Dimensionality Reduction: Principal Component Analysis

Prepared by:
Rutvy Dhanesh Makwana
Tirth Manojkumar Patel
Pruthvik Dash

Instructor: Dr. Yasser Alginahi
Date of Submission: November 10, 2023

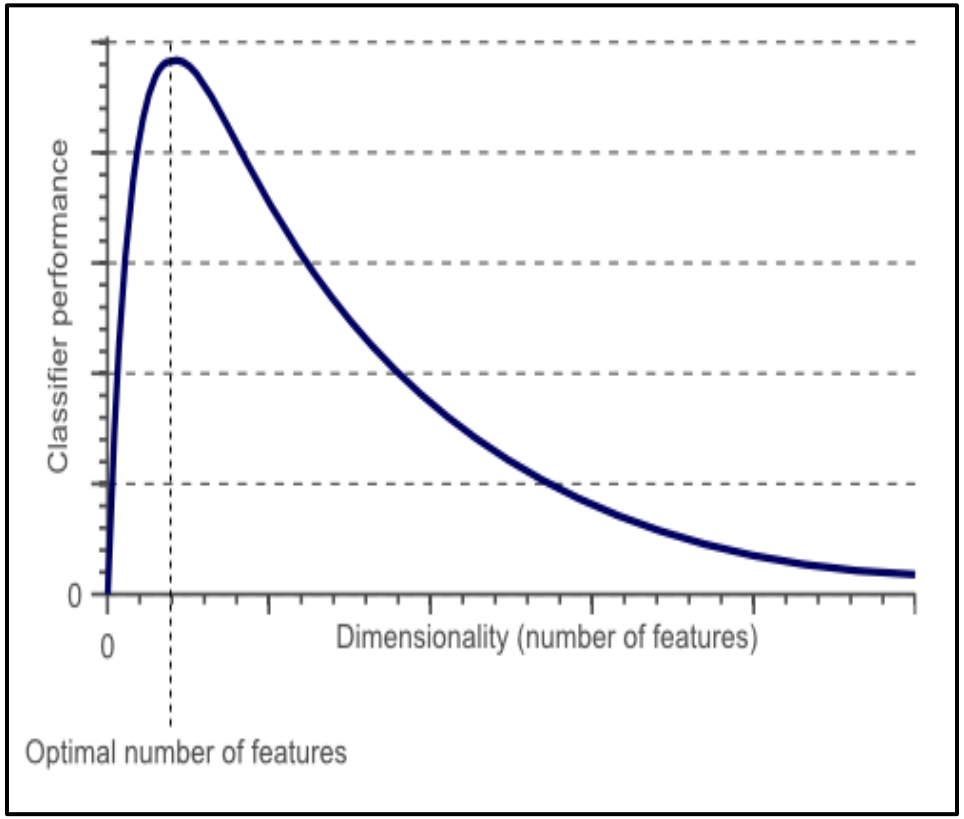
Table of Contents

Dimensionality Reduction

- Dimensionality Reduction refers to a technique for reducing the number of input variables in the training data.
- Useful when dealing with high dimensional data by projecting the data to a lower dimensional subspace.

Dimensionality Reduction

- Working with more dimensions of data initially boosts predictive ability but gets progressively challenging as the number of dimensions grows.



source: https://tinyurl.com/4nsdy98r

Why Dimensionality Reduction is needed?

1. Simplifies Models
2. Computational Complexity
3. Curse of Dimensionality
4. Enhances Visualization
5. Removes Redundant Information

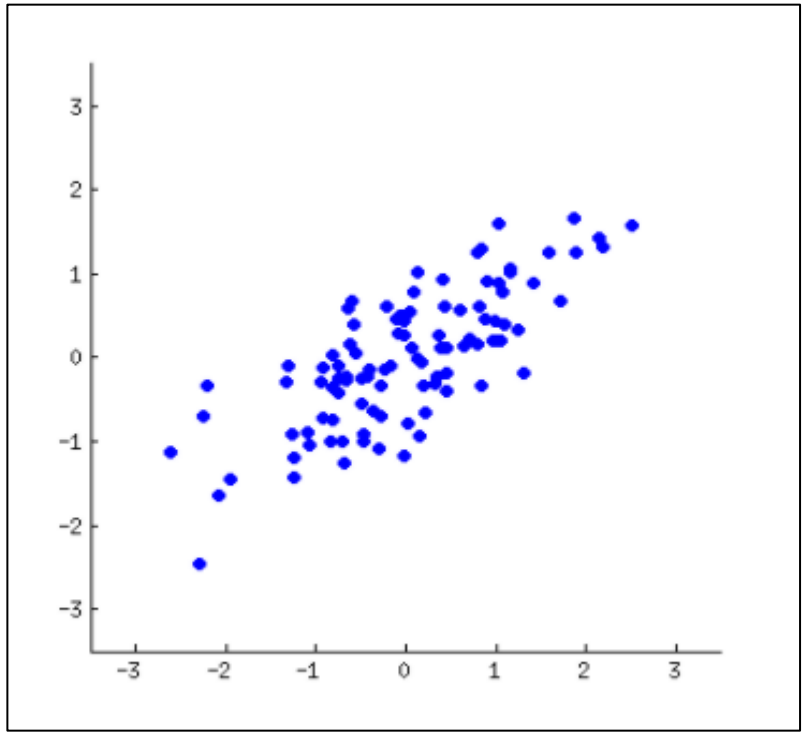
Principle Component Analysis

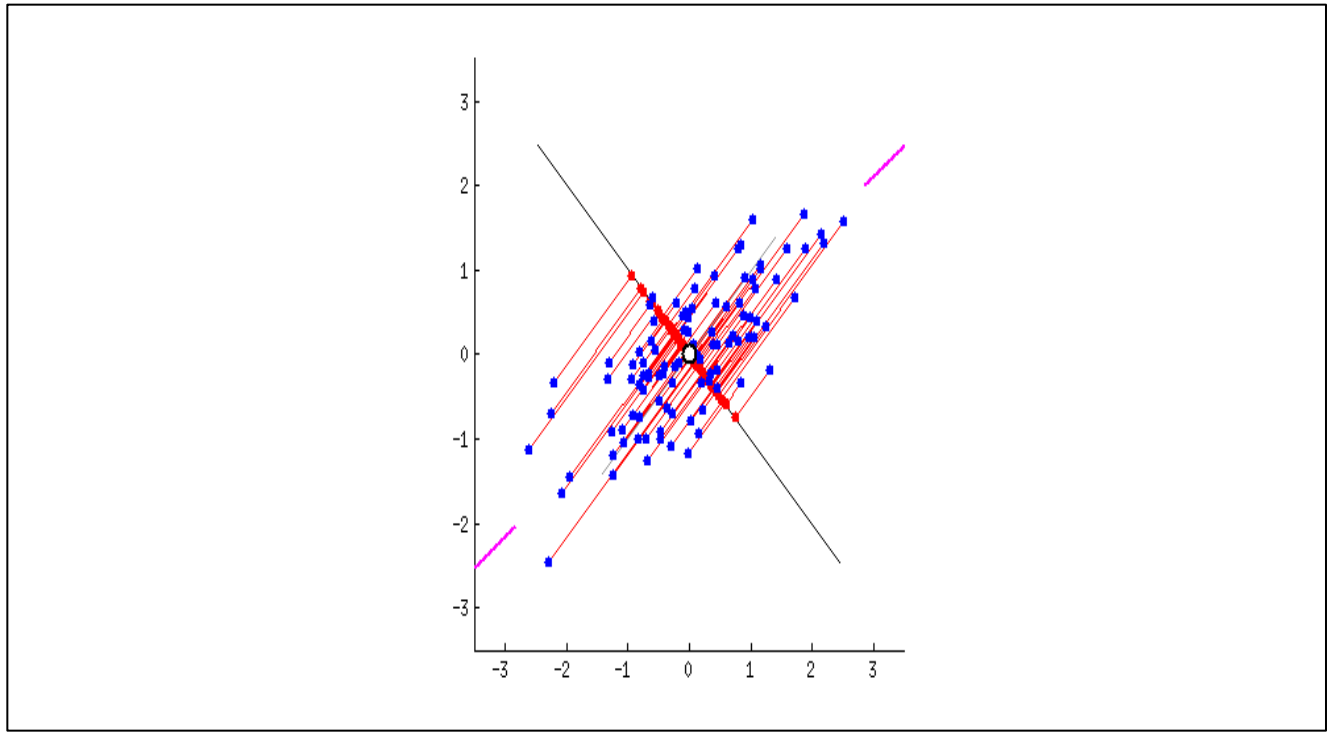
- PCA is a linear dimensionality reduction approach that turns a set of correlated high-dimensional features into a set of uncorrelated low-dimensional features.
- These uncorrelated features are called PRINCIPAL COMPONENTS.

