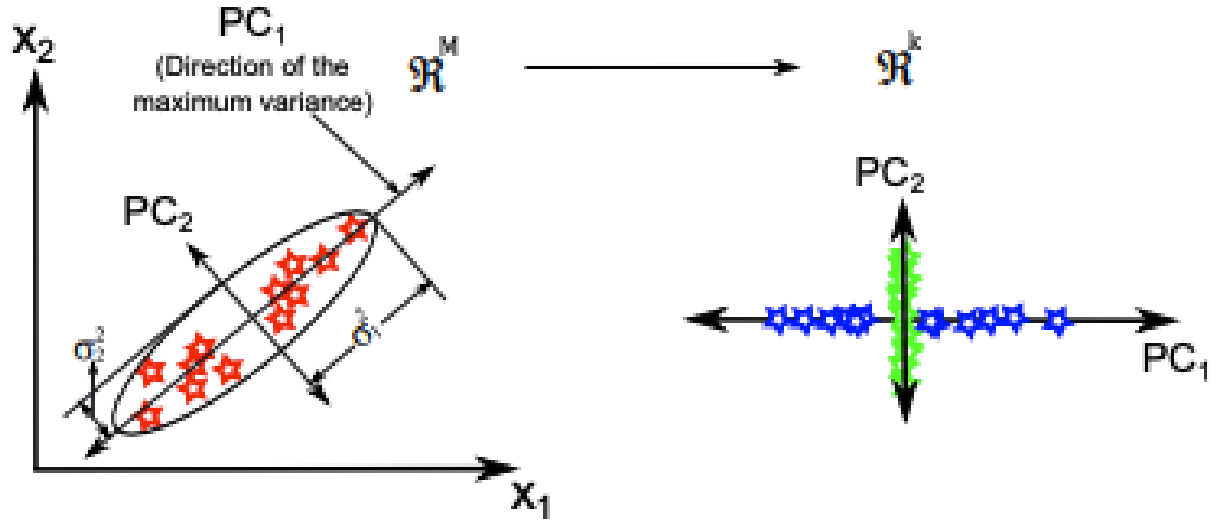
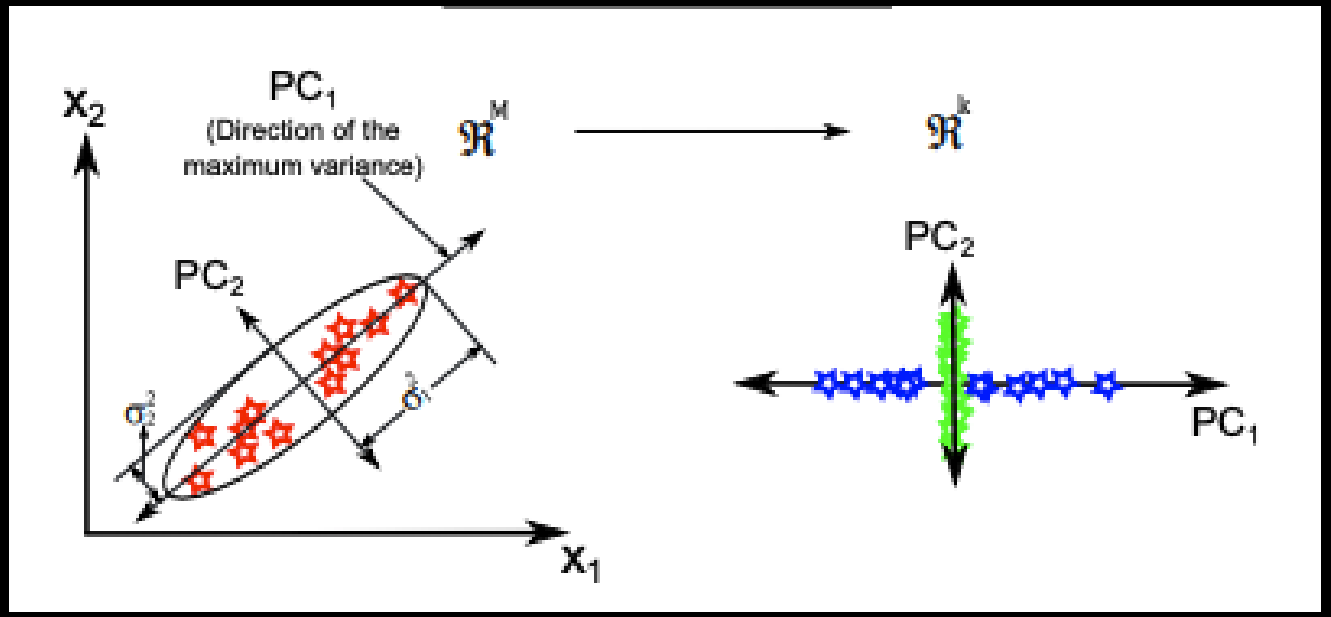
SINGULAR VALUE DECOMPOSITION IN PRINCIPAL COMPONENT ANALYSIS

Aidan Olson



MOTIVATION BEHIND PCA

1. Extract important information from data set
    1. Variance
    2. Primary Components
2. Compress dataset size while retaining essential features
3. Simplify data by reducing dimensions
4. Identify correlations between variables
    1. Covariance

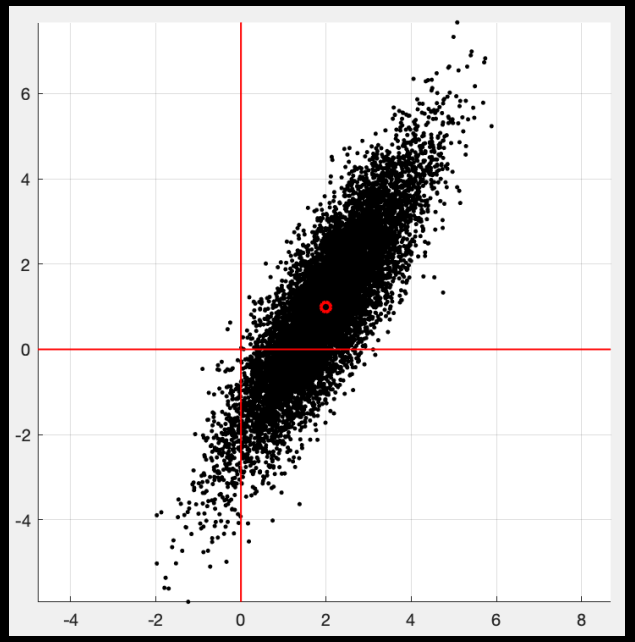


STATISTICAL APPROACH TO PCA

1. Center data around mean
2. Compute Covariance Matrix
3. Eigen system of Covariance matrix
    1. $CV = V\Lambda$
    2. Eigenvalues represent variances along principal axes
    3. Eigenvectors represent directions of principal axes
4. Extract Principal Components and Interpret Results
    1. $T = BV$ are projections of data onto principal axes
    2. Diagonal matrix Λ stores variance explained by each PC

STATISTICAL APPROACH TO PCA

1. Center data around mean
2. Compute Covariance Matrix
3. Eigen system of Covariance matrix
   1. $CV = V\Lambda$
   2. Eigenvalues represent variances along principal axes
   3. Eigenvectors represent directions of principal axes
4. Extract Principal Components and Interpret Results
   1. $T = BV$ are projections of data onto principal axes
   2. Diagonal matrix Λ stores variance explained by each PC



$A \in \mathbb{R}^{2\times m}$ is a collection of m, 2-D observations:

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,m} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,m} \end{bmatrix}.$$

$$\mu = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \quad \mu_1 = \frac{1}{m}\sum_{j=1}^{m} a_{1,j}, \quad \mu_2 = \frac{1}{m}\sum_{j=1}^{m} a_{2,j}.$$

