1. $X = U \Sigma V^T$

To PROVE:

$$\max_{v : \|v\| = 1} \|Xv\| \leq \sigma_1$$

Let us write v in the basis of the columns of V as $v = \sum_i \alpha_i v_i$. $\|v\| = 1$ and $\|v_i\| = 1 \; \forall i$,

so, $\alpha_i \leq 1$

We can write X as

$$X = \sum_{l=1}^{r} \sigma_l u_l v_l^T$$

$$Xv = \sum_{l=1}^{r} \sigma_l u_l v_l^T \sum_i \alpha_i v_i$$

$$= \sum_{l=1}^{r} \sigma_l u_l \sum_{i=1}^{r} \alpha_i \underbrace{v_l^T v_i}_{\substack{0 \text{ if } l \neq i \\ 1 \text{ if } l = i}}$$

$$= \sum_{l=1}^{r} \sigma_l \alpha_l u_l$$

$\hookrightarrow$ orthonormal vectors

$$\|Xv\|^2 = \|\sum_{l=1}^{r} \sigma_l \alpha_l\|^2$$

$$\leq \sigma_1^2$$ (weighted average is always less than or equal to the max value)

$$\|Xv\| \leq \sigma_1$$

$$\max_{v:\|v\|=1} \|Xv\| \leq \sigma_1 \quad //$$

2. To Prove:

For any x, the span of the first k right singular vectors gives a solution to the best-fit subspace problem for dimension k.

We can prove this by Induction:

1. Base Condition:

For any x, the span of the first two right singular vectors gives a solution to the best-fit subspace problem for dimension 2.

2. Assume that the span of the first $(k-1)$ right singular vectors gives a solution to the best-fit subspace problem for dimension $k-1$.

i.e. span of first $k-1$ right singular vectors maximizes $\underset{\dim(S)=k-1}{\text{var}(S;x)}$

3. To prove: span of first k right singular vectors maximizes $\underset{\dim(S)=k}{\text{var}(S;x)}$

$$\text{var}(S; X) = \sum_{i=1}^{k} \|X \cdot v_i\|^2$$

Assume the best-fit subspace to be S^*

$$\text{var}(S^*; X) = \sum_{i=1}^{k} \|X \cdot w_i\|^2$$, we w_i are orthonormal bases for S^*.

We know that the first $(k-1)$ right singular vectors had the maximum variance across any possible $(k-1)$ orthonormal vectors.

So $$\sum_{i=1}^{k-1} \|X \cdot v_i\|^2 \geq \sum_{i=1}^{k-1} \|X \cdot w_i\|^2 \quad - (1)$$

CLAIM: We can choose the orthonormal basis $\{w_1, w_2 \dots w_k\}$ for S^* such that w_k is perpendicular to the subspace $\bar{S} = \text{span}\{v_1, v_2 \dots v_{k-1}\}$

For the given subspace $\bar{S}$, we know that v_k is given by

$$v_k \doteq \arg\max_{\substack{\|v\| = 1 \\ v \perp \bar{S}}} \|X \cdot v\|$$

So, as w_k is also perpendicular to $\bar{S}$

$$\|X \cdot v_k\| \geq \|X \cdot w_k\|$$

$$\|X \cdot v_k\|^2 \geq \|X \cdot w_k\|^2 \quad - (2)$$

(1) + (2)

$$\sum_{i=1}^{k} \|X \cdot v_i\|^2 \geq \sum_{i=1}^{k} \|X \cdot w_i\|^2$$

i.e. span of first k right singular vectors maximizes var(S ; X)

dim(S) = k //

3. 1. Find largest Right Singular Vector of $Y = X^T X$ using Power Iteration. Let it be v_1.

2. Find $\sigma_1 = \|Y \cdot v_1\|$ or $\sigma_1 = \|X \cdot v_1\|$

3. $\bar{Y} = Y - \sigma_1 I$ and $\bar{Y} = Y - \sigma_1^2 I$

4. Now find the largest Right Singular Vector of $\bar{Y}$ using Power Iteration. Largest Right Singular Vector of $\bar{Y}$ is the Smallest right singular vector of Y (and hence X)

Power Iteration given Y

- Initialize $\bar{v}_0$ as $\begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$

- For $l = 1, \dots t$
  - $\bar{v}_l = Y \cdot \bar{v}_{l-1}$
  - $\bar{v}_l = \dfrac{\bar{v}_l}{\|\bar{v}_l\|}$