Working of Principal Component Analysis

- Principal Component Analysis is a projection based method that transforms the data by projecting it onto a set of orthogonal(perpendicular) axes.
- It essentially finds the best linear combinations of the original variables such that the variance is maximum.

Example: Reducing 2 dimensional to 1 dimensional



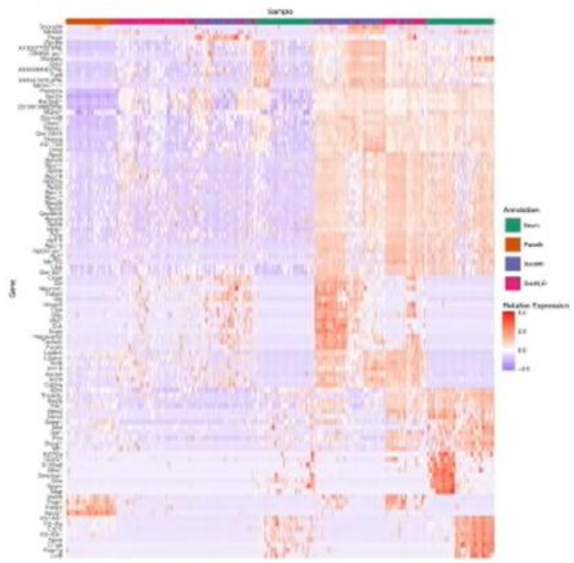


source: https://tinyurl.com/5eh46chz

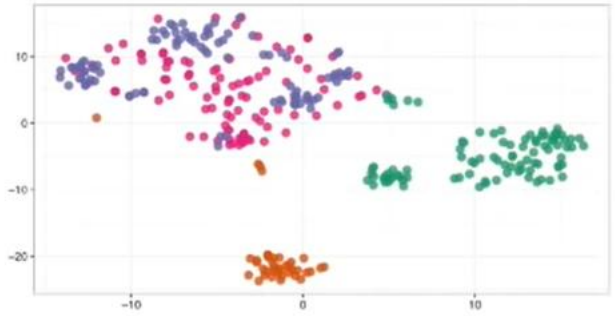
Dimensionality Reduction

- Different methods in Dimensionality Reduction:

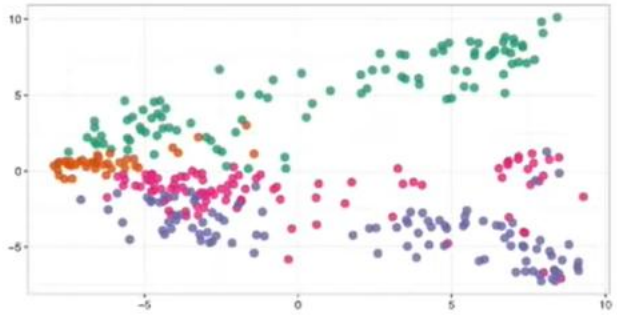
Heatmaps



t-SNE Plots

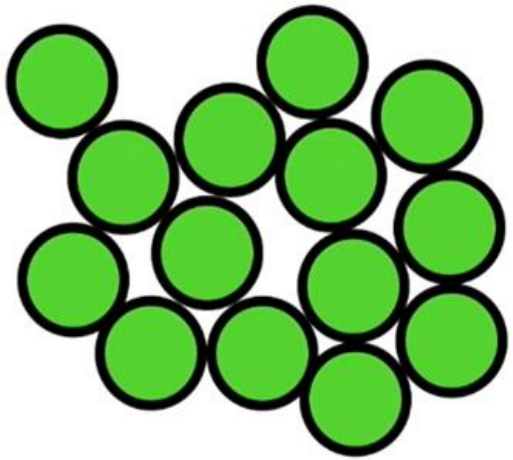


Multi-Dimensional Scaling (MDS)



source: https://tinyurl.com/24v5etfr

PCA with real life example



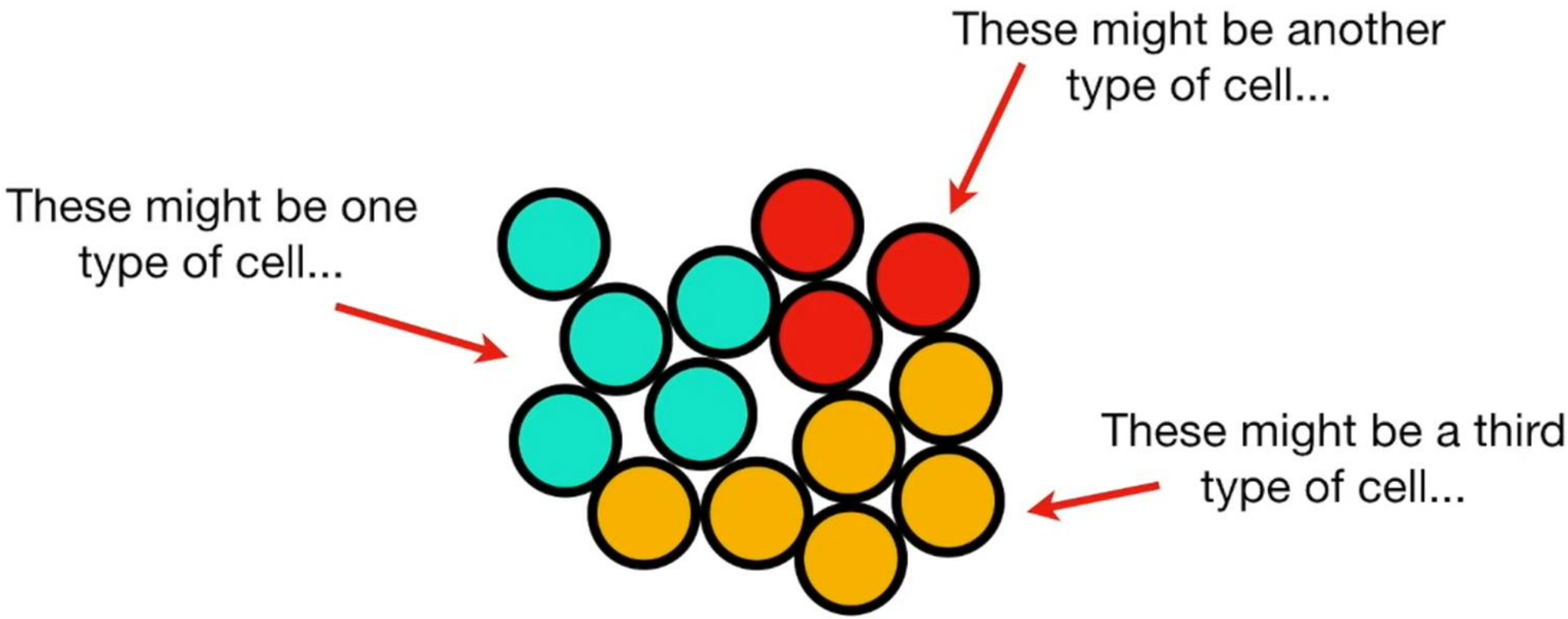
Lets say we had some normal cells…

Unfortunately, we can't observe the differences from the outside…

…so we sequence the mRNA in each cell to identify which genes are active. This tells us what the cell is doing.

