```python
import numpy as np

def calculate_W_by_svd(A,B):
    # Perform SVD of A
    u,s,vh = np.linalg.svd(A)

    # Find rank of A
    K = max(A.shape[0], A.shape[1])
    r = 0;
    while( r < A.shape[1] and abs(s[r]) >= abs(K*1e-6*s[0]) ):
        r = r+1;

    # Find least square solution
    v = vh.transpose()
    W = np.zeros((A.shape[1], B.shape[1]))
    for i in range(r):
        tmp = ((u[:,i].transpose() @ B)/s[i])
        for j in range(B.shape[1]):
            W[:,j] += tmp[j] * v[:,i]

    return W

def calculate_W_by_normal(A,B):
    W = np.linalg.inv(A.transpose() @ A) @ A.transpose() @ B
    return W
```