- Output $\bar{v}_t$

4. a. $L(y) = \sum_{(i,j)\in O} (x_{ij} - y_{ij})^2$

$$\nabla L(y) = -2 \sum_{(i,j)\in O} (x_{ij} - y_{ij})$$

$$\nabla L(y)_{i,j} = -2 (x_{ij} - y_{ij}) \cdot O_{ij}$$

$$\nabla L(y) = -2 (x - y) \cdot O$$

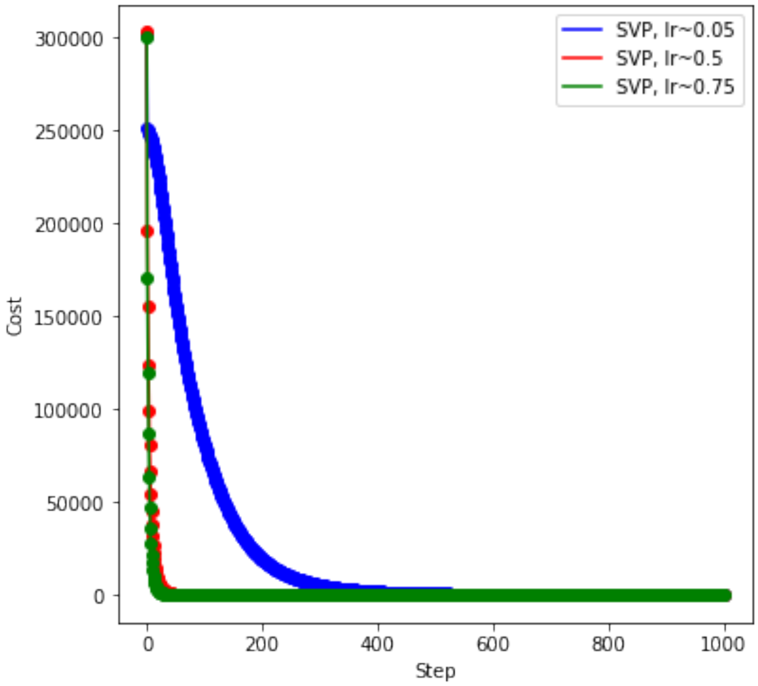
b. a.

```
def SVP(X, O, lr, T=1000):
    k = 5
    n = 1000; d = 500;
    X_pred = np.random.normal(0, 1, (n, d))
    costs = []

    for i in range(T):
        c = cost(X, X_pred, O)
        Y = X_pred - lr * gradient(X, X_pred, O)
        utrue, strue, vtrue = scipy.sparse.linalg.svds(Y, k = 5)
        X_pred = utrue @ np.diag(strue) @ vtrue
        costs.append(c)

    return np.array(costs)
```

b.

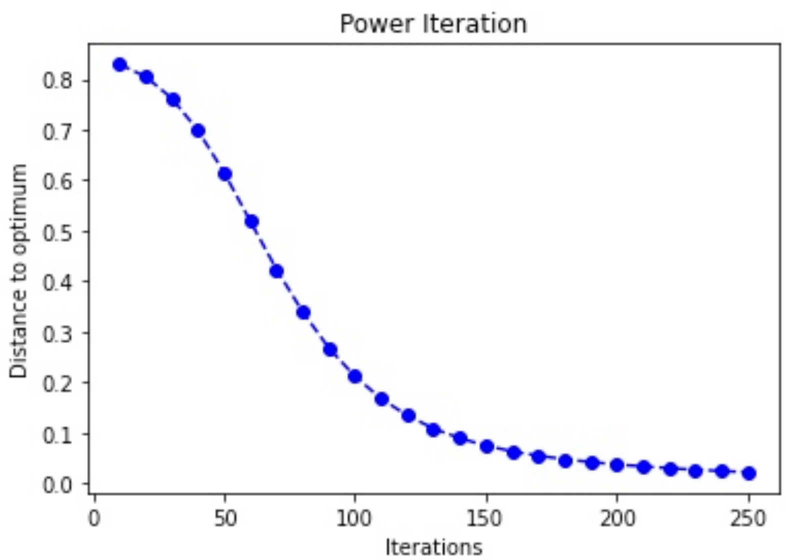


SVP, lr~0.05
SVP, lr~0.5
SVP, lr~0.75
300000
250000
200000
150000
100000
50000
0
Cost
0
200
400
600
800
1000
Step

5. a.

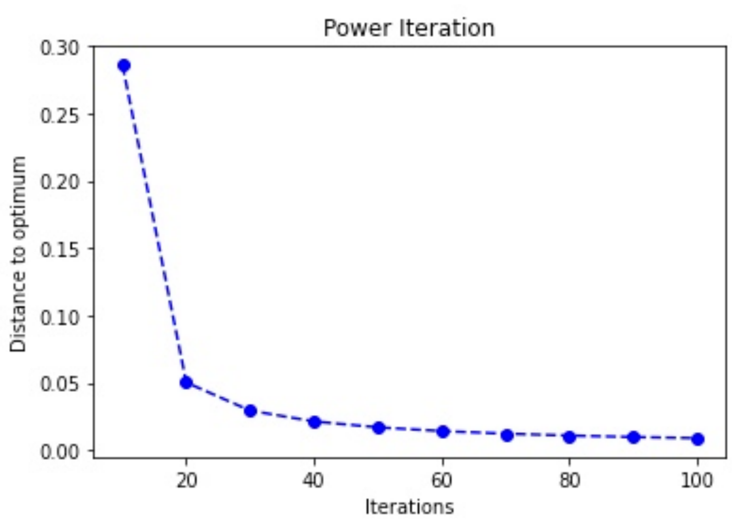
```
T        Time
In-built 6.122656726837159
10       0.611600136756897
20       1.2323015213012696
30       1.834072232246399
40       2.435634708404541
50       3.0591841459274294
60       3.6913986682891844
70       4.280641365051269
80       4.8633181095123295
90       5.455540823936462
100      6.042231750488281
```

b. We average across 10 runs. Multiple runs provide different graphs.



We can converge faster by controlling the gap between first and second singular values.

$\alpha = 0.07$



Distance to optimum for $T = 1000$:
(single run)