Machine Learning & Data Mining
Miniproject 2

Caltech CS/CNS/EE 155
February 2019.

---

1 Introduction [10 points]

Team Members

Matt Riker and Spencer Schneider

Team Name

Goose Jawns

Division of Labor

Matt: Piazza Post, Basic Visualizations, Matrix Factorization Visualizations, Report

Spencer: Piazza Post, Matrix Factorization Methods, Matrix Factorization Visualizations, Report

2 Basic Visualizations [20 points]

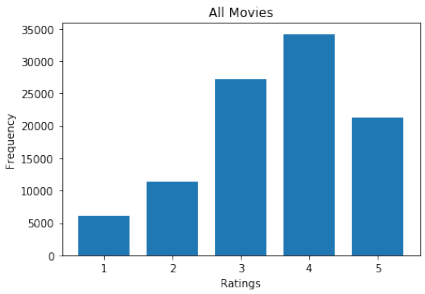

Figure 1: All Ratings

As we can see there is a pretty large bias toward higher ratings. This isn't unexpected as people are more likely to watch movies that they hear are good. In other words if a movie is bad not many people will watch it so frequency of bad ratings is expected to low.

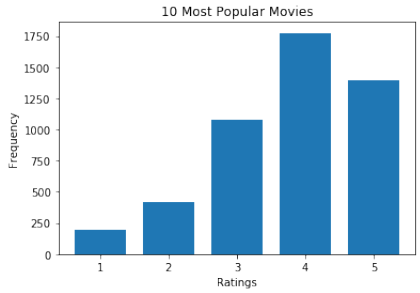

Figure 2: 10 Most Popular Movies' Ratings

For the most popular movies we see an even greater bias toward higher ratings than all movies. This makes sense as popular movies become popular because they are good movies and good movies are more likely to get higher ratings.

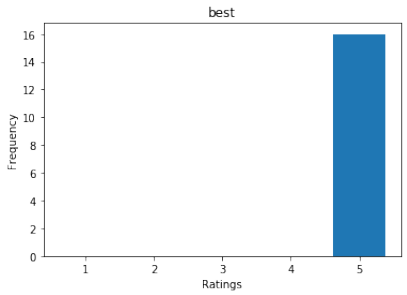

Figure 3: 10 Best Movies' Ratings

For the best rated movies we see only values of 5. It looks like these ratings are much higher rated than the popular movies, but it is just because we have very few ratings for each of these movies. In total there were only 16 ratings for these 10 movies, so often times it was just 1 5.0 rating that put the movie in this category. This makes sense since we have such a small sample size which leads to a high variance.

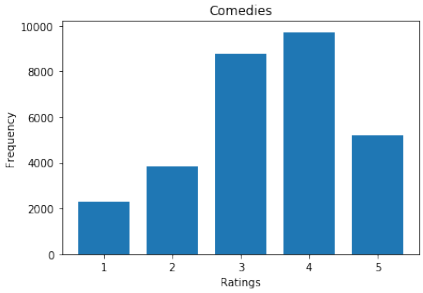

Figure 4: All Ratings from Comedies

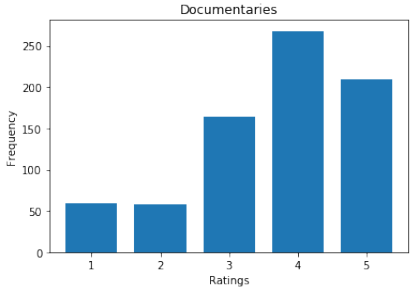

Figure 5: All Ratings from Documentaries

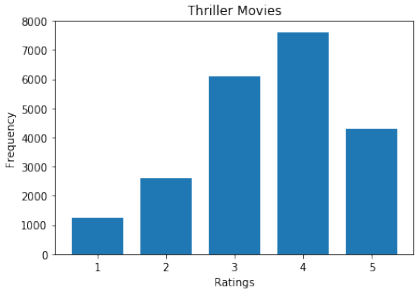

Figure 6: All Ratings from Thrillers

The thriller and comedy genres essentially follow the distribution of all movies. This makes sense as they are fairly popular movie genres so they make up a solid portion of all movies. The documentary genre slightly varies from this distribution as it is has better ratings. This makes sense because there are fewer documentary movies than comedy and thrillers, and people are more likely to watch a documentary on topics they like, leading to higher ratings.