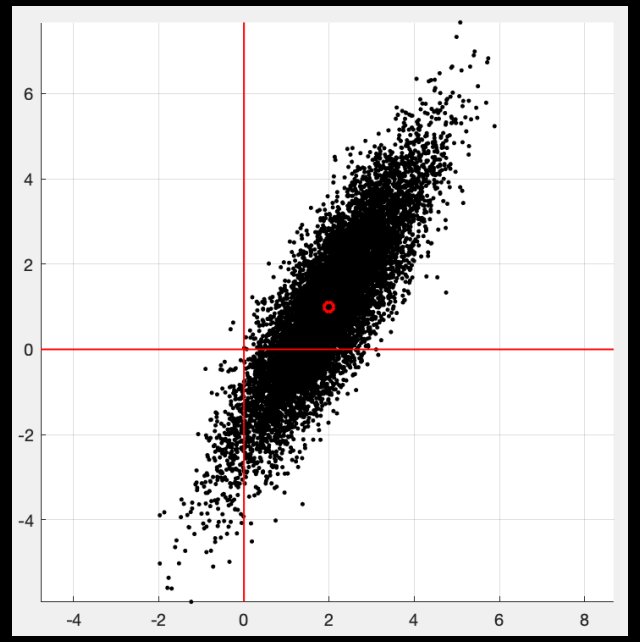
The Centered Data matrix B is computed as

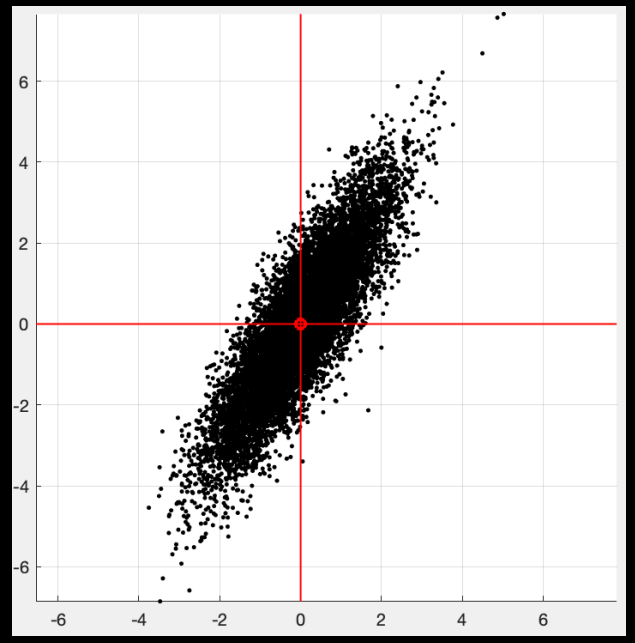
$$B = A - \mu[1, \ldots, 1],$$

and has mean $[0, 0]$

STATISTICAL APPROACH TO PCA

1. Center data around mean
2. Compute Covariance Matrix
3. Eigen system of Covariance matrix
    1. $CV = V\Lambda$
    2. Eigenvalues represent variances along principal axes
    3. Eigenvectors represent directions of principal axes
4. Extract Principal Components and Interpret Results
    1. $T = BV$ are projections of data onto principal axes
    2. Diagonal matrix Λ stores variance explained by each PC





STATISTICAL APPROACH TO PCA

1. Center data around mean
2. Compute Covariance Matrix
3. Eigen system of Covariance matrix
   1. $CV = V\Lambda$
   2. Eigenvalues represent variances along principal axes
   3. Eigenvectors represent directions of principal axes
4. Extract Principal Components and Interpret Results
   1. $T = BV$ are projections of data onto principal axes
   2. Diagonal matrix Λ stores variance explained by each PC

Covariance

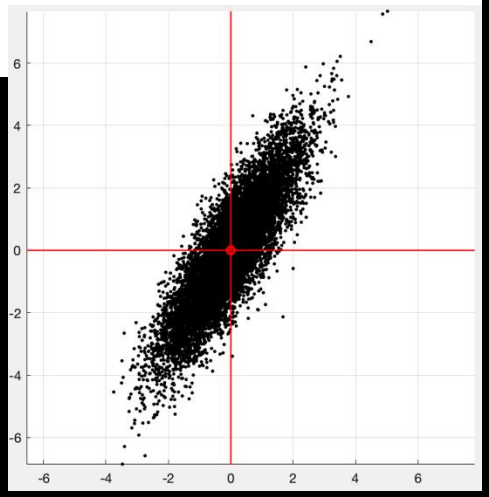
Let $B \in \mathbb{R}^{n \times m}$ be a mean centered data matrix. The covariance matrix is given by:

$$C = \frac{1}{n} B^T B,$$

The covariance matrix of a dataset reveals the relationships between variables in a dataset, showing whether they tend to move in the same direction or in opposite directions.
Each item in C describes the pairwise covariance between two variables. C is symmetric.

- High magnitude implies a strong correlation, while low magnitude implies a weak correlation.
- Positive covariance implies direct correlation.
- The main diagonal is the variance of each row.

In our 2 dimensional example, $C \in \mathbb{R}^{2 \times 2}$.



STATISTICAL APPROACH TO PCA

1. Center data around mean
2. Compute Covariance Matrix
3. Eigen system of Covariance matrix
   1. CV = VΛ
   2. Eigenvalues represent variances along principal axes
   3. Eigenvectors represent directions of principal axes
4. Extract Principal Components and Interpret Results
   1. T = BV are projections of data onto principal axes
   2. Diagonal matrix Λ stores variance explained by each PC
   3. Econ reduced V to n x m for m < n

Eigen System

The eigensystem of our covariance matrix reveals the principal components (dimensions of the highest variance) and the magnitudes of variance in each dimension.

$$C = VDV^T,$$

where:

$$V = \begin{bmatrix} v_{1,1} & v_{1,2} & \cdots & v_{1,n} \\ v_{2,1} & v_{2,2} & \cdots & v_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ v_{n,1} & v_{n,2} & \cdots & v_{n,n} \end{bmatrix}, \quad D = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix}.$$

- The decomposition $C = VDV^T$ separates the variability of the data into orthogonal directions (eigenvectors) and quantifies their significance (eigenvalues).
- The elements of D are ordered in descending magnitude.
- The vector v_1 always points in the direction of largest variance. All the following v_i for $1 < i \leq n$ are computed to point in the next largest direction of variance, but must also be orthogonal to v_j, $1 \leq j < i$.

STATISTICAL APPROACH TO PCA

1. Center data around mean
2. Compute Covariance Matrix
3. Eigen system of Covariance matrix
   1. CV =VΛ
   2. Eigenvalues represent variances along principal axes
   3. Eigenvectors represent directions of principal axes
4. Extract Principal Components and Interpret Results
   1. T = BV are projections of data onto principal axes
   2. Diagonal matrix Λ stores variance explained by each PC

Principal Components of X:

$$P_{X_j} = \sigma_i u_i + \mu.$$

Principal Components

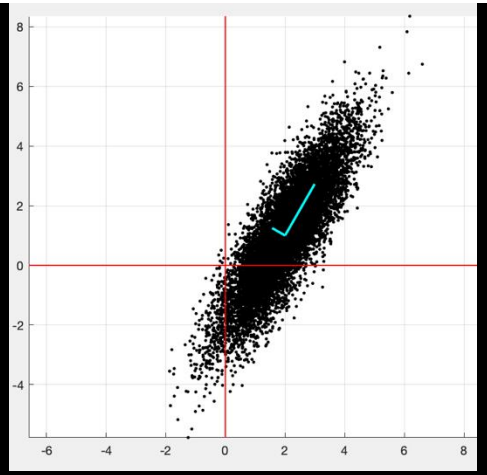
The vectors in V are called "loadings", which, when applied to B, yield the principal components:

$$P = BV$$

Notice that since B can be written as its Singular Value Decomposition, we have:

