Introduction

For this miniproject, we wanted to visualize movies and users, in clusters, based on an existing matrix Y, where we had m rows corresponding to m user_id values, and n columns for each of n movies. We first learned a Latent Factor Model in $U^{\intercal}$ and V, which each had dimension M x K and K x N respectively. We accomplished this by ysing Stochastic Gradient Descent after initializing U and V to random values between 0 and 1. See below for the gradient functions we optimized. Furthermore, after learning U and V, we visualized these by projecting them to 2 Dimensions and making a scatterplot of both movie and user data.

Stochastic Gradient Descent (Basic Formulation)

In implementing Stochastic Gradient descent, we computed the gradeint of the following expression w.r.t $u_n, v_m \forall n \in U, m \in V$ for a given point y_{ij}.

$$l = \min_{U,V} \frac{\lambda}{2N}(||U||_{Fro}^2 + ||V||_{Fro}^2) + (y_{ij} - u_i^{\intercal} v_j)^2$$

Stochastic Gradient Descent (Advanced Formulation)

Since we are implementing stochastic gradient descent, we want to compute the gradient of the following expression w.r.t $u_n, v_m \forall n \in U, m \in V$ for a given y_{ij}.

$$l = \min_{U,V,a,b} \frac{\lambda}{2N}(||U||_{Fro}^2 + ||V||_{Fro}^2 + ||a||_{Fro}^2 + ||b||_{Fro}^2) + \frac{1}{2}(y_{ij} - \mu - (u_i^{\intercal} v_j + a_i + b_j))^2$$

We have the following process for stochastic gradient descent:

1. Randomly pick a y_{ij}
2. Compute the gradient of the error function $l(i,j)$ with respect to every column in U and every column in V, and every entry in a and b.
3. Subtract $\eta\nabla$ from U V, a, and b
4. Repeat steps 1-3 until $\eta\nabla < \epsilon$.

We can compute the following partial derivatives:

$$\frac{\partial l}{\partial_{a_k}} = \frac{\lambda}{N} a_k - \mathbb{1}^{i=k}((y_{ij} - \mu) - (u_i^{\intercal} v_j + a_i + b_j))$$

$$\frac{\partial l}{\partial_{b_j}} = \frac{\lambda}{N} b_j - \mathbb{1}^{j=k}((y_{ij} - \mu) - (u_i^{\intercal} v_j + a_i + b_j))$$

$$\frac{\partial l}{\partial_{u_k}} = \frac{\lambda}{n} u_k - \mathbb{1}^{k=i} v_j((y_{ij} - \mu) - (u_i^{\intercal} v_j + a_i + b_j))$$

$$\frac{\partial l}{\partial_{v_k}} = \frac{\lambda}{n} v_k + \mathbb{1}^{k=j} u_i((y_{ij} - \mu) - (u_i^{\intercal} v_j + a_i + b_j))$$

Choice of Parameters

As per the set, we want to over-specify the rank of U and V, so we will pick $k = 20$. Based on some initial trials, we will pick $\eta = 0.001$.