3 Matrix Factorization Methods [40 points]

Normal SVD

Using our code from homework 5 we performed a Singular-Value Decomposition (SVD) without any bias terms. This is the only method we used that didn't have a bias term. How this works is we randomly initialize to matrices U and V and use stochastic gradient decent to get the U and V describe the data the best when being multiplied together. In our case the U matrix corresponds to the user data and the V matrix corresponds to the movie data. When we run SVD on V we can get the principle components of the movies and we then got the first two principal components so that we could more easily visualize them. We selected regularization and learning rate by plotting these variables vs out of sample error. We ended up with regularization strength of .01 and a learning rate of .01.

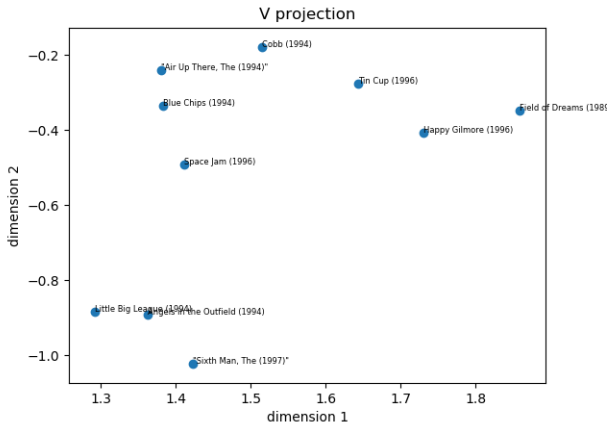

Figure 7: Our Choice of 10 Movies

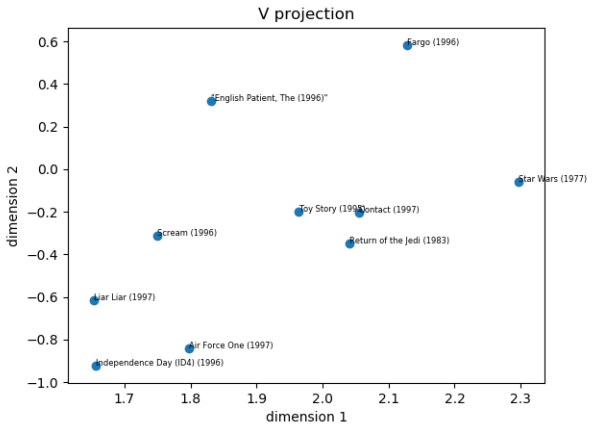