source: https://tinyurl.com/24v5etfr

PCA with real life example



source: https://tinyurl.com/24v5etfr

PCA with real life example

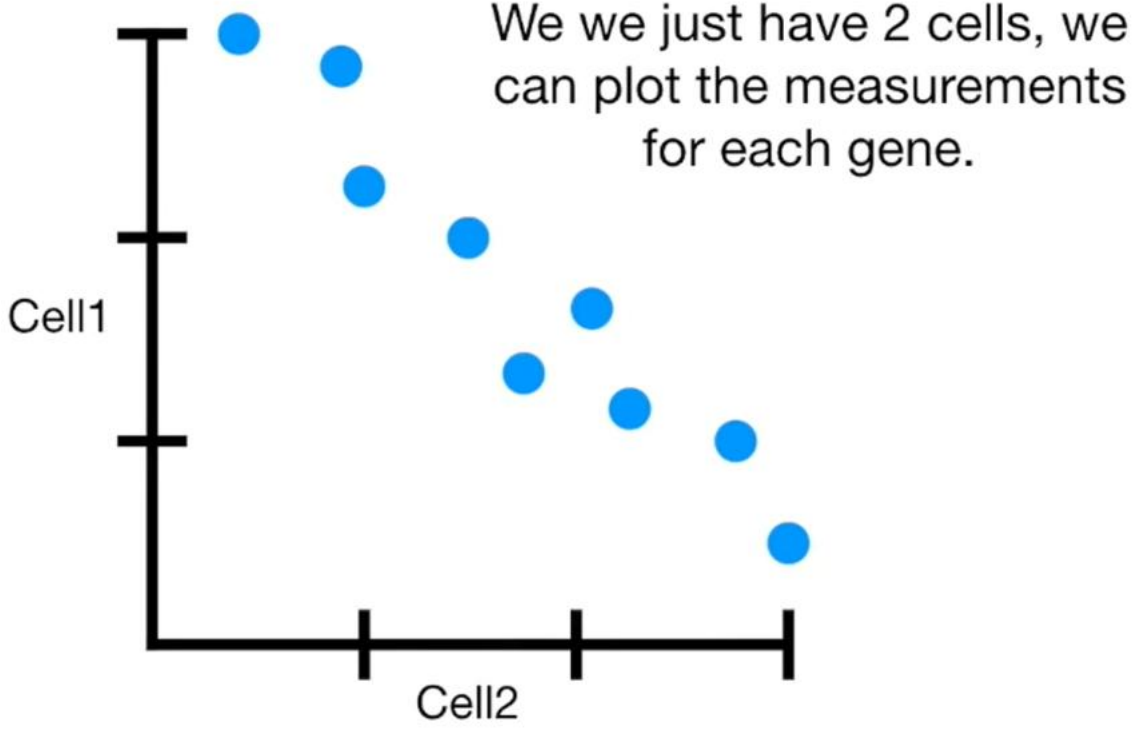
Here's the data…

- Each column shows how much each gene is transcribed in each cell.

	Cell1	Cell2	Cell3
Gene1	3	0.25	2.8
Gene2	2.9	0.8	2.2
Gene3	2.2	1	1.5
Gene4	2	1.4	2
Gene5	1.3	1.6	1.6
Gene6	1.5	2	2.1
Gene7	1.1	2.2	1.2
Gene8	1	2.7	0.9
Gene9	0.4	3	0.6

source: https://tinyurl.com/24v5etfr

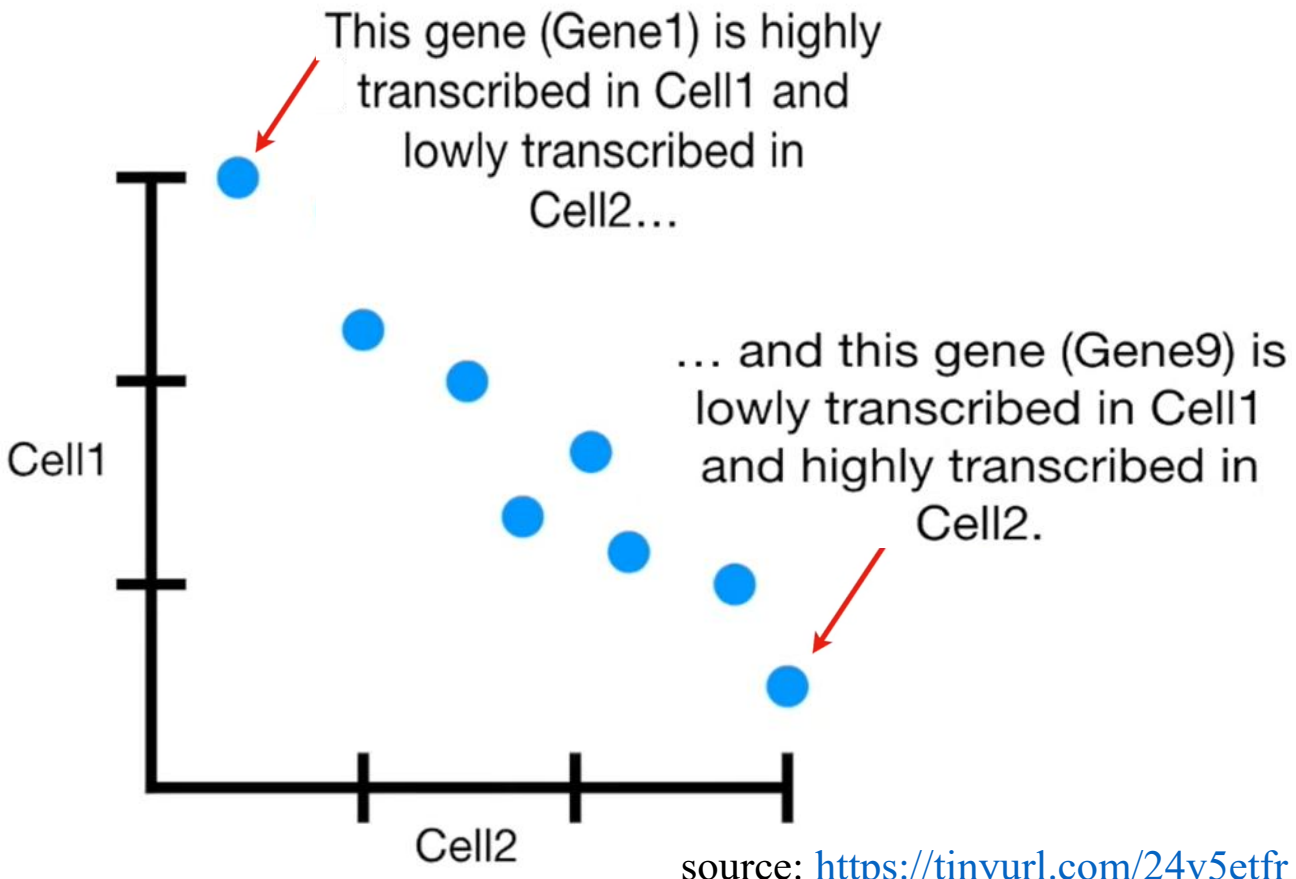
PCA with real life example



	Cell1	Cell2
Gene1	3	0.25
Gene2	2.9	0.8
Gene3	2.2	1
Gene4	2	1.4
Gene5	1.3	1.6
Gene6	1.5	2
Gene7	1.1	2.2
Gene8	1	2.7
Gene9	0.4	3