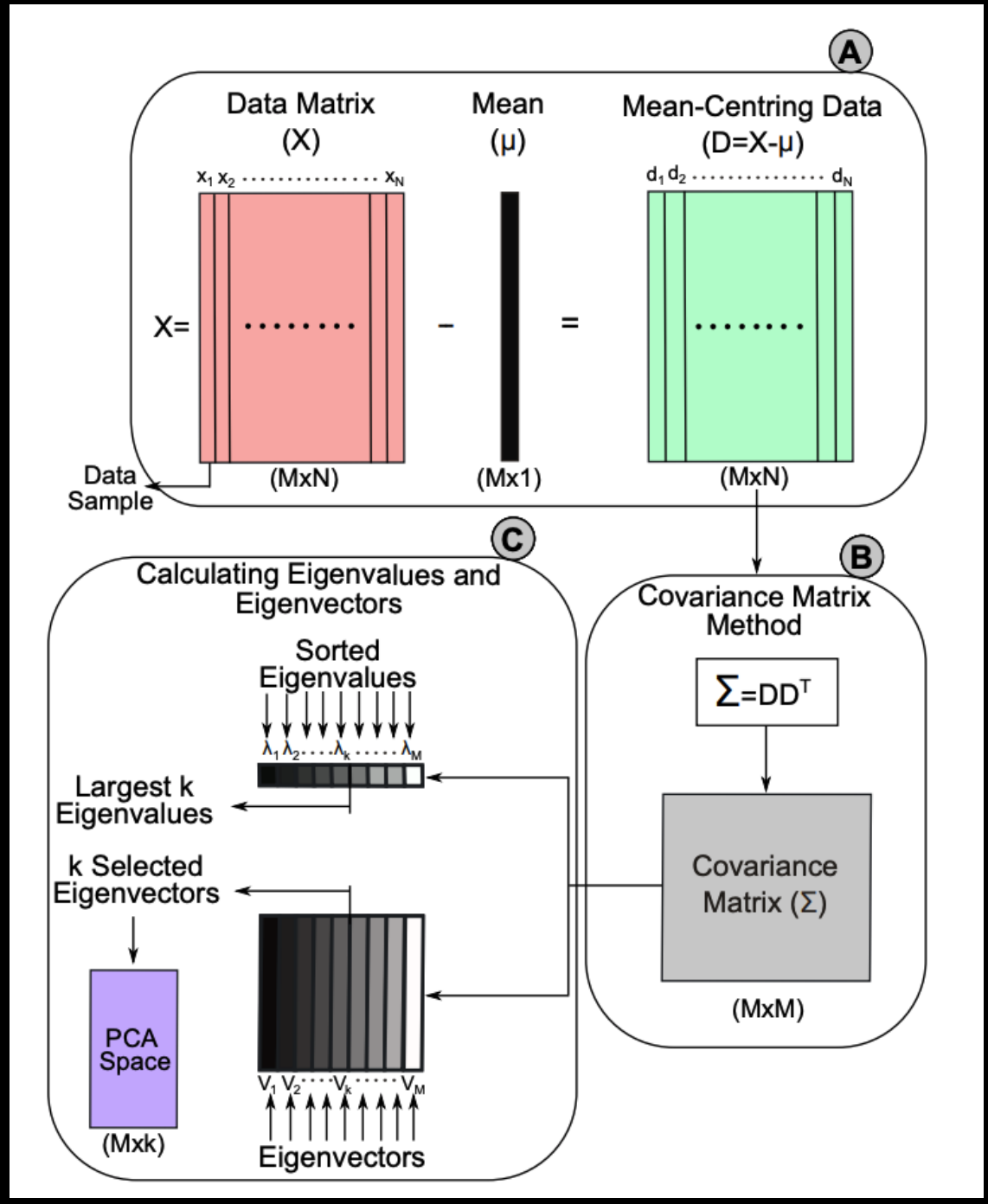
$$B = U\Sigma V^T$$
$$BV = U\Sigma = P.$$

Principal components and loading can be directly extracted from the SVD.



STATISTICAL APPROACH TO PCA

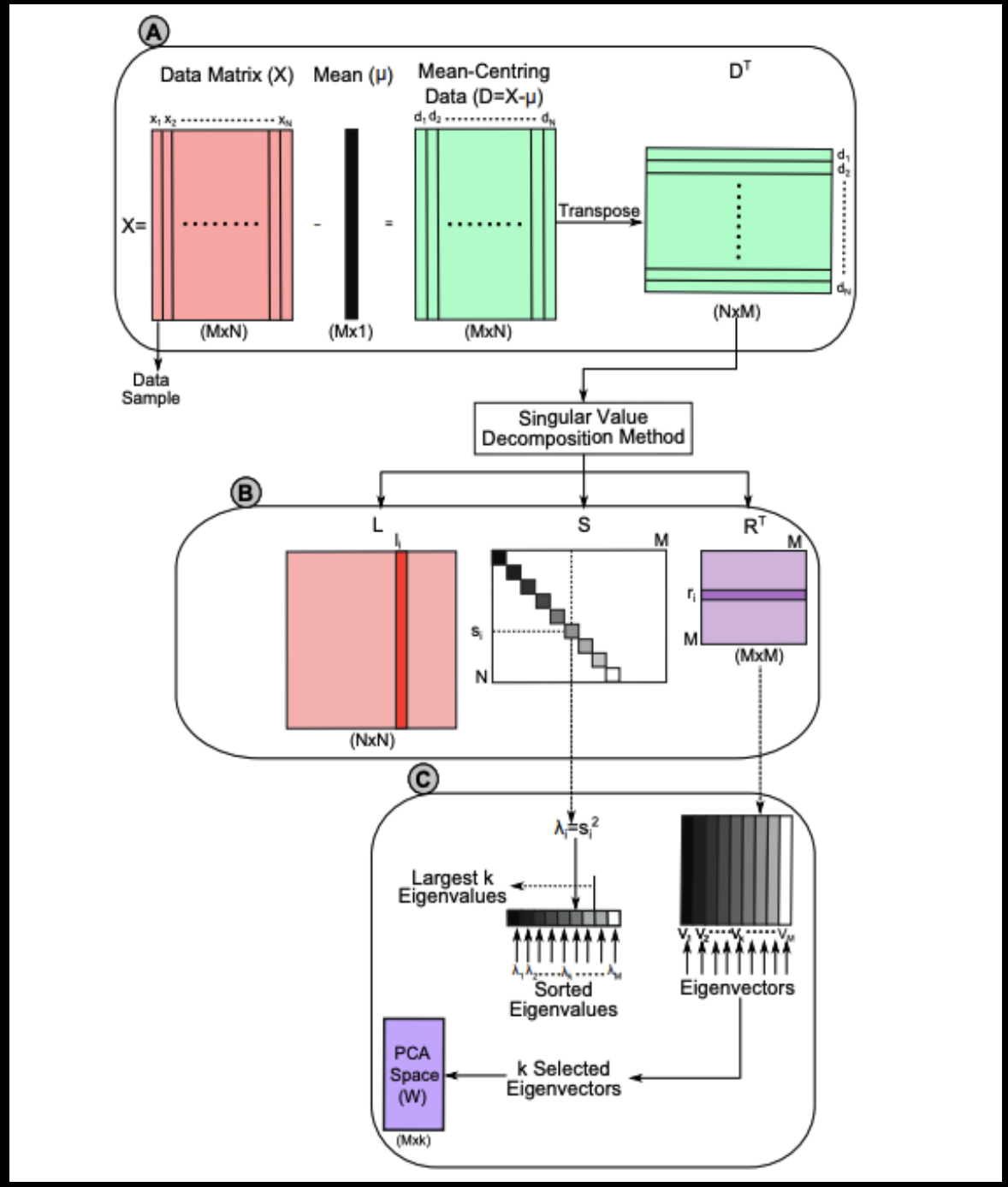
1. Center data around mean
2. Compute Covariance Matrix
3. Eigen system of Covariance matrix
    1. $CV = V\Lambda$
    2. Eigenvalues represent variances along principal axes
    3. Eigenvectors represent directions of principal axes
4. Extract Principal Components and Interpret Results
    1. $T = BV$ are projections of data onto principal axes
    2. Diagonal matrix Λ stores variance explained by each PC



EMERGENCE OF THE SVD

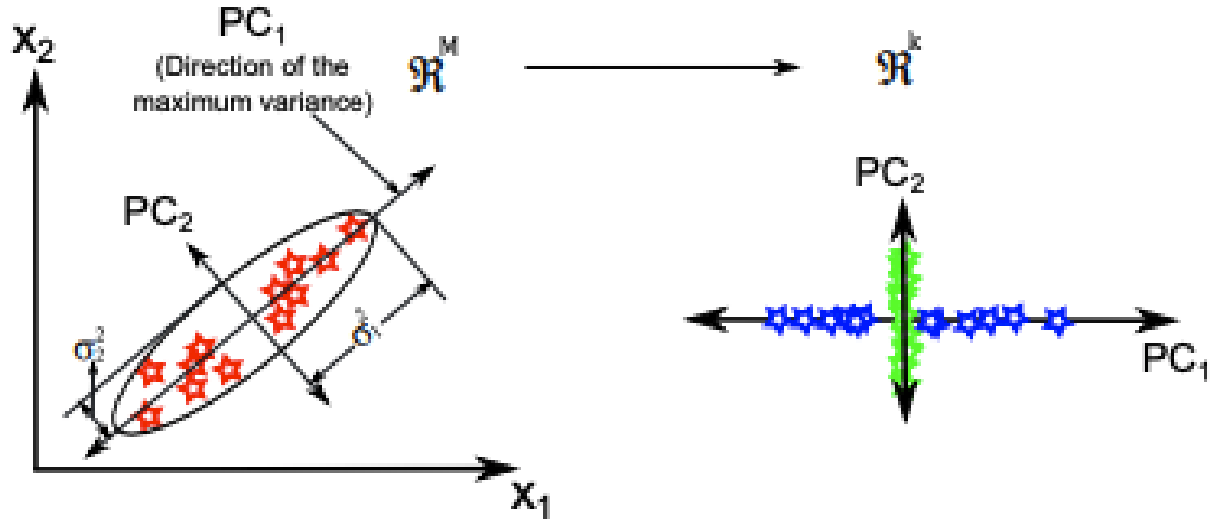
1. Expedited approach
2. Computationally efficient
3. Stable