Figure 8: 10 Most Popular Movies

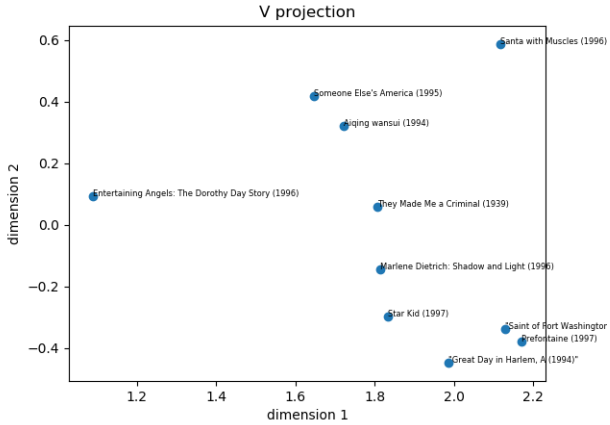

Figure 9: Best 10 Movies

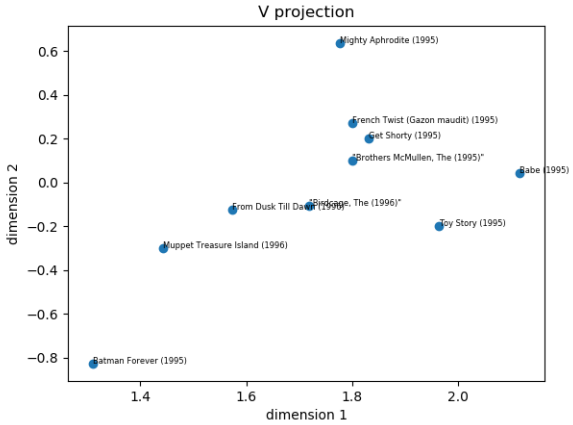

Figure 10: 10 Comedy Movies

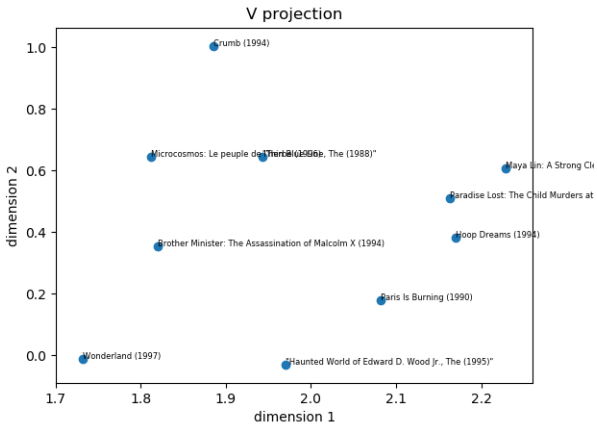

Figure 11: 10 Documentary Movies

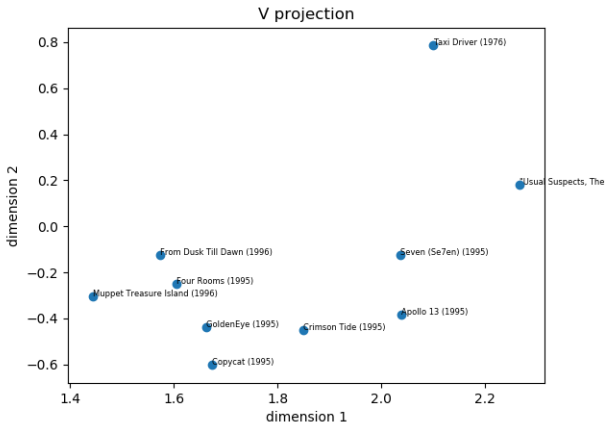

Figure 12: 10 Thriller Movies

SVD with bias

For our second method of Singular-Value Decomposition we used our code from homework 5 as described in method one with slight modifications. We added a bias terms that signified a user or movie specific deviation away from the global bias which was the average of all movie ratings in our case. So if a movie or user was significantly different from the average user or movie the error term and in turn the gradient decent would account for it.We selected regularization and learning rate by plotting these variables vs out of sample error. We ended up with regularization strength of .01 and a learning rate of .01.