source: https://tinyurl.com/24v5etfr

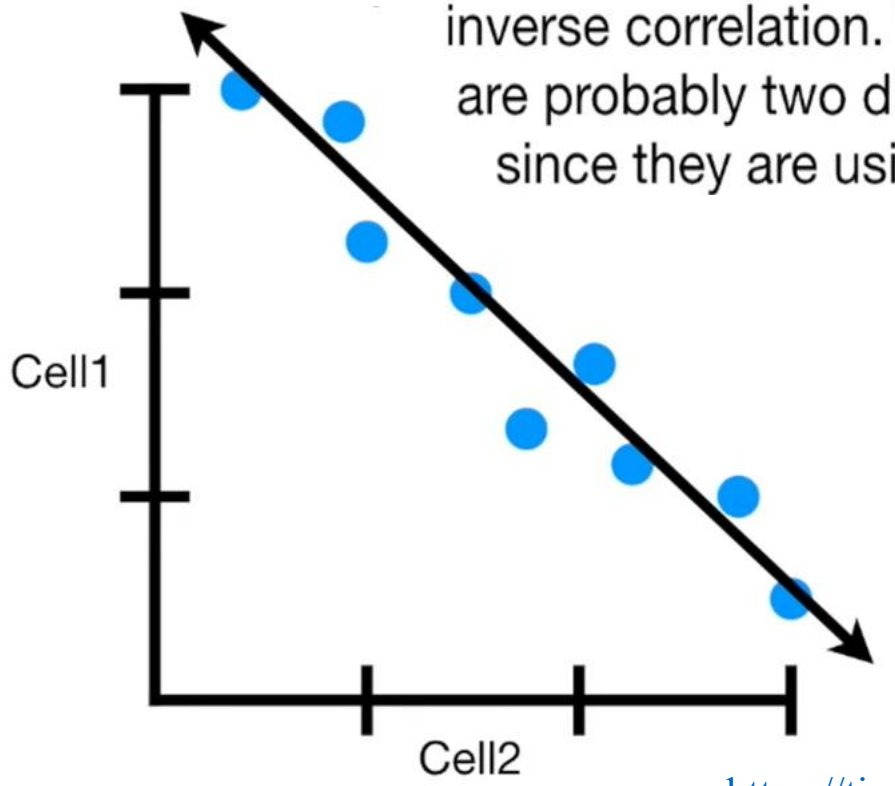
PCA with real life example



source: https://tinyurl.com/24v5etfr

	Cell1	Cell2
Gene1	3	0.25
Gene2	2.9	0.8
Gene3	2.2	1
Gene4	2	1.4
Gene5	1.3	1.6
Gene6	1.5	2
Gene7	1.1	2.2
Gene8	1	2.7
Gene9	0.4	3

PCA with real life example

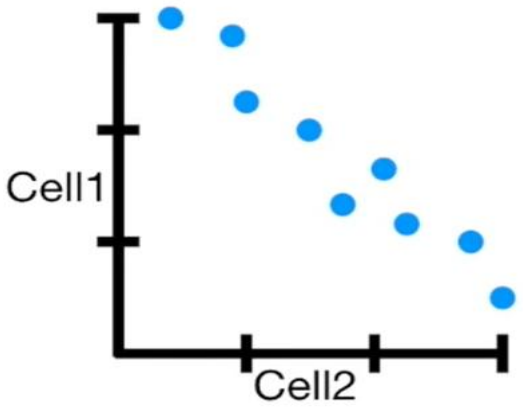


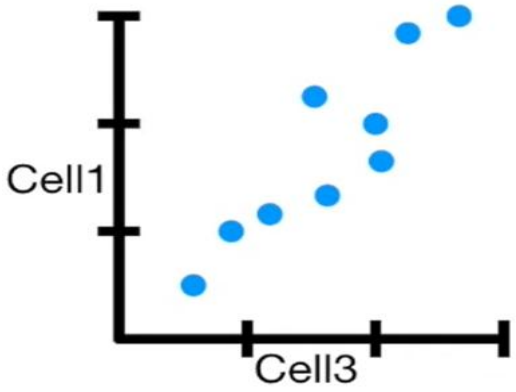
In general, Cell1 and Cell2 have an inverse correlation. This means that they are probably two different types of cells since they are using different genes.

	Cell1	Cell2
Gene1	3	0.25
Gene2	2.9	0.8
Gene3	2.2	1
Gene4	2	1.4
Gene5	1.3	1.6
Gene6	1.5	2
Gene7	1.1	2.2
Gene8	1	2.7
Gene9	0.4	3

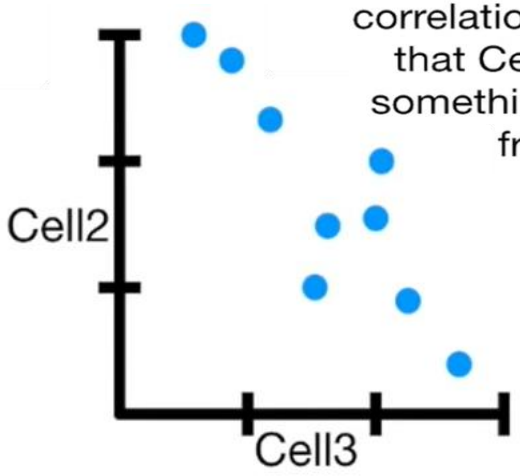
source: https://tinyurl.com/24v5etfr

PCA with real life example





Cell1 and Cell3 are positively correlated, suggesting they are doing similar things.

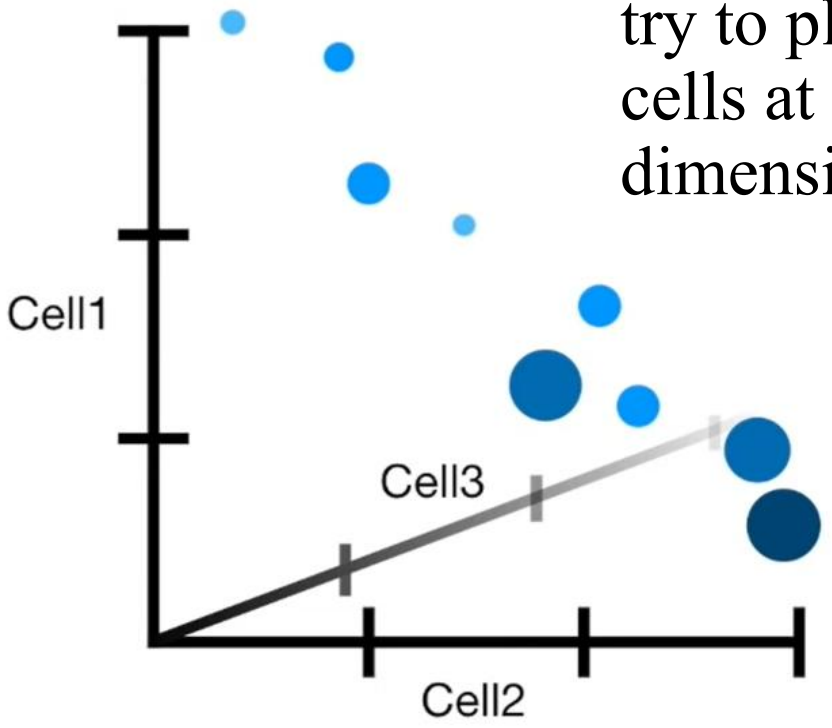


The negative correlation suggests that Cell2 is doing something different from Cell3…

	Cell1	Cell2	Cell3
Gene1	3	0.25	2.8
Gene2	2.9	0.8	2.2
Gene3	2.2	1	1.5
Gene4	2	1.4	2
Gene5	1.3	1.6	1.6
Gene6	1.5	2	2.1
Gene7	1.1	2.2	1.2
Gene8	1	2.7	0.9
Gene9	0.4	3	0.6

source: https://tinyurl.com/24v5etfr

PCA with real life example



- Alternatively, we could try to plot all three cells at once on a 3-dimensional graph.