Note*



SINGULAR VALUE DECOMPOSITION IN PRINCIPAL COMPONENT ANALYSIS

The point of PCA is to reduce dimensions of a data matrix. The approach is to project data to a smaller dimension k. When we conduct a PCA from 2D into 2D…



$$\min_{\Phi \in \mathcal{O}_{d,k}} \sum_{t=1}^{n} \|\mathbf{x}_t - \Phi y_t\|_2^2.$$

EXAMPLE

Part II: Linking Genome to Ovarian Cancer

GOAL

1. Can we predict a patient's susceptibility to ovarian cancer given a set of 4000 numerical gene expressions?
2. Perhaps there are some "eigen genomes" that capture variance in patient genomes and relate to a risk of developing ovarian cancer.
3. If we can identify some primary genomes, then we can project otherwise high dimensional data into a lower dimension.

$B \in \mathbb{R}^{216\times 4000}$ is a mean centered data matrix:

Patient	Gene 1	Gene 2	Gene 3	Gene 4	Gene 5	Gene 6	Gene 7	Gene 8	Gene 9
1	0.06391536	0.03324173	0.01848414	0.00861769	0.0356288	0.03792548	0.02886469	0.06173086	0.06310009
2	0.02540862	0.05108479	0.05630495	0.02173849	0.02740998	0.0149138	0.02245485	0.02395709	0.06052699
3	0.02553625	0.03612279	0.05419524	0.00973498	0.02752051	0.05225475	0.04281249	0.06908704	0.06987317
4	0.01281732	0.02965184	0.07928965	0.05067696	0.03973674	0.05771286	0.04449233	0.03458092	0.04258727
5	0.01984628	-0.0105772	-0.0075045	0.0190416	0.06878639	0.0617643	0.0390362	0.02044466	0.02598819
6	0.03904781	0.03935459	0.00134341	0.02622134	0.04409051	0.04395255	0.03962859	0.04792624	0.0468918
7	0.02319523	0.05382612	0.03695354	0.02155368	0.03882545	0.038917	0.05162363	0.03469103	0.01780947
8	0.02701748	0.01013702	0.01765632	0.01116838	0.03668468	0.01603467	0.00667543	0.02749022	0.02912227

	1	2	3	4	5	6	7	8	9	10
1	-6.0174e-04	-0.0010	5.1654e-04	-6.2815e-04	0.0027	-8.1287e-04	0.0014	0.0060	0.0041	-0.0017
2	-5.7528e-04	-8.2589e-04	9.1247e-04	-7.2196e-04	0.0031	-0.0016	0.0015	0.0080	0.0048	-9.4733e-04
3	-5.2042e-04	-0.0010	4.9837e-04	-4.8087e-04	0.0027	-0.0013	0.0018	0.0072	0.0047	-0.0027
4	-4.9379e-04	-9.9202e-04	6.5023e-04	4.9608e-04	0.0018	-0.0020	0.0016	0.0050	0.0044	-0.0031
5	-5.1443e-04	-0.0012	9.0853e-04	1.6537e-05	8.1084e-04	-0.0014	0.0015	0.0027	0.0031	-0.0027
6	-5.3182e-04	-0.0012	0.0012	-4.6326e-04	0.0016	-0.0017	4.3215e-04	0.0042	0.0047	-5.8042e-04
7	-6.3545e-04	-0.0015	1.0593e-04	-3.5955e-04	0.0021	-7.7052e-04	6.8566e-04	0.0057	0.0035	-0.0020
8	-8.6523e-04	-0.0023	3.4743e-04	3.4847e-04	0.0035	-4.9156e-04	6.0505e-04	0.0090	0.0046	-0.0048
9	-8.8558e-04	-0.0024	2.6131e-04	8.5273e-04	0.0030	-9.6942e-04	0.0010	0.0076	0.0040	-0.0066
10	-7.3392e-04	-0.0017	2.5494e-04	3.3011e-04	0.0025	-0.0010	0.0011	0.0063	0.0044	-0.0037
11	-5.5360e-04	-0.0012	1.7370e-04	3.4988e-04	0.0025	-0.0010	8.8894e-04	0.0043	0.0031	-0.0031
12	-5.1235e-04	-0.0011	2.4407e-04	-1.5715e-04	0.0022	-5.7886e-04	5.8352e-05	0.0036	0.0027	-0.0014
13	-5.9838e-04	-0.0011	8.4830e-04	-3.5652e-04	0.0026	-0.0014	0.0018	0.0037	0.0054	4.7675e-04
14	-6.0705e-04	-0.0013	3.2266e-04	-7.9253e-04	0.0028	-0.0016	0.0020	0.0058	0.0034	-0.0021
15	-6.0930e-04	-0.0016	8.1347e-04	-5.2983e-04	0.0027	-0.0015	1.7814e-04	0.0053	4.9288e-04	-0.0042
16	-6.0016e-04	-0.0018	0.0010	4.2147e-04	0.0016	-0.0023	-9.4344e-05	0.0059	0.0018	-0.0029
17	-6.7727e-04	-0.0014	4.6205e-04	8.3861e-04	0.0013	-0.0017	-9.6104e-04	0.0063	0.0040	-0.0026
18	-9.1597e-04	-0.0017	4.6425e-04	8.4291e-04	0.0032	-0.0020	6.4470e-04	0.0077	0.0056	-0.0034
19	-0.0011	-0.0024	-1.9841e-04	9.4578e-04	0.0042	-0.0016	7.6211e-05	0.0102	0.0042	-0.0022
20	-9.4418e-04	-0.0024	-1.8925e-04	8.8883e-05	0.0030	-7.2244e-04	5.9302e-04	0.0076	0.0025	-0.0013
21	-7.1030e-04	-0.0021	-1.1977e-04	1.0347e-04	0.0015	-0.0010	-4.6040e-04	0.0059	0.0024	-0.0033
22	-4.9405e-04	-0.0015	2.5002e-04	-1.4336e-05	0.0016	-1.3140e-04	8.9723e-04	0.0024	0.0018	-0.0019

$= V \in \mathbb{R}^{4000 \times 216}$

$\Sigma =$

	1	2	3	4	5	6	7	8	9	10	11	12
1	829.2802	0	0	0	0	0	0	0	0	0	0	0
2	0	103.0249	0	0	0	0	0	0	0	0	0	0
3	0	0	49.3769	0	0	0	0	0	0	0	0	0
4	0	0	0	39.2474	0	0	0	0	0	0	0	0
5	0	0	0	0	31.9608	0	0	0	0	0	0	0
6	0	0	0	0	0	27.3468	0	0	0	0	0	0
7	0	0	0	0	0	0	23.7796	0	0	0	0	0
8	0	0	0	0	0	0	0	20.0224	0	0	0	0
9	0	0	0	0	0	0	0	0	16.6298	0	0	0
10	0	0	0	0	0	0	0	0	0	14.7454	0	0
11	0	0	0	0	0	0	0	0	0	0	13.3579	0
12	0	0	0	0	0	0	0	0	0	0	0	11.9953