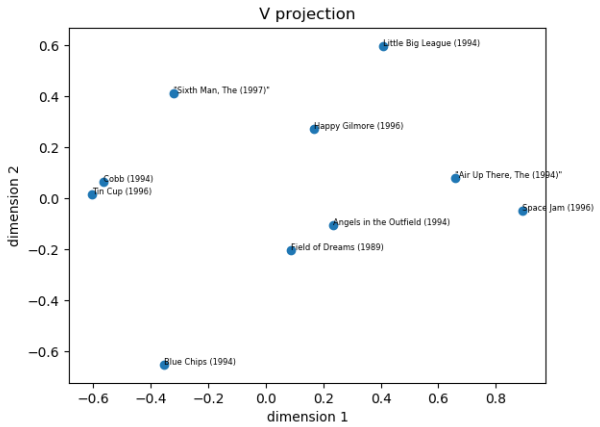

Figure 13: Our Choice of 10 Movies

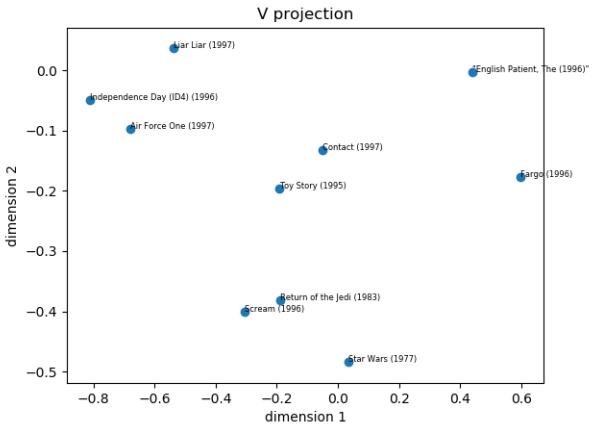

Figure 14: 10 Most Popular Movies

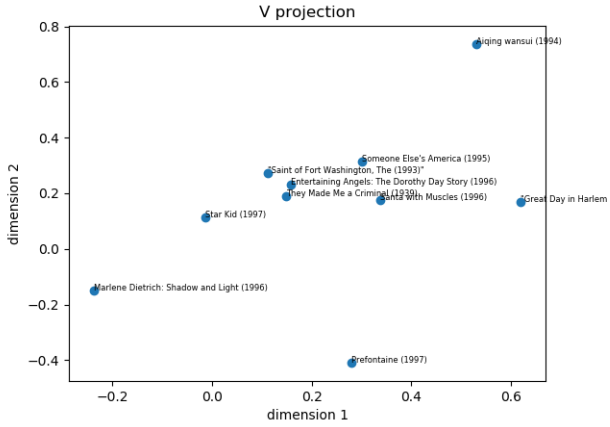

Figure 15: Best 10 Movies

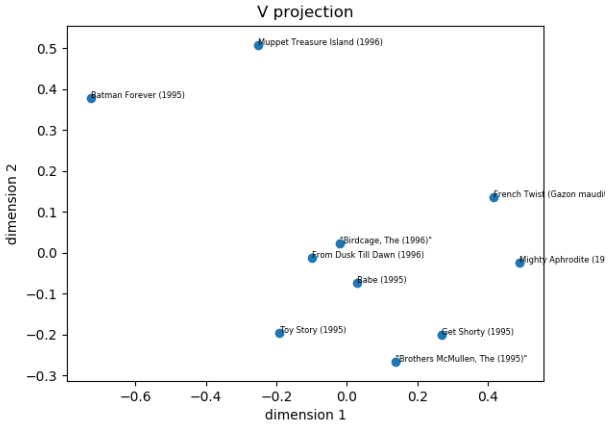

Figure 16: 10 Comedy Movies

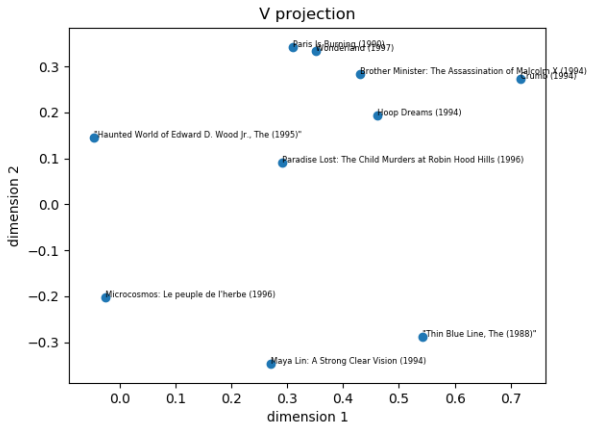

Figure 17: 10 Documentary Movies

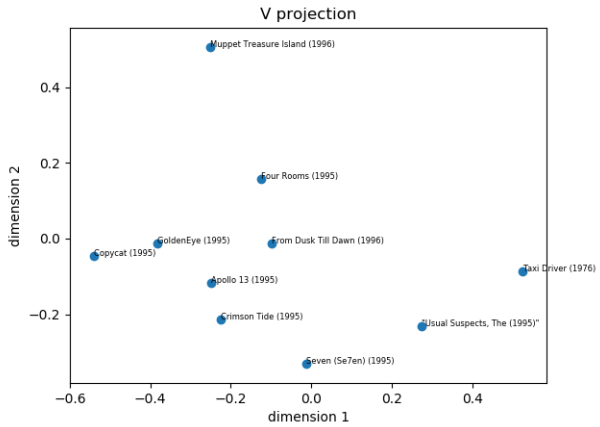

Figure 18: 10 Thriller Movies

Off the Shelf Implementation

We used the Surprise package for our off the shelf implementation. We used grid search to find the best parameters to use for our SVD. This implementation is essentially the same as the SVD with bias that we coded. We ended up with regularization strength of 0.1.

