1 Different S matrices

This is from a svd decompostion of both movie and song matrices (movies are the user source and songs are the items source). left s uses the s from the user source svd decomposition. right s uses the s from the item source svd decomposition. The combined multiplies the two s matrices together, then takes the square root of the diagonal elemnts to be the s matrix. Both matrices were 8 by 8. The error is obtained by permuting the columns of the item source matrix, then decomposing it, and using its right (lf x item) matrix and the left (user x lf) matrix from the user source decompoistion to recreate a ratings matrix. Then the difference between the true ratings matrix (obtained with by taking the known loadings of users from the users source matrix, and the known item loadings of the item source matrix) was found, and the mean sum of the sqaured errors was gotten. below is the summary statistics of one "test" (? sample maybe?)

S - type	Mean	Median	Std. Dev
left (user source) s	0.17867	0.17685	0.0198
right (item source) s	0.27572	0.27333	0.02191
combined	0.2199	0.21775	0.0208

1.1 next steps

Some obvious next steps are:

- Is the square score just the (weighted byt stddev) aveage of the two
- More thourough test of errors for statistical significance (bootstrapping)
- How does the size of the matrix affect the errors
- How does the make up of the columns (more "diverse" vs. less) affect the distributions of errors using a type of s-matrix
  - does move diversity in item source make item-s better than user-s or vice versa
- time results
- nmf?
- number of latent features fit
- truncation of latent features

2 NMF

same set up as svd, but now nmf decompostion

	mean	median	std
nmf	1880	0.31855	265371

2.1 next steps

- compare nmf to svd for different size matrices and number of latent features