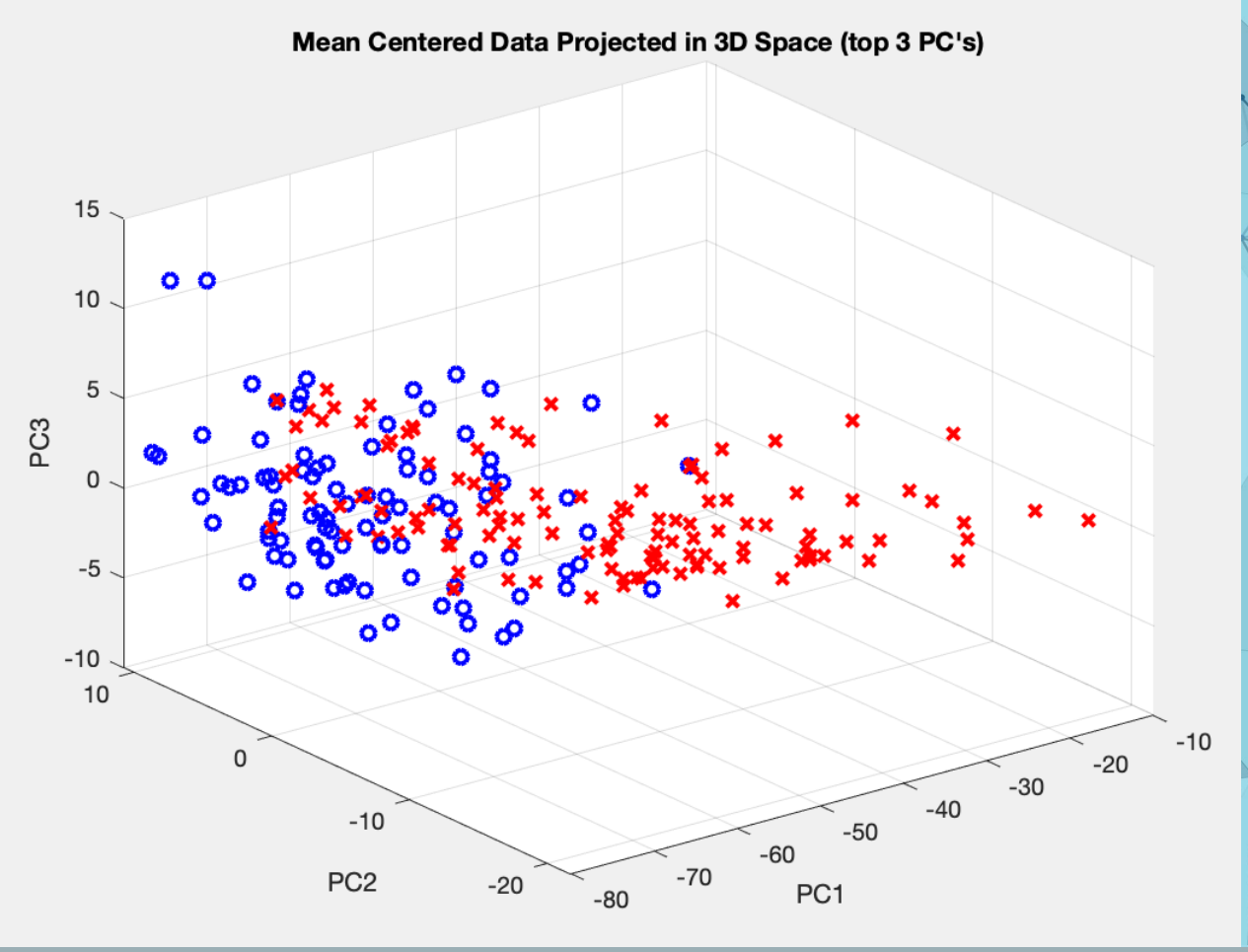
```
% obs = genetic data from 216 patients.
% grp = signifier of ovarian cancer. 1 means a patient has cancer.

[U,S,V] = svd(obs,'econ');

figure(2);
for i=1:size(obs,1)
    x = V(:,1)'*obs(i,:)';
    y = V(:,2)'*obs(i,:)';
    z = V(:,3)'*obs(i,:)';
    if(grp{i}=='Cancer')
        plot3(x,y,z,'rx','LineWidth',2); hold on;
    else
        plot3(x,y,z,'bo','LineWidth',2); hold on;
    end
end
```

Dimensional Reduction from 4000 to 3

The loading space V is size 4000 by 216. Each column is an "eigen genome" which expresses the direction of maximum variance. Most genomes can be expressed as a linear combination of a few unique genomes.
This code takes an inner product of the top three eigen genomes with each observation. This process transforms the original data into a more visualizable 3-dimensional space where we can observe clustering patterns