	Cell1	Cell2	Cell3
Gene1	3	0.25	2.8
Gene2	2.9	0.8	2.2
Gene3	2.2	1	1.5
Gene4	2	1.4	2
Gene5	1.3	1.6	1.6
Gene6	1.5	2	2.1
Gene7	1.1	2.2	1.2
Gene8	1	2.7	0.9
Gene9	0.4	3	0.6

source: https://tinyurl.com/24v5etfr

PCA with real life example

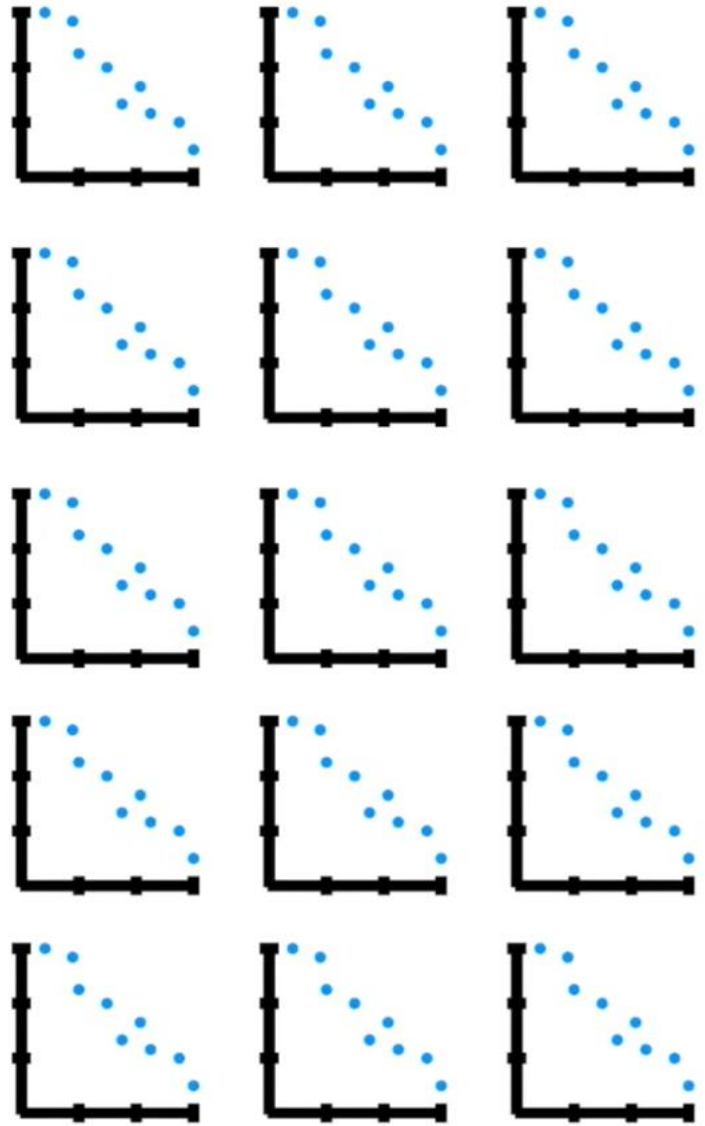
But what do we do when we have 4 or more cells?

	Cell1	Cell2	Cell3	Cell4	...
Gene1	3	0.25	2.8	0.1	...
Gene2	2.9	0.8	2.2	1.8	...
Gene3	2.2	1	1.5	3.2	...
Gene4	2	1.4	2	0.3	...
Gene5	1.3	1.6	1.6	0	...
Gene6	1.5	2	2.1	3	...
Gene7	1.1	2.2	1.2	2.8	...
Gene8	1	2.7	0.9	0.3	...
Gene9	0.4	3	0.6	0.1	...

source: https://tinyurl.com/24v5etfr

PCA with real life example

Draw tons and tons of 2 cell plots and try to make sense of them all?

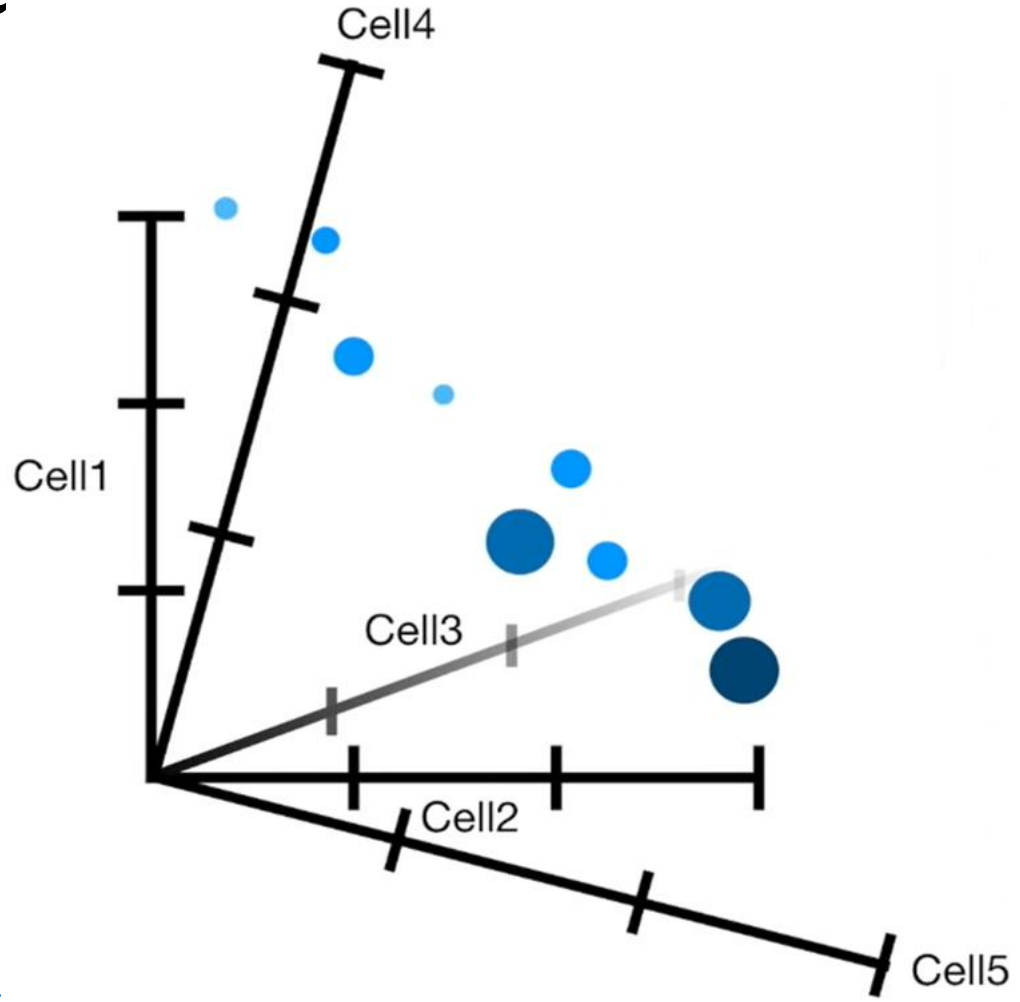


source: https://tinyurl.com/24v5etfr

PCA with real life example

Or

draw some crazy graph that has an axis for each cell and makes our brain explode?



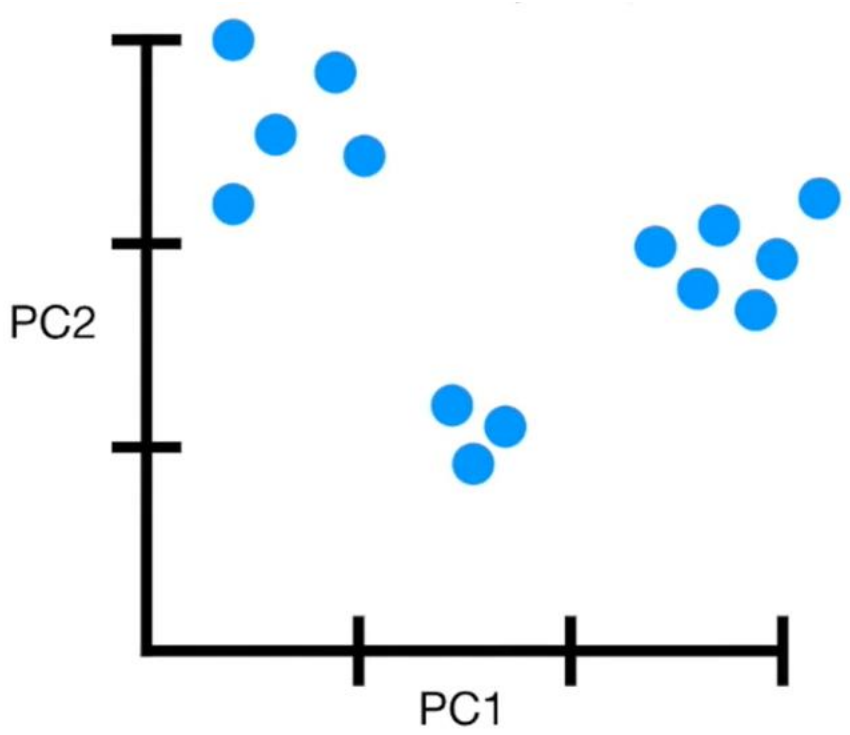
source: https://tinyurl.com/24v5etfr

PCA with real life example

No, both of those options will not work.

