. Naturally, as k gets closer to the rank of our data matrix, which would be less than or equal to the number of data parameters, our approximation gets closer to the original matrix.

We do this for each emotion label. By doing so, we have hopefully crafted SVDs that can properly capture the underlying data structures specific to each emotion label.

Reconstruction and Error Calculation

For the test data X_{test} , the reconstruction error is computed by projecting the test data onto the left singular vectors U_k for each of the corresponding emotion class. The error for each sample is then measured as the Euclidean distance between the original test sample and the reconstructed approximation:

$$\text{error}_i = \|X_{\text{test},i} - U_k S_k V_k^T\|_2$$

The resulting error matrix is used to classify the test samples by selecting the class with the smallest reconstruction error. The logic on this is that the left singular vectors should be able to accurately reconstruct data vectors in their domain, hence, data with the same emotion label.

Entropy and Confidence Calculation

To quantify the uncertainty of the model's predictions, we compute the entropy for each classifier based on the structure of the classifier itself. The methodology of using a SVD-based entropy calculation comes from this paper, <https://cinc.org/archives/2024/pdf/CinC2024-150.pdf>, wherein they give this formula:

$$H = - \sum_{i=1}^M \bar{\sigma}_i \log_2(\bar{\sigma}_i)$$

where $\bar{\sigma}_i$ are the normalized singular values. They can be calculated as the singular value divided by the sum of all singular values.

Using such calculations for the entropies, we can get a general understanding on the certainty that each classifier has in classifying data related to it. Lower entropy means higher certainty, while higher entropy means lower certainty. With \log_2 entropy calculations, the highest entropy we could get would be $\log_2(6) \approx 2.5849625$, which would imply that the model makes uniform classifications.

Benefits of SVD-based Classification

The SVD-based classification approach offers several advantages:

- **Dimensionality Reduction:** SVD effectively reduces the dimensionality of the feature space, which can improve both computational efficiency and model generalization.
- **Interpretability:** The singular values and vectors obtained from SVD can provide insight into the underlying structure of the data.
- **Robustness to Noise:** By focusing on the most significant components of the data, SVD can reduce the impact of noise in the lower-rank components.

Disadvantages of SVD-based Classification

Despite its benefits, SVD-based classification has some limitations:

- **Computational Complexity:** Computing SVD can be expensive for large datasets. While our dataset only has 6 class labels, it does have a high amount of samples and around 1k data points for each sample. This can lead to the calculations taking quite a long time.
- **Limited Performance for Complex Data:** While SVD works well for linearly separable data, its performance may degrade with more complex, non-linear data distributions. As the landmark face data is not linearly separable, it is susceptible to this issue.
- **Choice of k:** The performance heavily depends on the choice of k , the rank of the decomposition. Selecting an optimal value for k would require experimentation and cross-validation.

If we wanted to have the SVD-based Classification tests be more akin to realistic testing, we would allocate around half of the training images to serve as a validation set to find the optimal k for the SVD, where we would then run classification on the training set. However, this would require testing a wide number of k values, and this would take quite a long time, which we will discuss later.

Testing SVD-based Classification

For the testing section, classification was done with the same separation of training and testing data, with 70 percent of the data for training and the remainder for testing. I began by testing the classification results with different rank k approximations:

$k = 25$:

Average Entropy: 0.9092454603026966

Entropies per data class:

- Angry: 0.9401354466223886
- Disgust: 0.9219908474864223
- Fear: 0.8909344278311391
- Happy: 0.943929383029909
- Neutral: 0.8707473664034893
- Sad: 0.8877352904428307

Accuracy Rate: 0.373464600061099

Confusion Matrix:

8285	3253	5621	2285	4724	3575
3425	13276	5465	3666	2491	3820
4588	3224	11072	2167	4007	4978
2778	2693	3709	14739	2265	1770
3176	1653	3980	1483	8868	5192
4242	3446	6678	2016	6327	8552

Overall, the entropy values suggest that the model is not simply guessing at random. The entropy per class remains relatively consistent, with the lowest confidence observed in the classes related to Anger and Happiness. This is a noteworthy observation, as these emotions may be inherently more difficult to distinguish from other classes (such as Fear or Sadness) due to similarities in feature representations.

The average entropy across the test samples is around 0.91, which is not close to the maximum entropy value of $\log_2(6) \approx 1.79$ (which would indicate complete uncertainty or random guessing). Thus, the model does exhibit some degree of confidence in its predictions, although it's not highly confident.

The accuracy of the model is approximately 37 percent, which is a modest result, but it does suggest that the model has some predictive power. Given that this is a multi-class classification problem with six different emotions, achieving 37 percent accuracy indicates that the model is performing better than random guessing, which would yield an accuracy close to 16.7 percent (since there are 6 possible classes). Although 37 percent is not ideal, it's a reasonable starting point, and we expect the accuracy to improve as we increase the rank of the SVD (i.e., the value of k). This would allow the model to capture more of the underlying structure in the data, potentially improving classification performance.

I would like to note here that the time to calculate the SVDs and classify the test data over all the data available took 4 minutes and 20 seconds.

k = 50:

Average Entropy: 0.9838690442852428

Entropies per data class:

- Angry: 1.0165242460159976
- Disgust: 1.0003959298225205
- Fear: 0.9637729112723595
- Happy: 1.0208975849866897
- Neutral: 0.9411158876307879
- Sad: 0.9605077059831006

Accuracy Rate: 0.4251102322832856

Confusion Matrix:

9282	1969	4663	2206	5419	4204
2859	17382	4432	1714	1764	3992
3822	2404	11493	2276	4518	5523
2421	1986	2674	15954	2885	2034
2782	914	3489	1010	11897	4260
3421	3140	6123	1335	7049	10193

As you can see, the entropy values for most classes hover around 1. While this indicates that the model's certainty is higher than complete randomness (entropy of $\log_2(6) \approx 1.79$), the increase in entropy shows that the model is becoming more uncertain about its classifications as the dimensionality of the features increases.

The accuracy of the model has improved to 42.5 percent, which is a marginal improvement from the earlier accuracy of 37.3 percent. This suggests that increasing k helps the model capture more variance and thus improves its classification ability, albeit slightly. Additionally, we can see from the confusion matrix that the same issues with classification were present between the different sizes of k , although they were slightly reduced with the larger k .

I would like to note as well that these calculations took 12 minutes and 38 seconds.

Conclusions on SVD-Based Classification

Overall, SVD-Based Classification of emotion based on landmark face data is possible, but not the best method to go about classifying the data. One issue we brought up earlier that was not done in testing was the need to separate our data into training, testing and validation data. The validation data set would serve to let us test different values k to pick the best performing one to do our final test predictions on. However, we can see from the test results that the overall computation time increases as k increases, and this increase is not linear. If we wanted to test a range of k 's, say 10 to 100, we could have the program running for a few hours to get all the math done. And, this would need to be done each time we run the program, as the SVDs we get are entirely based on the training data itself.

We could also see from our testing that even though the test accuracy increased when k was higher, it actually led to an increase in entropy. That means that even though we are predicting well, the model is not predicting these values with that high of confidence. We want for our model to not only be correct in its predictions, but we want it to have a good degree of confidence. However, our calculations indicate that there may exist a trade-off between the accuracy of the model and its prediction confidence. We can't just ignore the entropy even if we are getting better results, because we ultimately want to report our results with a large degree of confidence.

With such lackluster results coming from the SVD-based Classification method, it is time to move on to another method that may prove more fruitful in developing an emotion classifier using landmark face data.