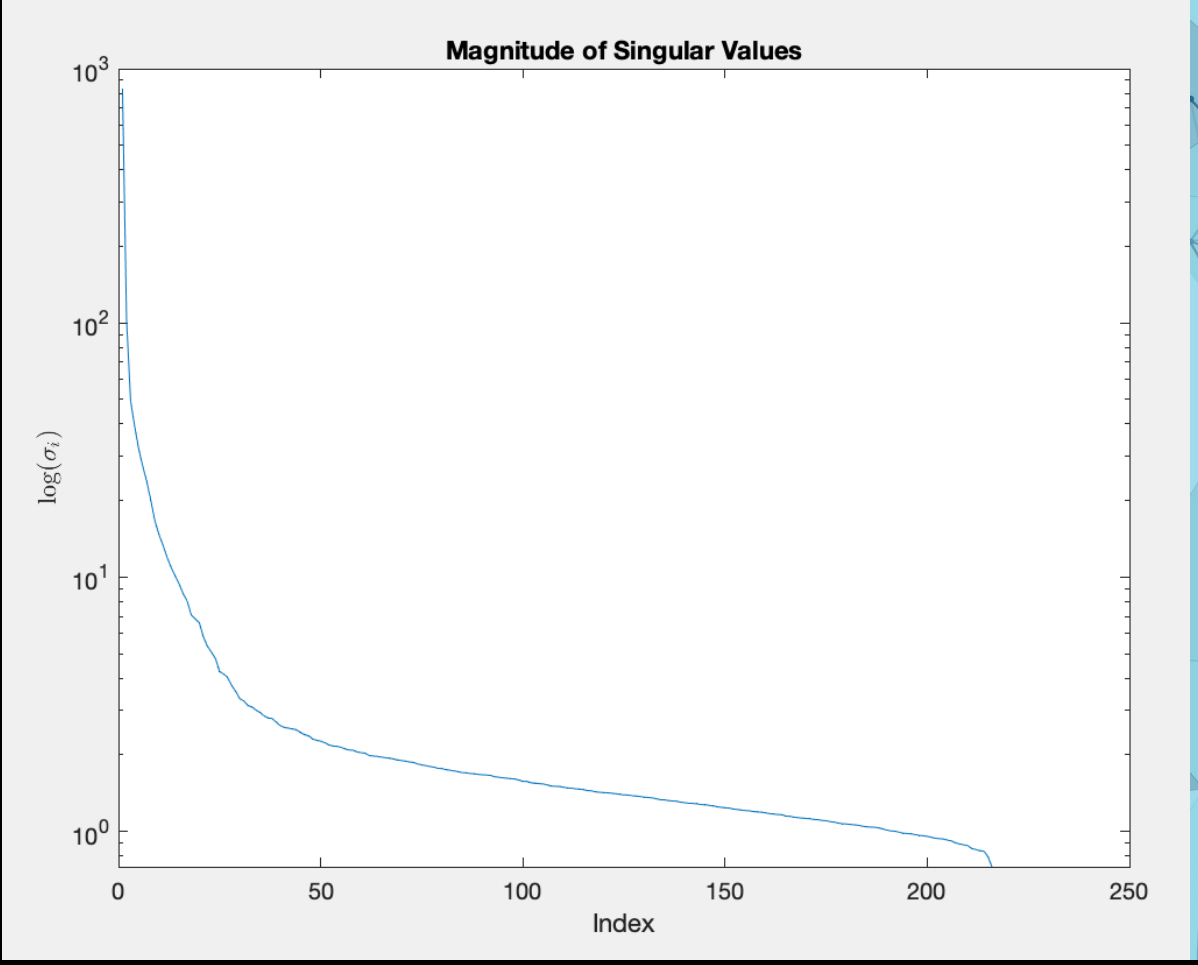
x = Cancer

o = No Cancer

Takeaway

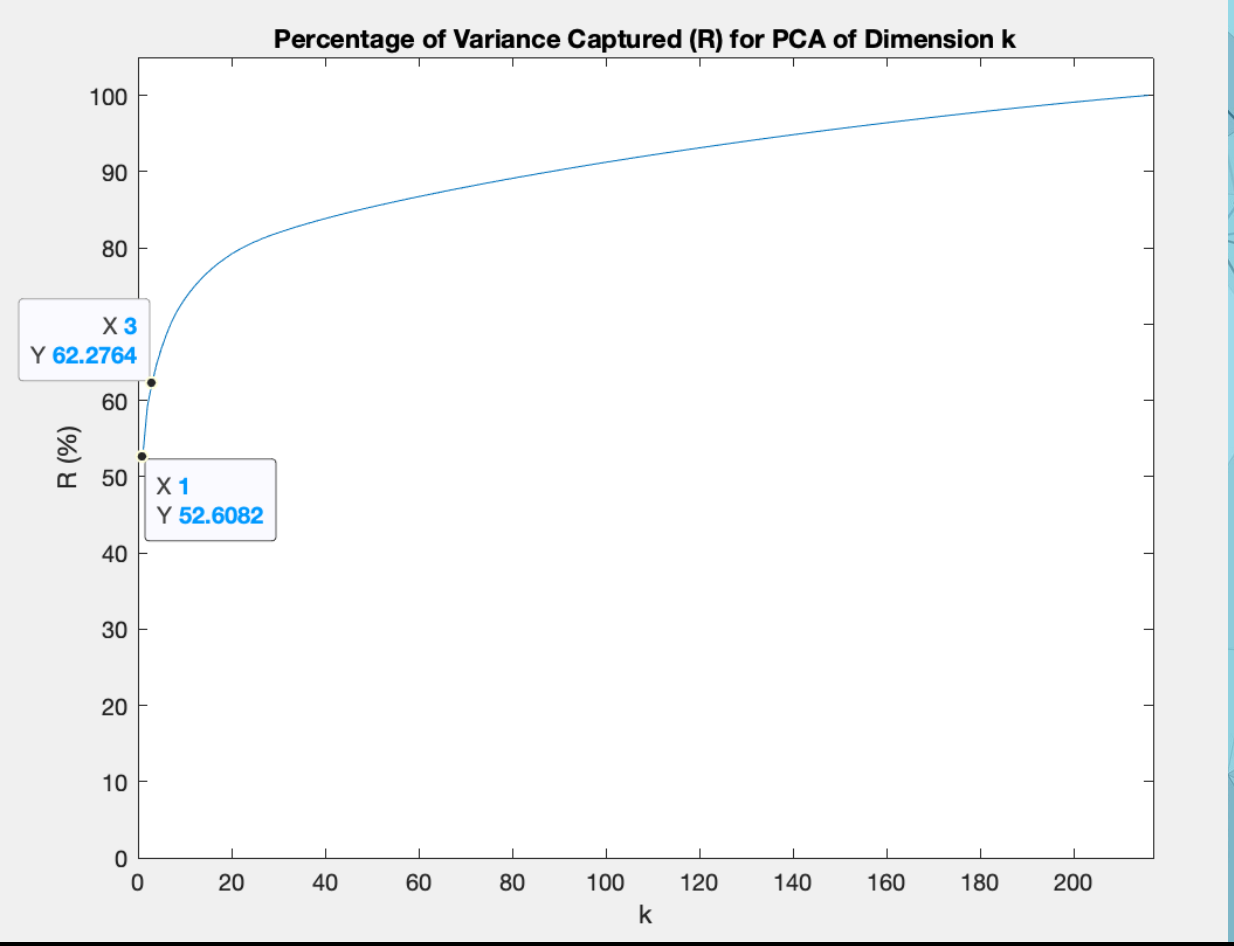
If we know the inner produce of a patient's genome with our three indicator genomes, then we can reasonably predict susceptibility to ovarian cancer.

Rather than checking 4000 genes expressions, the PCA allows us to check 3 inner products instead.



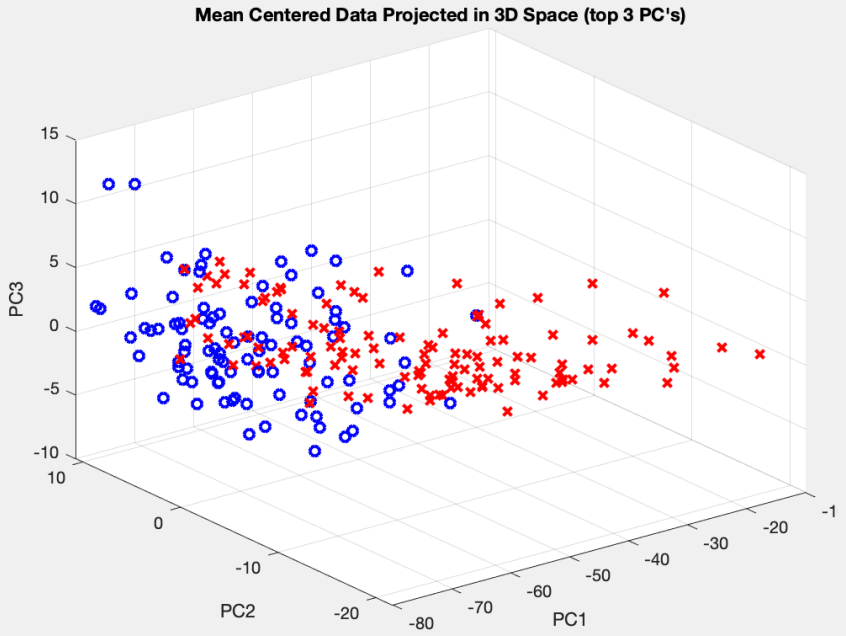
Singular Values

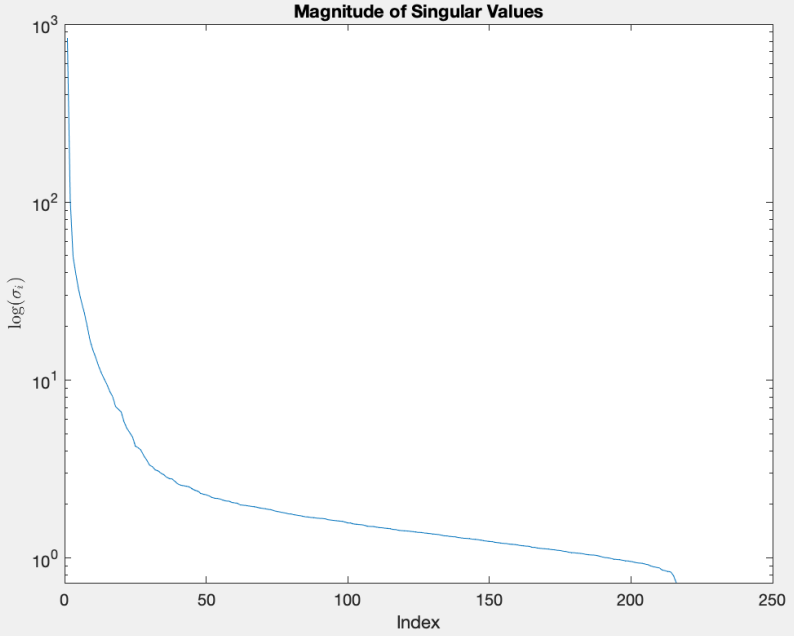
This rapid decay of our singular values indicates that our dimensional reduction still preserves a large portion of the original data, keeping mostly the important things. Can we quantify how much of the important data we are keeping?