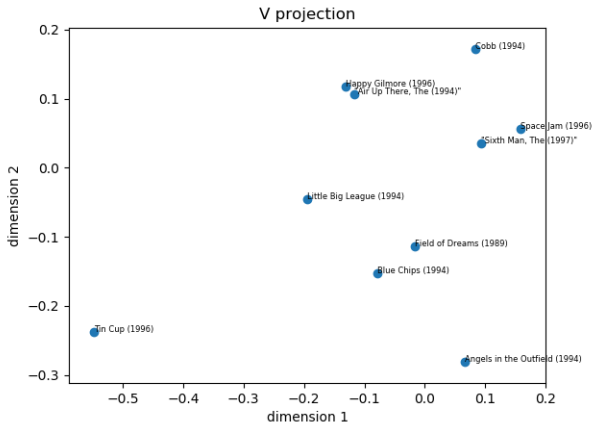

Figure 19: Our Choice of 10 Movies

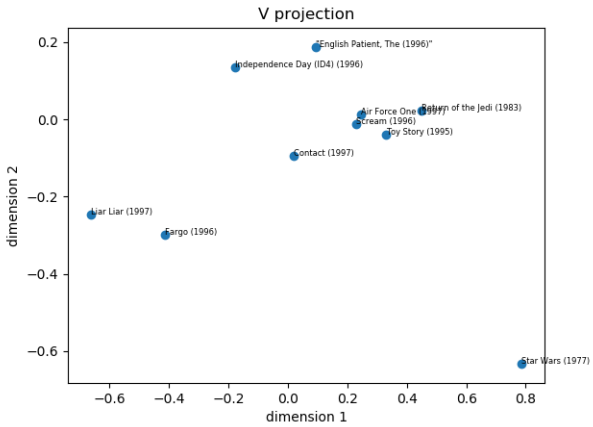

Figure 20: 10 Most Popular Movies

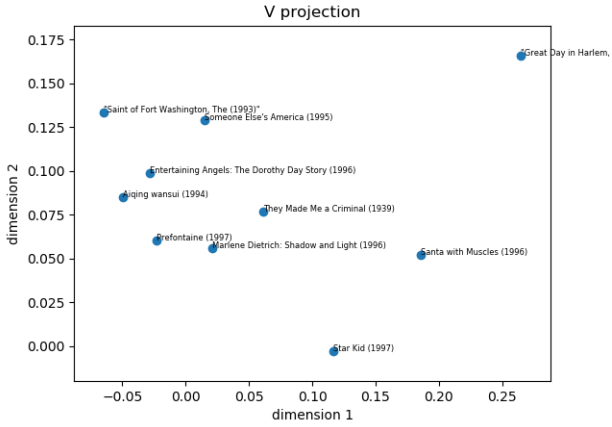

Figure 21: Best 10 Movies

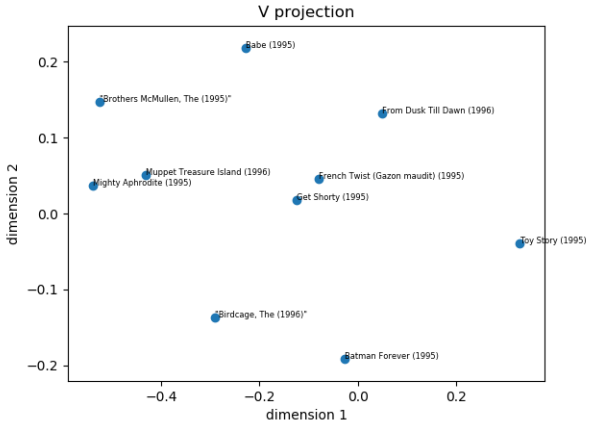

Figure 22: 10 Comedy Movies

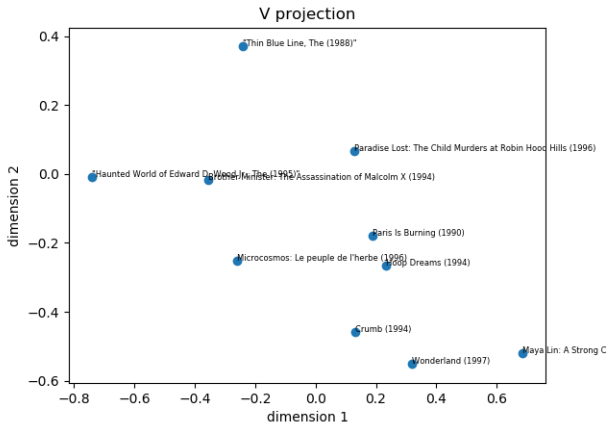

Figure 23: 10 Documentary Movies

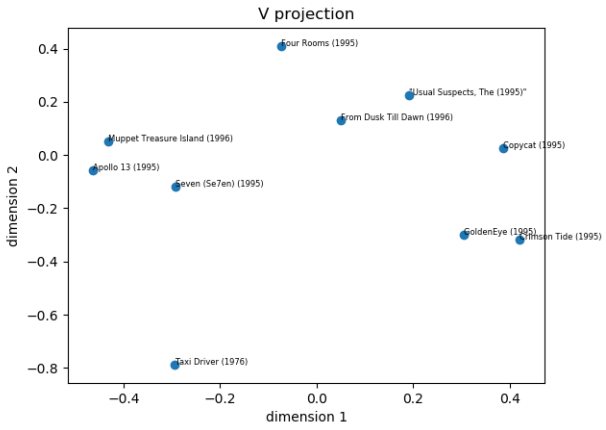

Figure 24: 10 Thriller Movies

Method Comparisons

As mentioned the off the self implementation is very similar to our implementation with bias. Thus, these two methods produced similar patterns and results while the first method showed different trends. The implementation with no bias had an error of .431 on the test set. Our implementation with bias had an error of .267 on the test set. The off the self implementation had an error of .261 on the test set. We had to square and half the error of the surprise package error because it is RMSE rather than 0.5 * MSE. We can see that the method with no bias performed significantly worse than the ones with bias. This can be explained by the fact that the bias helps account for more diverse users and movies. The bias term allows the SVD to better model users and movies that have qualities that are different from the average movie. The method with no bias doesn't have this aid so it isn't accurate.

4 Matrix Factorization Visualizations [30 points]

Overall, for most of the plots the placement of the points appears to be random, and there are no strong definite trends.

The distribution of movies of the most popular movies is different than the visualization for the best movies for all three methods. For method 1, the range of values for dimension 1 and dimension 2 are the same, the movies are just distributed differently throughout different parts of the graph. For method 2, the values on the axis are different as the points are much more spread out along dimension 1 for most popular movies (ranging from -0.8 to 0.6) and more clustered along