Matrix factorization attempts to identify "latent factors" that determine the interest level of a given *user* in a given *item*, such as a movie or a book. In particular, it attempts to represent both a user's preferences and an item's composition by an F-dimensional vector whose entries reflect some abstract factors. We then estimate a user's rating of an item with the dot product between the user's vector and the item's vector.

Suppose we have N users and M items (with $F < M$). Let the $(n, m)^{\text{th}}$ entry in the matrix R give user n's rating of item m (for $n \in \{1, 2, \dots, N\}$ and $m \in \{1, 2, \dots, M\}$). Since users do not necessarily rate all M items, R will have many missing values.

Matrix factorization attempts to approximate the matrix $R \in \mathbb{R}^{N \times M}$ with a user-factor matrix $U \in \mathbb{R}^{N \times F}$ and an item-factor matrix $V \in \mathbb{R}^{M \times F}$. The n^{th} row in U represents user n's F latent factors, and the m^{th} row in V represents item m's F latent factors. We then estimate R with

$$\hat{R} = UV^\top.$$

Note that, as desired, this models user n's rating of item m with the dot product between user n's factor vector and item m's factor vector. For instance, the $(n, m)^{\text{th}}$ entry in $\hat{R}$, $\hat{r}_{nm}$, is given by the dot product

$$\hat{r}_{nm} = \mathbf{u}_n^\top \mathbf{v}_m = \sum_{f=1}^{F} u_{nf} v_{mf},$$

where $\mathbf{u}_n$ is the n^{th} row of U and $\mathbf{v}_m$ the m^{th} row of V, each represented as an F-dimensional column vector.

Now that we have a model specified, we must estimate the parameters: the entries in U and V. We start by choosing a loss function. We aim to minimize the following:

$$L = \frac{1}{2} ||R - \hat{R}||^2,$$

where $|| \cdot ||^2$ is the sum of the matrix's entries squared. In other words, we want to minimize the sum of the squared differences between the observed and fitted ratings. This can equivalently be written as

$$L = \frac{1}{2} \sum_{(n,m)} (r_{nm} - \hat{r}_{nm})^2 = \frac{1}{2} \sum_{(n,m)} \left(r_{nm} - \sum_{f=1}^{F} u_{nf} v_{mf} \right)^2$$

for all user-item pairs(n, m). Note, however, that R has many missing values. Since it's impossible to calculate the loss of an estimate of an unknown value, we can just set the missing values in R and the corresponding values in $\hat{R}$ to 0.

We will estimate the entries of U and V with *gradient descent*. So, our next step is to calculate the partial derivative of the loss function with respect to these entries. First consider the derivative with respect to u_{nf}, the n^{th} user's f^{th} latent factor:

$$\frac{\partial L}{\partial u_{nf}} = -\sum_{m} \left((r_{nm} - \hat{r}_{nm}) \cdot v_{mf} \right).$$

Similarly for v_{mf}, the m^{th} item's f^{th} latent factor:

$$\frac{\partial L}{\partial v_{mf}} = -\sum_n \left((r_{nm} - \hat{r}_{nm}) \cdot u_{nf} \right).$$

We can then conduct gradient descent by iteratively adjusting U and V with these derivatives. However, we can more conveniently express the derivatives in matrix form. First, let E be the matrix of residuals:

$$E = \hat{R} - R \in \mathbb{R}^{(N \times M)}.$$

Then note that the expression for $\partial L / \partial u_{nf}$ implies that

$$\frac{\partial L}{\partial U} = EV$$

and similarly the expression for $\partial L / \partial v_{mf}$ implies that

$$\frac{\partial L}{\partial V} = E^\top U.$$

This matrix representation shows how to calculate the derivative of the loss function with respect to every parameter simultaneously, which is useful for *batch gradient descent*. We may wish instead to conduct *stochastic gradient descent*. For stochastic gradient descent, we can update our parameters based on the gradient of the loss function applied to just one user-item pair at a time—i.e. rather than looking at L we look at $L_{nm} = \frac{1}{2}(r_{nm} - \hat{r}_{nm})^2$.

We can start by considering the partial derivative of L_{nm} with respect to just one entry in U: u_{nf} (note that the partial derivative with respect to $u_{\ell f}$ is 0 for $\ell \neq n$):

$$\frac{\partial L_{nm}}{\partial u_{nf}} = -(r_{nm} - \hat{r}_{nm}) \cdot v_{mf},$$

and similarly,

$$\frac{\partial L_{nm}}{\partial v_{mf}} = -(r_{nm} - \hat{r}_{nm}) \cdot u_{nf}.$$

Then, we can simultaneously calculate the partial derivative with respect to all F entries in $\mathbf{u}_n$ and all F entries in $\mathbf{v}_m$:

$$\begin{aligned} \frac{\partial L_{nm}}{\partial u_{nf}} &= -(r_{nm} - \hat{r}_{nm}) \mathbf{v}_m \\ \frac{\partial L_{nm}}{\partial v_{mf}} &= -(r_{nm} - \hat{r}_{nm}) \mathbf{u}_n. \end{aligned}$$