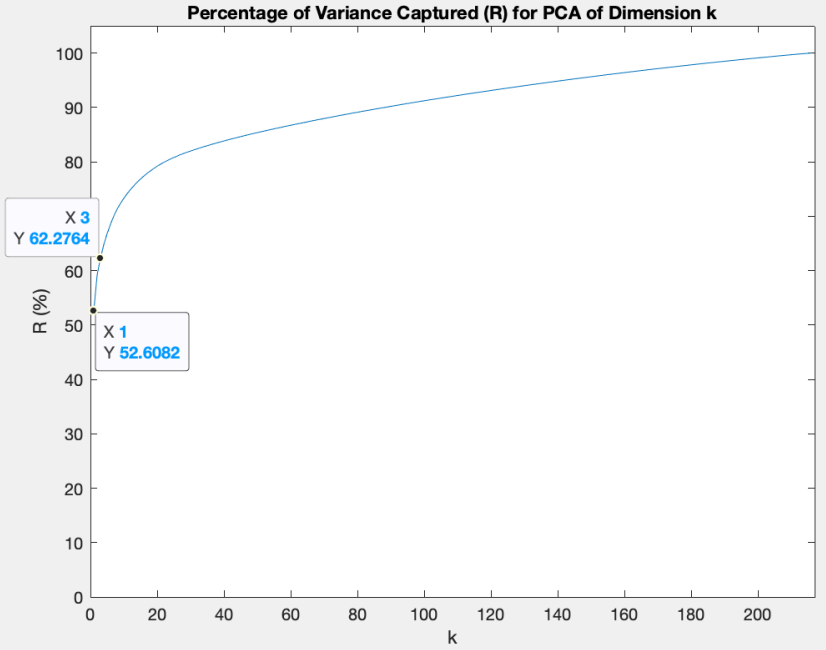


Robustness of PCA Space

$$R = \frac{\text{Total Variance of } W}{\text{Total Variance}} = \frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{M} \lambda_i}$$







Conclusion

Based on these three figures, we can say that this PCA dimension reduction provides a simple, accurate, and efficient way to represent a large data set.

If a priority of ours was to maximize accuracy at the expense of data visualizability, we could increase k beyond 3.

MORE EXAMPLES

Part III: Student Alcohol Consumption

&

Patient Diabetes / Health Data

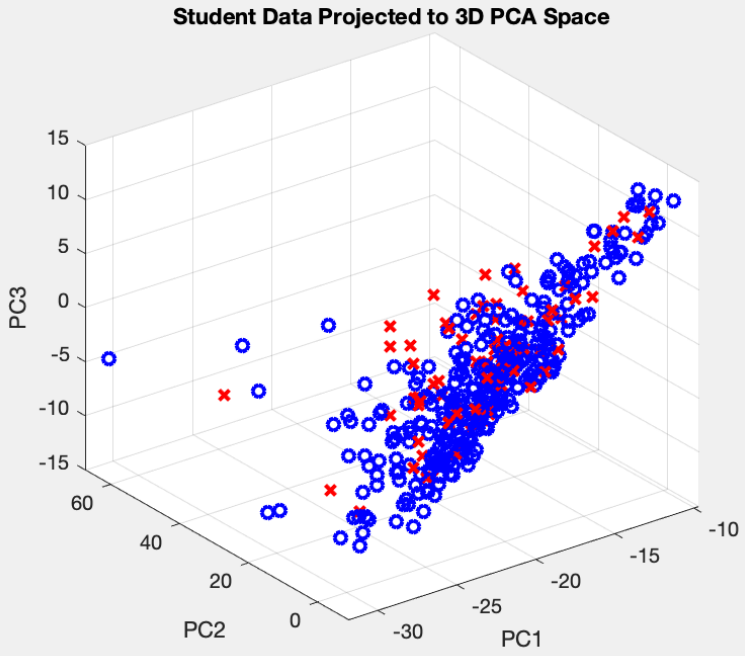
STUDENT ALCOHOL CONSUMPTION DATA

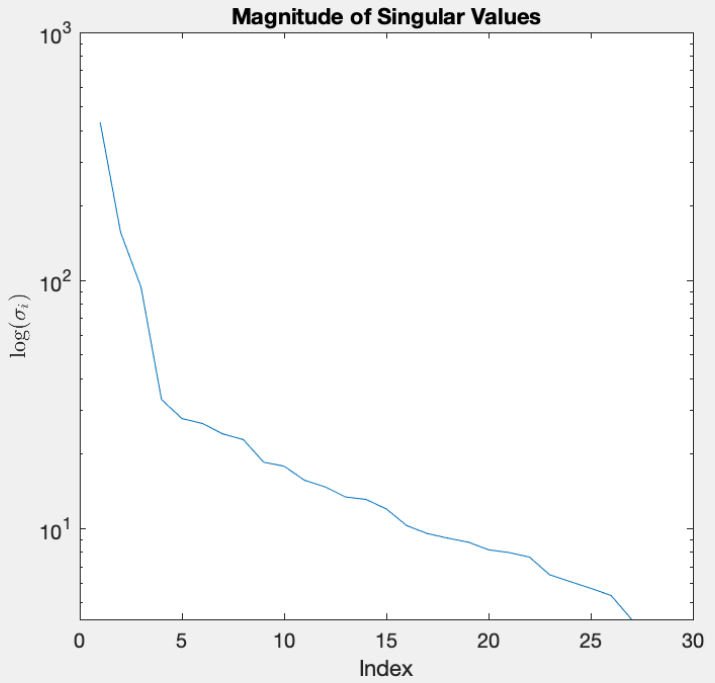
Question: What factors (if any) relate to a student's "Weekend Alcohol Consumption" quantity (from 1-5)?

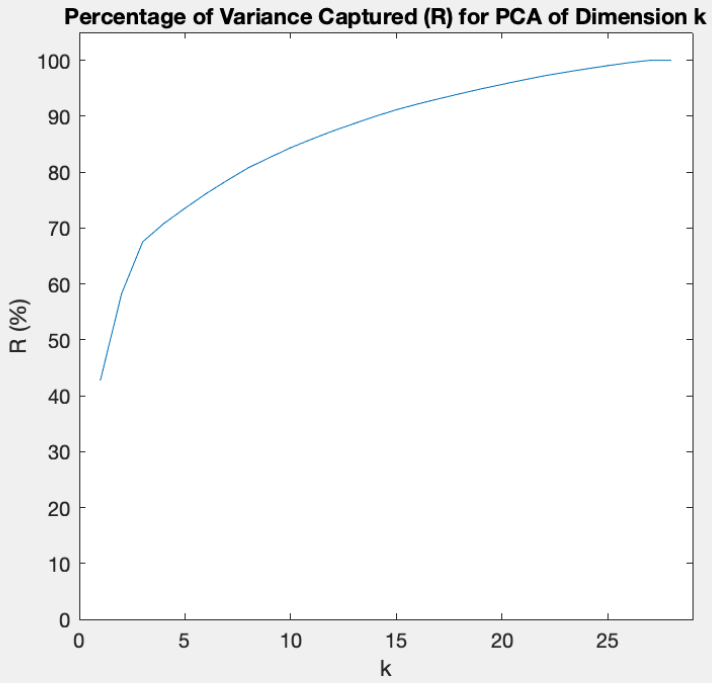
Procedure: Convert data to numeric, identify principal components, visualize results.

1. school - student's school (binary: 'GP' - Gabriel Pereira or 'MS' - Mousinho da Silveira)
2. sex - student's sex (binary: 'F' - female or 'M' - male)
3. age - student's age (numeric: from 15 to 22)
4. address - student's home address type (binary: 'U' - urban or 'R' - rural)
5. famsize - family size (binary: 'LE3' - less or equal to 3 or 'GT3' - greater than 3)
6. Pstatus - parent's cohabitation status (binary: 'T' - living together or 'A' - apart)
7. Medu - mother's education (numeric: 0 - none, 1 - primary education (4th grade), 2 – 5th to 9th grade, 3 – secondary education or 4 – higher education)
8. Fedu - father's education (numeric: 0 - none, 1 - primary education (4th grade), 2 – 5th to 9th grade, 3 – secondary education or 4 – higher education)
9. Mjob - mother's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g. administrative or police), 'at_home' or 'other')
10. Fjob - father's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g. administrative or police), 'at_home' or 'other')
11. reason - reason to choose this school (nominal: close to 'home', school 'reputation', 'course' preference or 'other')
12. guardian - student's guardian (nominal: 'mother', 'father' or 'other')
13. traveltime - home to school travel time (numeric: 1 - <15 min., 2 - 15 to 30 min., 3 - 30 min. to 1 hour, or 4 - >1 hour)
14. studytime - weekly study time (numeric: 1 - <2 hours, 2 - 2 to 5 hours, 3 - 5 to 10 hours, or 4 - >10 hours)
15. failures - number of past class failures (numeric: n if 1<=n<3, else 4)
16. schoolsup - extra educational support (binary: yes or no)
17. famsup - family educational support (binary: yes or no)
18. paid - extra paid classes within the course subject (Math or Portuguese) (binary: yes or no)
19. activities - extra-curricular activities (binary: yes or no)
20. nursery - attended nursery school (binary: yes or no)
21. higher - wants to take higher education (binary: yes or no)
22. internet - Internet access at home (binary: yes or no)
23. romantic - with a romantic relationship (binary: yes or no)
24. famrel - quality of family relationships (numeric: from 1 - very bad to 5 - excellent)
25. freetime - free time after school (numeric: from 1 - very low to 5 - very high)
26. goout - going out with friends (numeric: from 1 - very low to 5 - very high)
27. Dalc - workday alcohol consumption (numeric: from 1 - very low to 5 - very high)
28. Walc - weekend alcohol consumption (numeric: from 1 - very low to 5 - very high)
29. health - current health status (numeric: from 1 - very bad to 5 - very good)
30. absences - number of school absences (numeric: from 0 to 93)

STUDENT ALCOHOL CONSUMPTION PCA, K = 3







Weekend Alcohol Consumption (1 to 5):

- x > 3
- o ≤ 3

DIABETES AND HEALTH DATA

Question: Can a combination of these variables indicate the presence of diabetes?