dimension 2 (-0.5 to 0.0). On the contrary, for best movies 7 of the 10 movies are all clustered together, 0.0 to 0.6 in dimension 1 and 0.1 to 0.4 in dimension 2. This trend continued in method 3 as all 10 points are spread out over a wider range of values for the most popular movies, but all 10 points were clustered together over a much smaller range of values for both dimension 1 and dimension 2 for the best movies. This trend suggests that the 10 best movies all had certain similar features that our methods trained on. This could be because the best movies all have very few ratings so there is not much to be learned from them so they are clustered together despite their thematic differences. The popular movies on the other hand have very many ratings to they're more spread out as their features are more easily learned.

The three genres we selected were comedy, documentary, and thriller. For method 1, all three of the distributions of movies appear to be random for each graph, with no definite trends. The range of values for dimension 1 appears to be constant among all three genres, however dimension 2 contains a range of both positive and negative values for thriller and comedy movies, but these values are only positive for documentary movies. This suggests that for this method we could have a gradient going along the y-axis from action to non-action packed. Most of the thriller movies fall under action whereas about half of the comedies do and none of the documentaries do. For method 2, we see a very similar trend to this one however, it is the range of values that appear to be random for dimension 2, and the values for dimension 1 seems to follow a very similar trend that dimension 2 did for method 1. This suggests we could have the same action packed to non-action packed gradient for dimension 1. For method 3, no definite trends appear and the distribution of points seems to be random among similar values for both dimension 1 and dimension 2 for comedy, documentary, and thriller.