Instead, we draw a Principal Component Analysis(PCA) plot…

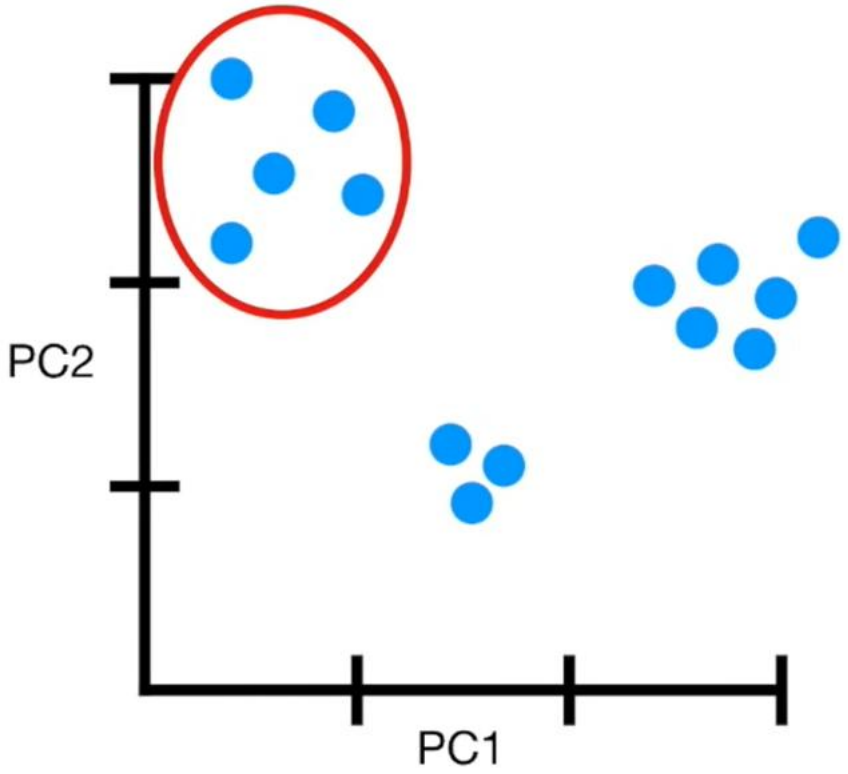
A PCA plot converts the correlations (or lack thereof) among all the cells into a 2-D graph.



source: https://tinyurl.com/24v5etfr

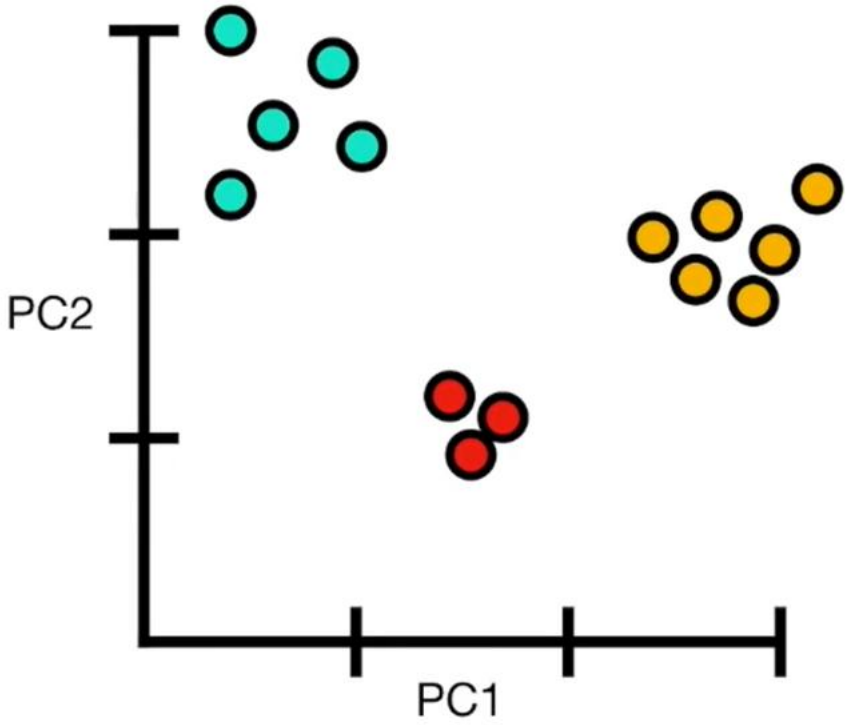
PCA with real life example

- This cluster of cells are highly correlated with each other…
- Cells that are highly correlated cluster together…



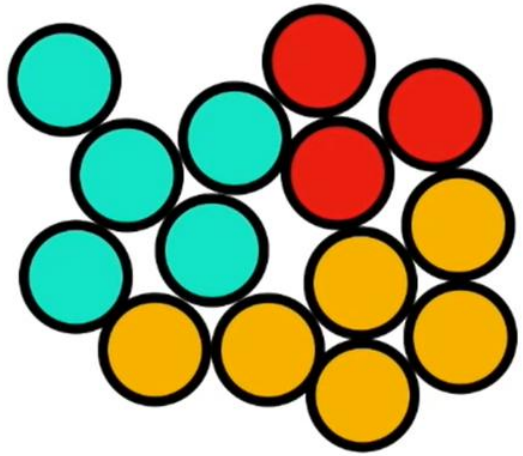
source: https://tinyurl.com/24v5etfr

PCA with real life example



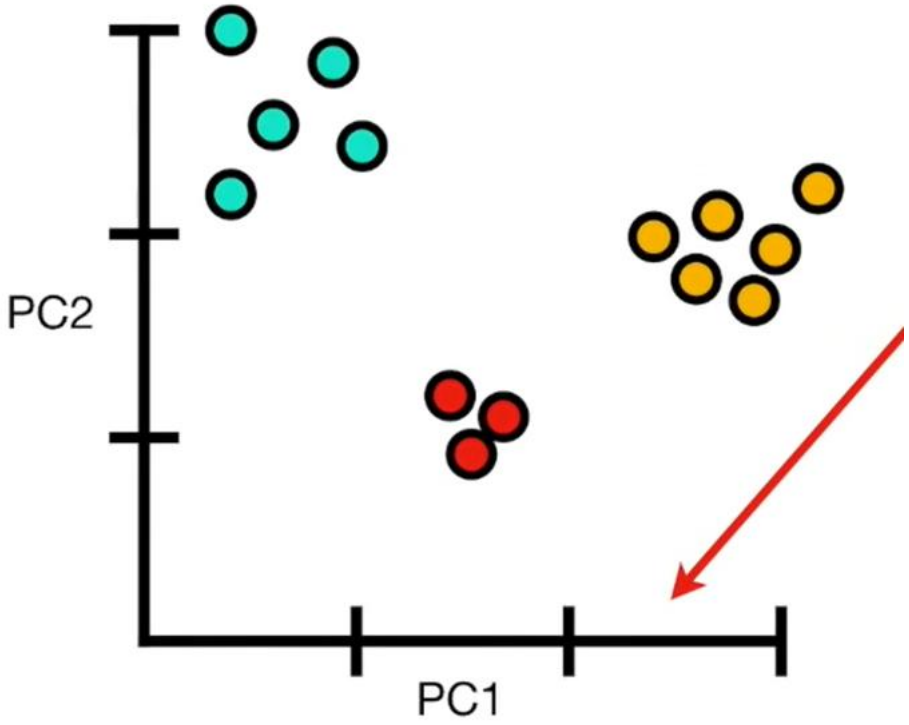
source: https://tinyurl.com/24v5etfr

- Once we've identified the clusters in the PCA plot,we can go back to the original cells….



- …and see that they represent 3 different types of cells doing 3 different things with their genes!!!!