Procedure: Convert data to numeric, identify principal components, visualize results.

Pregnancies: To express the Number of pregnancies

Glucose: To express the Glucose level in blood

BloodPressure: To express the Blood pressure measurement

SkinThickness: To express the thickness of the skin

Insulin: To express the Insulin level in blood

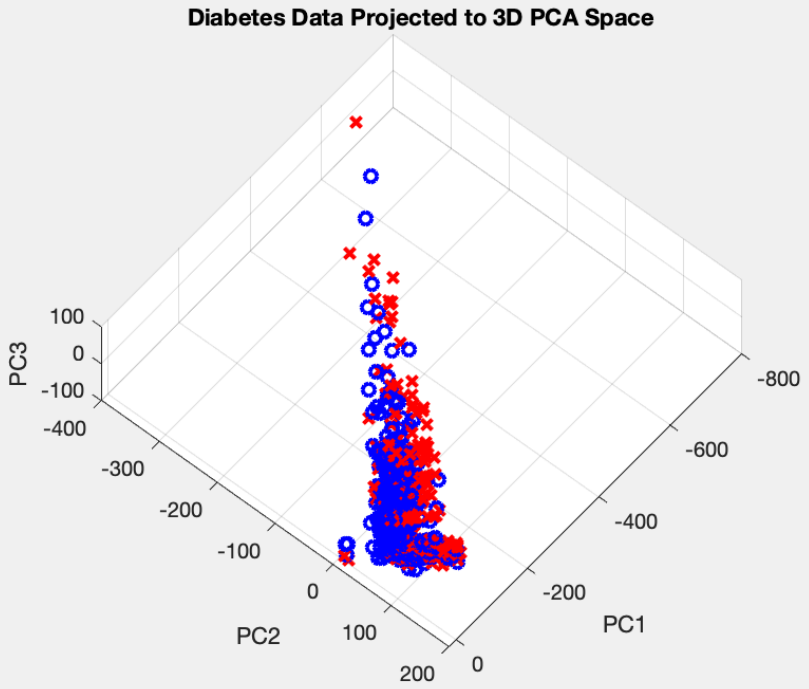
BMI: To express the Body mass index

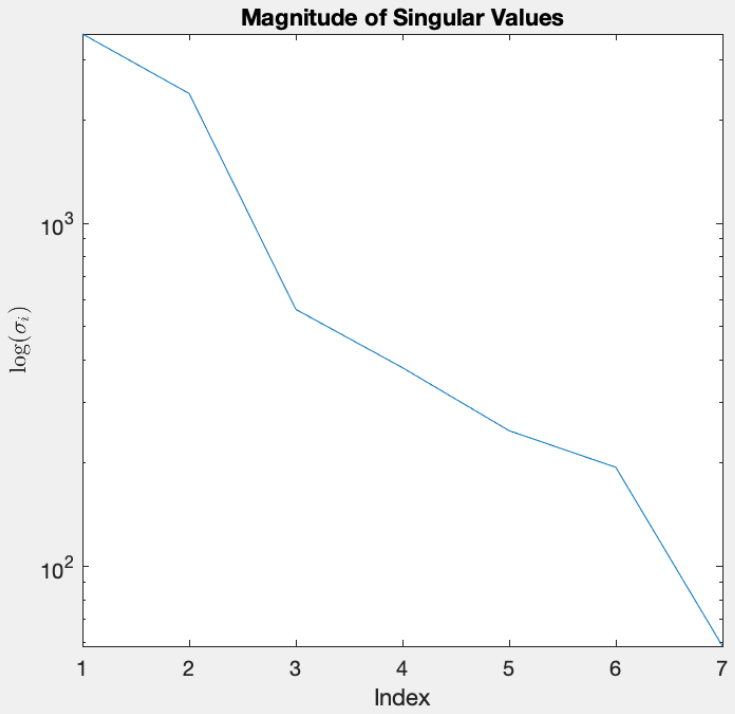
DiabetesPedigreeFunction: To express the Diabetes percentage

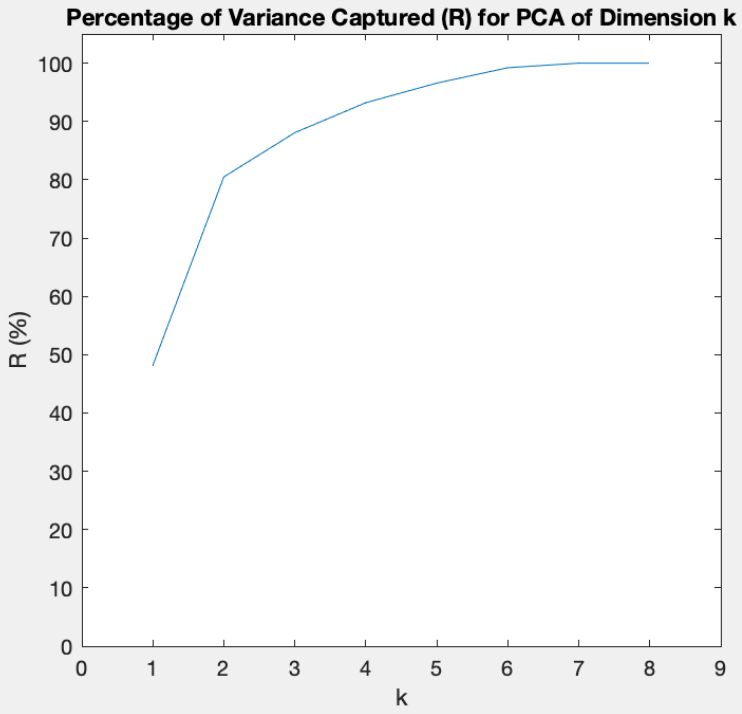
Age: To express the age

Outcome: To express the final result 1 is Yes and 0 is No

DIABETES DATA PCA, K = 3







x = Diabetes

o = No Diabetes

Online PCA

Part IV: An Iterative Principal Component Construction Method

With a touch of Gram-Schmidt

APPENDING THE SVD

Other approaches include a vector wise update to the covariance matrix. A similar approach can be made directly to the SVD:

Incremental SVD Update

When the $t+1^{th}$ row ($b_{t+1} \in \mathbb{R}^{1\times n}$) is added to a data matrix B, we must update the SVD.

$$B_{t+1} = \begin{bmatrix} B \\ b_{t+1} \end{bmatrix} = U\Sigma V^T + \begin{bmatrix} \mathbf{0} \\ b_{t+1} \end{bmatrix},$$

where $U\Sigma V^T$ is the current SVD of B, b_{t+1} is the newly added row, and $\mathbf{0}$ is the zero matrix of size $t \times n$.

$$p = b_{t+1}V,$$

where V is the $n \times t$ matrix of right singular vectors of B.

Calculate the residual r, which is the portion of b_{t+1} orthogonal to the current basis:

$$r = b_{t+1} - pV^T.$$

If $r = 0$, then b_{t+1} already lies in the span of V and can be discarded. Otherwise, append U:

$$u_{t+1} = \frac{r}{\|r\|}.$$

Adjust the SVD matrices accordingly:

$$\Sigma_{t+1} = \begin{bmatrix} \Sigma & 0 \\ 0 & \|r\| \end{bmatrix}, \quad V_{t+1} = \begin{bmatrix} V & 0 \\ 0 & 1 \end{bmatrix},$$

$$\text{and,} \quad B_{t+1} = U_{t+1}\Sigma_{t+1}V_{t+1}^T.$$