Among all three methods there appears to be a similarity in all three of the thriller movie graphs.

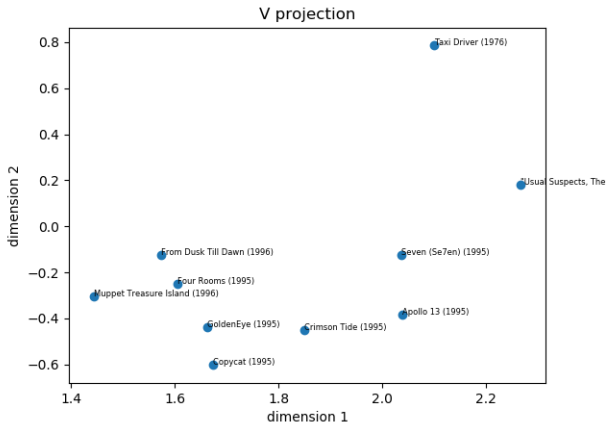

Figure 25: Thriller Movies
Method 1

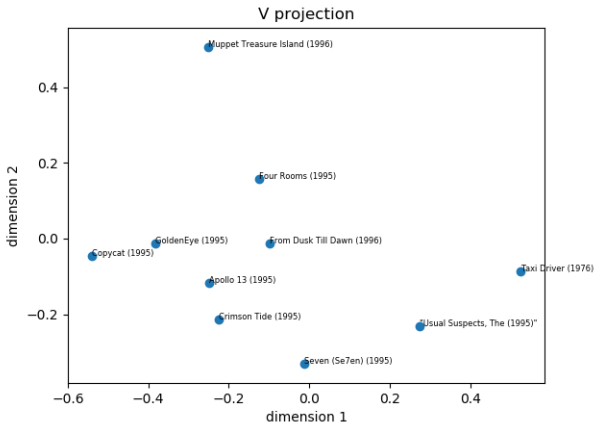

Figure 26: Thriller Movies
Method 2

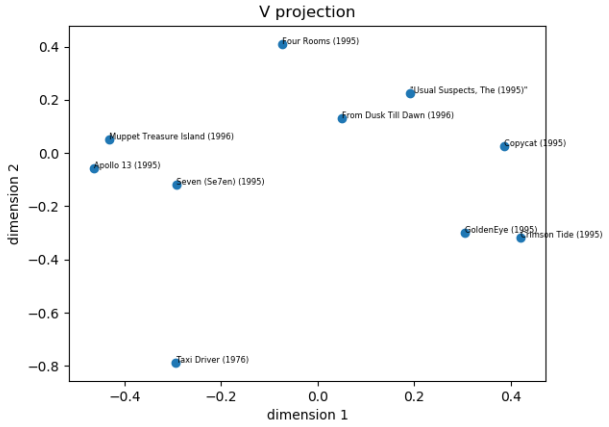

Figure 27: Thriller Movies
Method 3

For each one, the movies are randomly scattered along the x dimension, however 9 of them are predominately in one region of the y dimension, and then there is one outlier point away from these 9. Interestingly, this outlier point is different for method 2, but it is the same Taxi Driver movie for methods 1 and 3. It appears however that the outlier for method 2, Muppet Treasure Island, seems to make more sense as an outlier as it is more of a children's movie than the rest. For method 3, this movie is much lower along the y-axis than than the other points, but is much higher along the y-axis than the other points for method 1.

Overall, the biggest differences between the three methods that we used was the range of values for dimension 1. Using method 1 all of the graphs ranged from anywhere above 1 to above 2, whereas in methods 2 and 3 nearly all of the graphs produced ranges from negative values to positive values (all between -1 and 1). Other than these, all three methods generally produced random distributions with

similar values along dimension 2.

We expected to see method 2 and 3 produce similar results because they are doing essentially the same thing in their factorization. However, we saw that this wasn't the case very consistently. In fact, there are some instances where method 1 and 3 produce more similar results. This is surprising (maybe this is where the package name comes from), but could stem from the fact that we don't know if surprise is doing the exact same thing as our method 2. While we know it has bias terms we don't know if the gradient descent is the same.