PCA with real life example

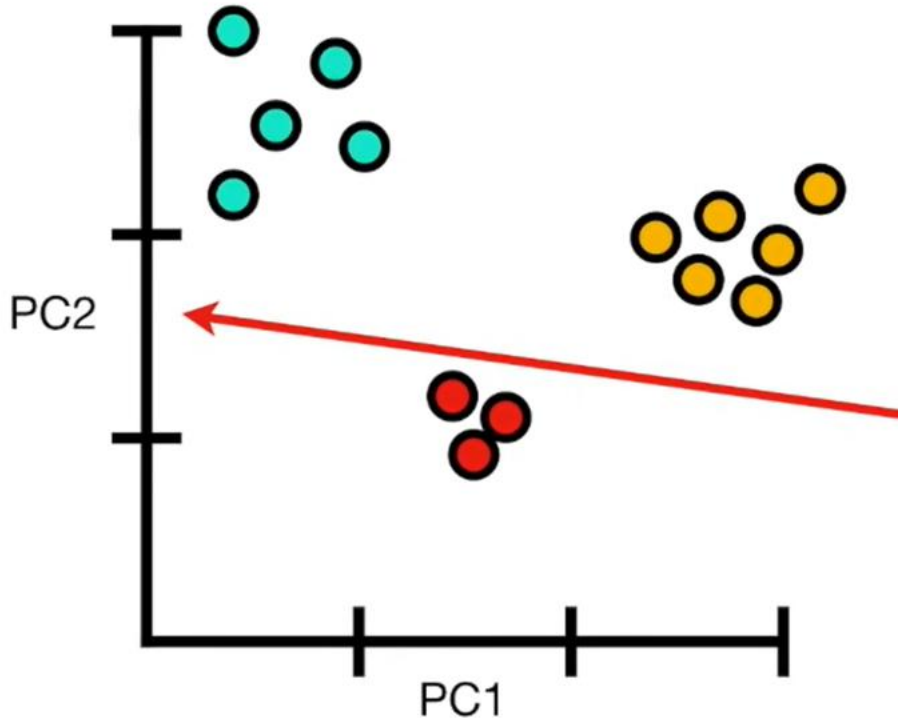


- Here's one last main idea about how to interpret PCA plots:
- The axes are ranked in order of importance.

Differences along the first principal component axis(PC1)…

source: https://tinyurl.com/24v5etfr

PCA with real life example

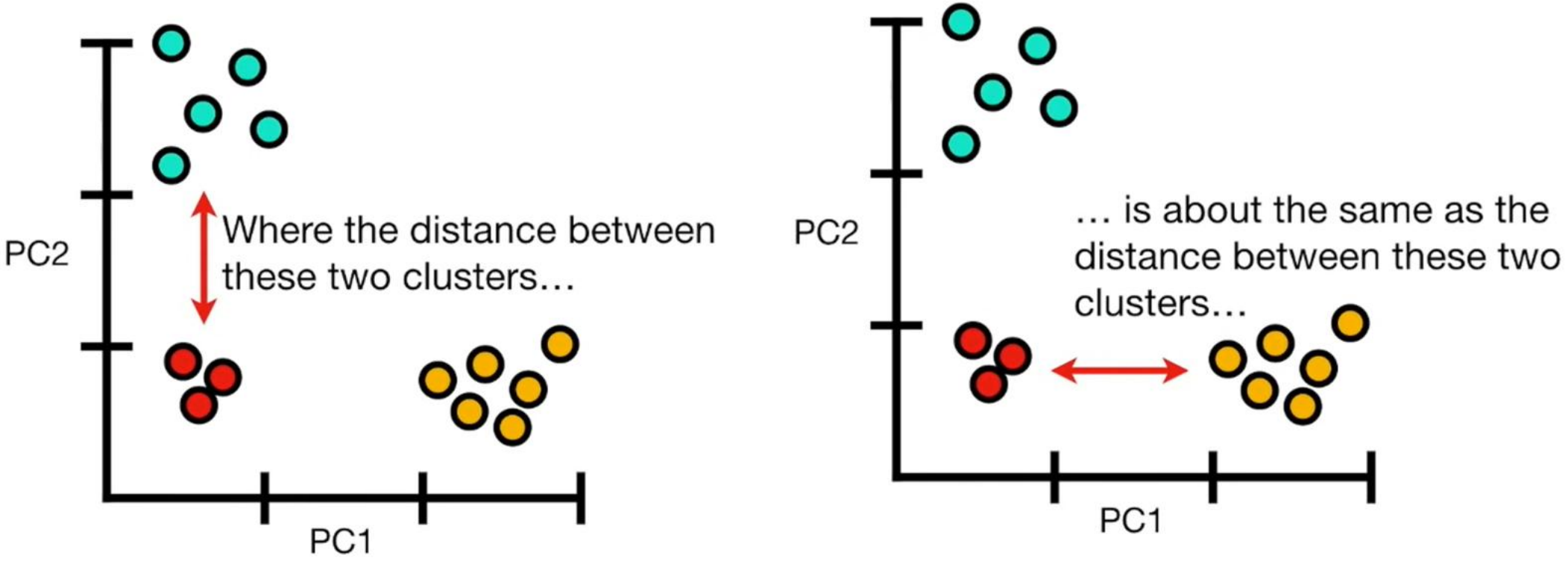


Differences along the first principal component axis(PC1)…

…are more important than differences along the second principal component axis(PC2).

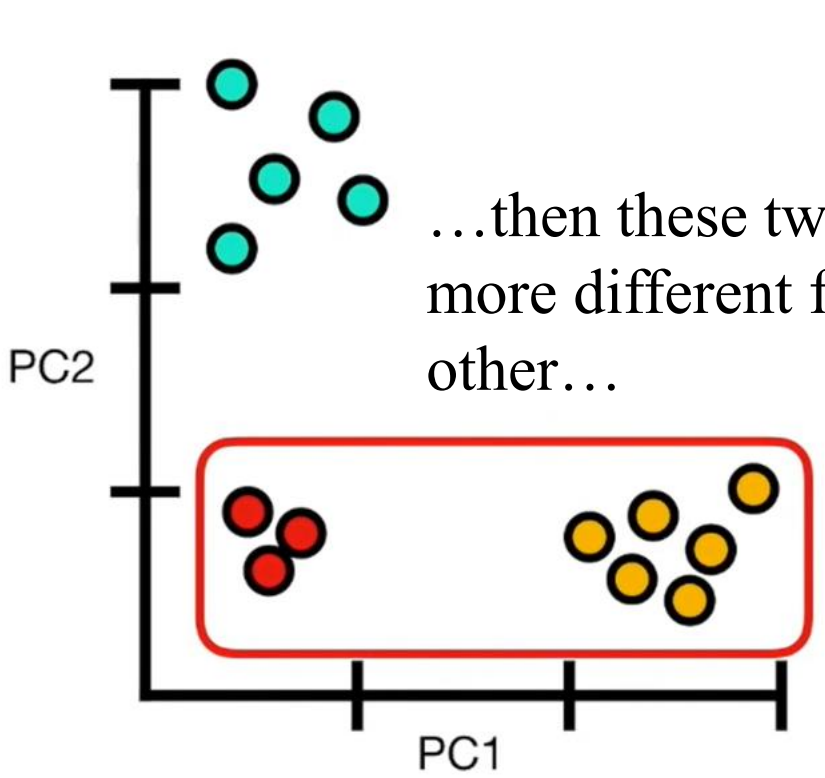
source: https://tinyurl.com/24v5etfr

PCA with real life example

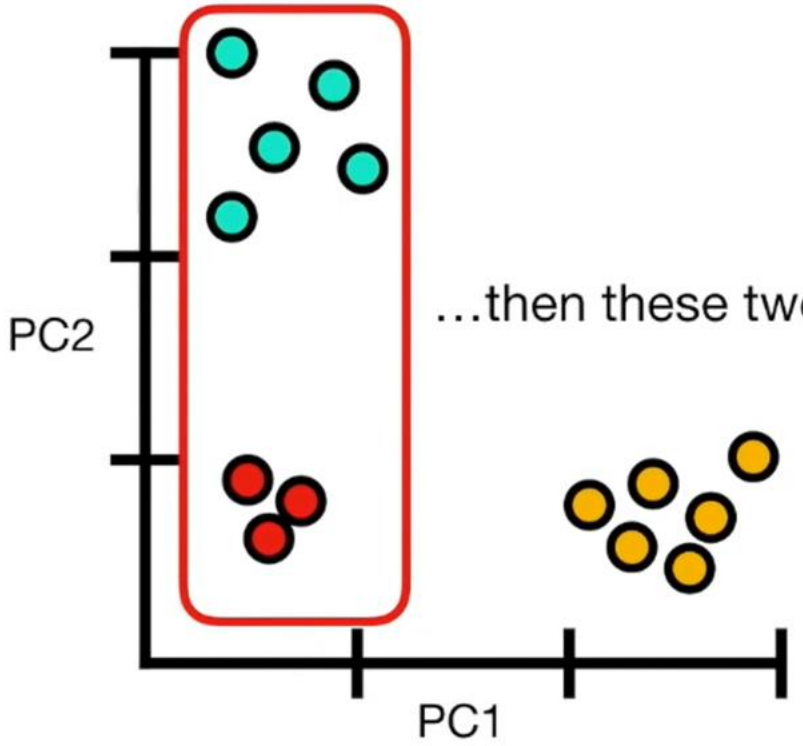


source: https://tinyurl.com/24v5etfr

PCA with real life example



…then these two clusters are more different from each other…



…then these two clusters.

source: https://tinyurl.com/24v5etfr

Steps in PCA

1.Standardize the Data:

Ensure all features have the same scale.

