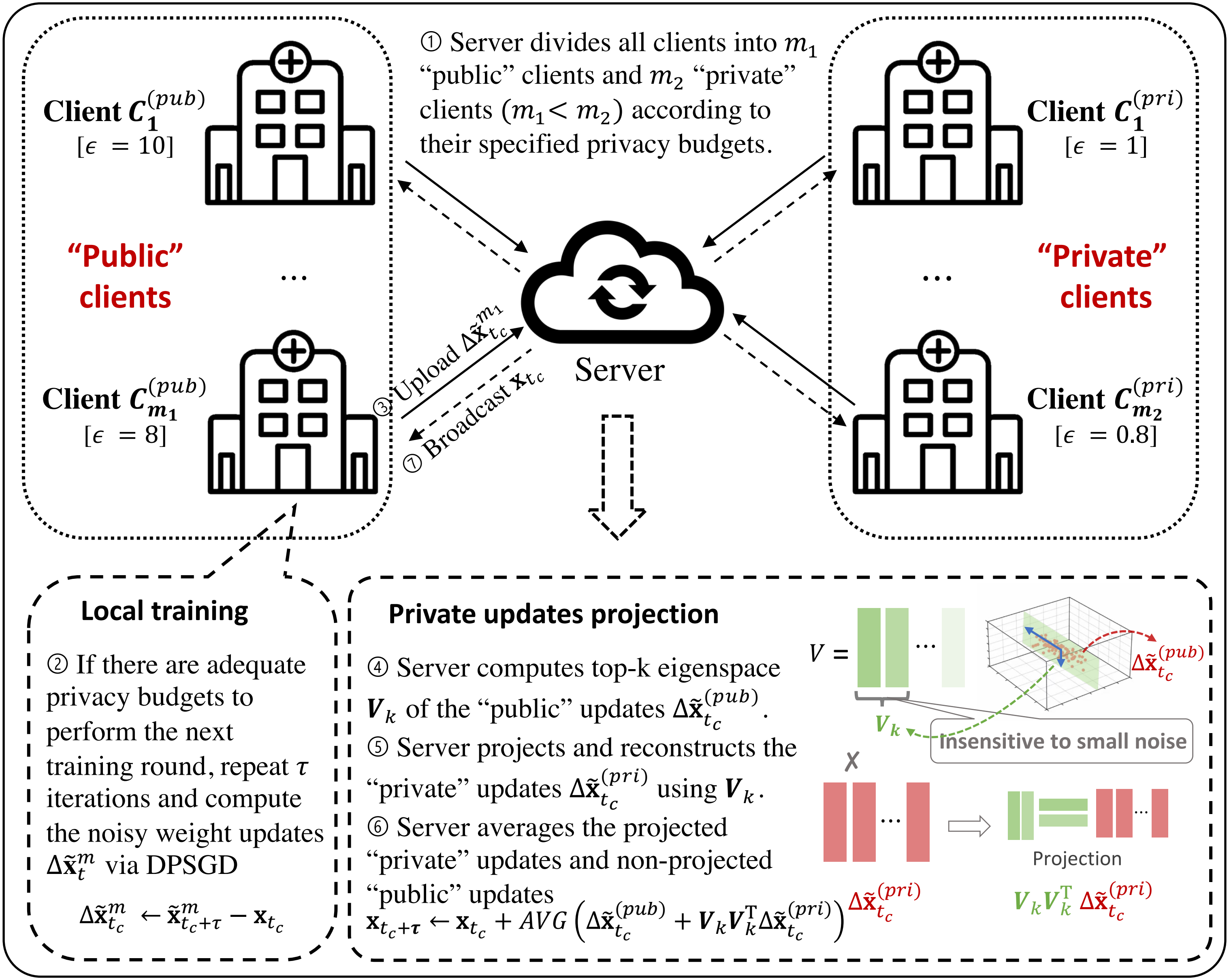
① Server divides all clients into m_1 "public" clients and m_2 "private" clients ($m_1 < m_2$) according to their specified privacy budgets.
Client $C_1^{(pub)}$ [ϵ = 10]
Client $C_{m_1}^{(pub)}$ [ϵ = 8]
"Public" clients
Client $C_1^{(pri)}$ [ϵ = 1]
Client $C_{m_2}^{(pri)}$ [ϵ = 0.8]
"Private" clients
Server
③ Upload $\Delta\tilde{\mathbf{x}}_{t_c}^{m_1}$
⑦ Broadcast $\mathbf{x}_{t_c}$
Local training
② If there are adequate privacy budgets to perform the next training round, repeat τ iterations and compute the noisy weight updates $\Delta\tilde{\mathbf{x}}_t^m$ via DPSGD
$\Delta\tilde{\mathbf{x}}_{t_c}^m \leftarrow \tilde{\mathbf{x}}_{t_c+\tau}^m - \mathbf{x}_{t_c}$
Private updates projection
④ Server computes top-k eigenspace $\boldsymbol{V}_k$ of the "public" updates $\Delta\tilde{\mathbf{x}}_{t_c}^{(pub)}$.
⑤ Server projects and reconstructs the "private" updates $\Delta\tilde{\mathbf{x}}_{t_c}^{(pri)}$ using $\boldsymbol{V}_k$.
⑥ Server averages the projected "private" updates and non-projected "public" updates
$\mathbf{x}_{t_c+\tau} \leftarrow \mathbf{x}_{t_c} + AVG\left(\Delta\tilde{\mathbf{x}}_{t_c}^{(pub)} + \boldsymbol{V}_k\boldsymbol{V}_k^{\mathrm{T}}\Delta\tilde{\mathbf{x}}_{t_c}^{(pri)}\right)$
V =
$\Delta\tilde{\mathbf{x}}_{t_c}^{(pub)}$
$\boldsymbol{V}_k$
Insensitive to small noise
$\Delta\tilde{\mathbf{x}}_{t_c}^{(pri)}$
Projection
$\boldsymbol{V}_k\boldsymbol{V}_k^{\mathrm{T}}\ \Delta\tilde{\mathbf{x}}_{t_c}^{(pri)}$