1 Factoring Matrices

A surprising number of useful things can be described as matrix factorization. Given matrix X, find matrices A and B such that $X = AB$. For example, when you factor the identity matrix I, you end up with a matrix and its inverse, e.g. $AA^{-1} = I$.

Obviously if you try to factor any arbitrary matrix X into A and B the result is not unique—factoring number 6, you could end up with 3 and 2 or 6 and 1. Similarly, the result is likely not going to be exact, so you end up with a residual error matrix E, e.g.: $X = AB + E$.

The actual factors A and B depend on the factoring algorithm, and implicit and explicit constraints. Different factoring rules lead to different semantics which may be useful in different ways.

2 Nonnegative Matrix Factorization

Nonnegative matrix factorization takes a matrix X with all positive numbers, and produces two factor matrixes A and B both with only poistive numbers.

We decide on the dimensions of A and B, initialize them to random values, then iterate: $Y = AB$, then compute cost via $(X - Y)^2$, if it is small, end iteration, and A and B are the factors. Else adjust A via $A = (A^T X) \times 1/(A^T Y)$ and B via $B = XB^T \times 1/(A * B * B^T)$ where $\times$ is a scalar multiplication.

3 Eigenvectors & Eigenvalues

An eigenvector of a matrix is a vector that is not rotated by the matrix—only scaled. For example, $Xu = u\lambda$. The λ scalar is the eigenvalue.

Instead of just multiplying by one vector, we can multiply by a lot of vectors, arranged as column vectors in matrix U. Similarly, we can have lots of corresponding λs in a diagonal matrix Σ, leading to: $XU = U\Sigma$. Multiplying both sides by U^{-1} we end up with $X = U\Sigma U^{-1}$. This is a special case of a more general Singular Value Decomposition (SVD), which ends up with $X = U\Sigma V^*$ where V^* is a conjugate transpose of V. In the general SVD, the matrices U and V do not have to be related (e.g. they are not inverses of each other).

Often, the diagonal matrix Σ (with coresponding U and V entries) is sorted in decreasing order. The resulting matrices are now fully determined by the starting matrix X, with the largest eigenvalue corresponding to an eigenvector representing the dimension of highest variation, and second largest eigenvalue to dimension of second-highest variation, and so on.

What this means is you can take a dataset, apply linear transformation to it, apply SVD to it, and the eigenvectors you get (in correct descending order) will be the same eigenvectors you get before the linear transformation.

Practical example of this: Consider 10 cameras capturing the same object bouncing around. Each camera registers the x and y coordinate of an object, moving on its 2D screen. (in reality, the object is bouncing around in 3D). You can put all the x, y coordinates from each camera into a single matrix, and do SVD. Even though you started with 10 cameras each providing x, y (e.g. 20 dimensions), they are all observing something that is happening in 3-dimensions, so after SVD, the top three singular values will correspond to the 3-dimensions of variation, with remaining singular values being nearly zero.

4 Principal component analysis

TODO:...