2.Calculate Covariance Matrix:

Describes how different features vary together.

3.Compute Eigenvectors and Eigenvalues:

Eigenvectors: Directions of maximum variance.

Eigenvalues: Magnitudes of the variance in each direction.

4.Choose Principal Components:

Select the top eigenvectors based on eigenvalues.

5.Project Data Onto Principal Components:

Transform the data into the new reduced-dimensional space.

Example For PCA

- **Suppose we have 2D dataset.**

$$X = \begin{bmatrix} 1 & 2 \\ 2 & 4 \\ 3 & 6 \\ 4 & 8 \end{bmatrix}$$

1. **Calculate the Mean:**

$$\text{Mean}(X_1) = \frac{1+2+3+4}{4} = 2$$
$$\text{Mean}(X_2) = \frac{2+4+6+8}{4} = 5$$

2. **Center the data:**

$$X' = \begin{bmatrix} -1 & 0 \\ 0 & 2 \\ 1 & 4 \\ 2 & 6 \end{bmatrix}$$

Example For PCA

3. **Calculate Covariance Matrix:**

$$\text{Cov}(X_1', X_2') = \frac{\sum (X_1' \cdot X_2')}{n-1}$$
$$\text{Cov}(X_1', X_2') = \frac{(-1 \cdot 0)+(0 \cdot 2)+(1 \cdot 4)+(2 \cdot 6)}{3} = \frac{20}{3}$$

4. **Calculate Eigenvectors and Eigenvalues:**

Solving for eigenvectors and eigenvalues, an eigenvector [1,1] and eigenvalue 10.

$$\det(A - \lambda I) = 0$$

$$(A - \lambda I)\mathbf{v} = \mathbf{0}$$

$$\det\left(\begin{bmatrix} \frac{20}{3} - \lambda & 0 \\ 0 & \frac{20}{3} - \lambda \end{bmatrix}\right) = 0$$

$$\begin{bmatrix} -\frac{14}{3} & 0 \\ 0 & -\frac{14}{3} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\lambda = \frac{20}{3}$$

$$\mathbf{v} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

Example For PCA

5. Project Data Onto Principal Component:

$$\text{Projection} = X' \times \text{Top Eigenvector}$$

$$\text{Projection} = \begin{bmatrix} -1 & 0 \\ 0 & 2 \\ 1 & 4 \\ 2 & 6 \end{bmatrix} \times \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$\text{Projection} = \begin{bmatrix} -1+0 \\ 0+2 \\ 1+4 \\ 2+6 \end{bmatrix} = \begin{bmatrix} -1 \\ 2 \\ 5 \\ 8 \end{bmatrix}$$

So, in this simplified example, we retained only one dimension, and the projected data is [−1,2,5,8].

Coded Example of PCA

Libraries

```
[1]  # importing Library
     import pandas as pd
     import matplotlib.pyplot as plt
     import numpy as np
     import seaborn as sns
     %matplotlib inline
```

The Data Inbuilt Breast Cancer data set

```
[2]  # importing data set
     from sklearn.datasets import load_breast_cancer
```

```
[3]  # uploading data set in variable
     cancer = load_breast_cancer()
```

Code Link Access:
https://colab.research.google.com/drive/18O8QXvwjDIgePfGSfLN2MKoP5lrIphhu#scrollTo=pADSbZ77u34J

```
[4]  cancer.keys()
```

```
dict_keys(['data', 'target', 'frame',
'target_names', 'DESCR', 'feature_names',
'filename', 'data_module'])
```

```
[5]  # Information of Data set
     print(cancer['DESCR'])
```

source: https://tinyurl.com/4phpne3x

Coded Example of PCA

```
[6]  df = pd.DataFrame(cancer['data'],columns=cancer['feature_names'])
     #(['DESCR', 'data', 'feature_names', 'target_names', 'target'])
```

```
[7]  df.head()
```

```
[8]  from sklearn.preprocessing import StandardScaler
```

```
[9]  scaler = StandardScaler()
     scaler.fit(df)
```

```
▾ StandardScaler
StandardScaler()
```

source: https://tinyurl.com/4phpne3x

Coded Example of PCA

```
[10] scaled_data = scaler.transform(df)
```

```
[12] from sklearn.decomposition import PCA
```

```
[13] pca = PCA(n_components=2)
```

```
pca.fit(scaled_data)
```

```
PCA(n_components=2)
```

```
[15] x_pca = pca.transform(scaled_data)
```

```
[16] scaled_data.shape
```

```
(569, 30)
```

```
[17] x_pca.shape
```

```
(569, 2)
```

```
[18] plt.figure(figsize=(8,6))
     plt.scatter(x_pca[:,0],x_pca[:,1],c=cancer['target'],cmap='plasma')
     plt.xlabel('First principal component')
     plt.ylabel('Second principal component')
```

source: https://tinyurl.com/4phpne3x

Applications

1. Image Compression
2. Face Recognition
3. Speech Recognition

Advantages of Principal Component Analysis

1. Noise Reduction
2. Visualization
3. Collinearity Handling

Disadvantages of Principal Component Analysis

- **Information Loss**

Dimensionality reduction involves some loss of information.

The challenge is to strike a balance between reducing dimensionality and retaining enough information for the specific task at hand.

- **Non-linearity**

PCA assumes linear relationships between features. Non-linear dimensionality reduction techniques (e.g., t-Distributed Stochastic Neighbor Embedding - t-SNE) exist for capturing more complex relationships.

Conclusion

- **Key Takeaways**

1.PCA is Powerful:

1. PCA is a powerful technique for dimensionality reduction.

2.Simplification and Retention:

1. It simplifies data while retaining essential information.

3.Careful Component Selection:

1. Carefully choose the number of components based on explained variance.

Refrences

- [1] J. Brownlee, “Introduction to dimensionality reduction for machine learning,” MachineLearningMastery.com, https://machinelearningmastery.com/dimensionality-reduction-for-machine-learning/#:~:text=Dimensionality%20reduction%20refers%20to%20techniques%20for%20reducing%20the%20number%20of,%E2%80%9Cessence%E2%80%9D%20of%20the%20data. (accessed Nov. 10, 2023).
- [2] K. Parte, “Dimensionality reduction: Principal Component Analysis,” Medium, https://medium.com/analytics-vidhya/dimensionality-reduction-principal-component-analysis-d1402b58feb1 (accessed Nov. 10, 2023).
- [3] N. S. Chauhan, “Dimensionality reduction with principal component analysis (PCA),” KDnuggets, https://www.kdnuggets.com/2020/05/dimensionality-reduction-principal-component-analysis.html (accessed Nov. 10, 2023).
- [4] A. K. Pal, “Tutorial: Understanding dimension reduction with principal component analysis,” Paperspace Blog, https://blog.paperspace.com/dimension-reduction-with-principal-component-analysis/ (accessed Nov. 10, 2023).
- [5] “Dimensional Reduction| Principal Component Analysis,” *www.youtube.com*. https://www.youtube.com/watch?v=OFyyWcw2cyM&ab_channel=KrishNaik (accessed Nov. 10, 2023).