Machine Learning and Computational Intelligence
Lecture 2

Sanjeeb Prasad Panday, PhD
Associate Professor
Dept. of Electronics and Computer Engineering
Director (ICTC)
IOE, TU

Overfitting and Underfitting

- We kind of saw that the relationship in a dataset can be represented using a polynomial of this form:

$$y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \ldots + w_M x^M$$

- Just assume that this is any polynomial of order, M. We want to understand want happens when the order gradually increases from 0 to say 13.
- Also assume that the w terms are constant.
- Remember our goal is to ensure that this polynomial fits the dataset. That is, after plotting the dataset as a scatter plot, then we fit this polynomial through it.

Overfitting and Underfitting

- **How M relates to E**
- Also recall that we would like to minimize the error term E(x, w*). This is so that the difference between predicted values and the actual values is very small.

$$y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \ldots + w_M x^M$$

- Now as M (the order of the polynomial) increases, the error, E decreases. This also means that the model complexity increases since we have higher order polynomial.

- Similarly, when M is low, the complexity of the model reduces, meaning that the model becomes kind of simple.

Overfitting and Underfitting

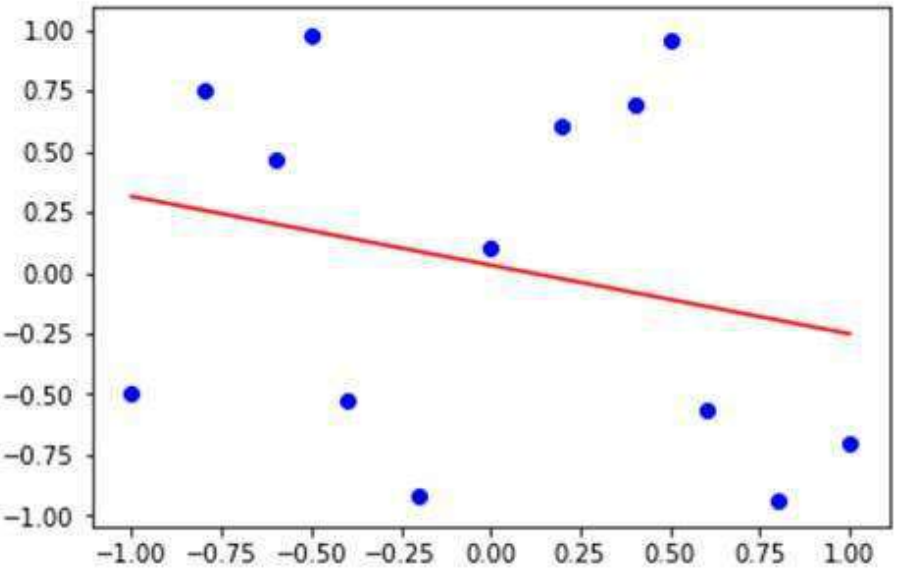
- So, it then appear that to reduce the error E to the barest minimum, or even zero, we simply keep increasing M. But there is a problem.

- Let's now see what happens when M is too low and when M is too high.

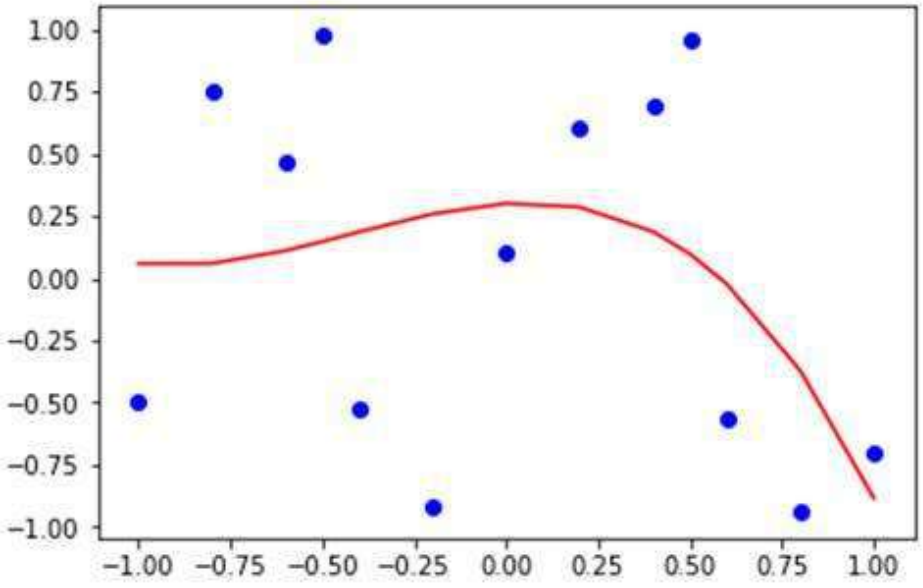
Overfitting and Underfitting

- **If M is very low (Underfitting)**
- As mentioned, if M is very low, then polynomial will not be able to properly model the relationship between the variable. So, the following will be true:
  - the model (polynomial) will not give a good fit to the dataset
  - the model is simple and easy to manipulate
  - the error is high
  - You can see how the polynomial looks for values of M =1 and M = 3 in the next slide.

Overfitting and Underfitting



M = 1



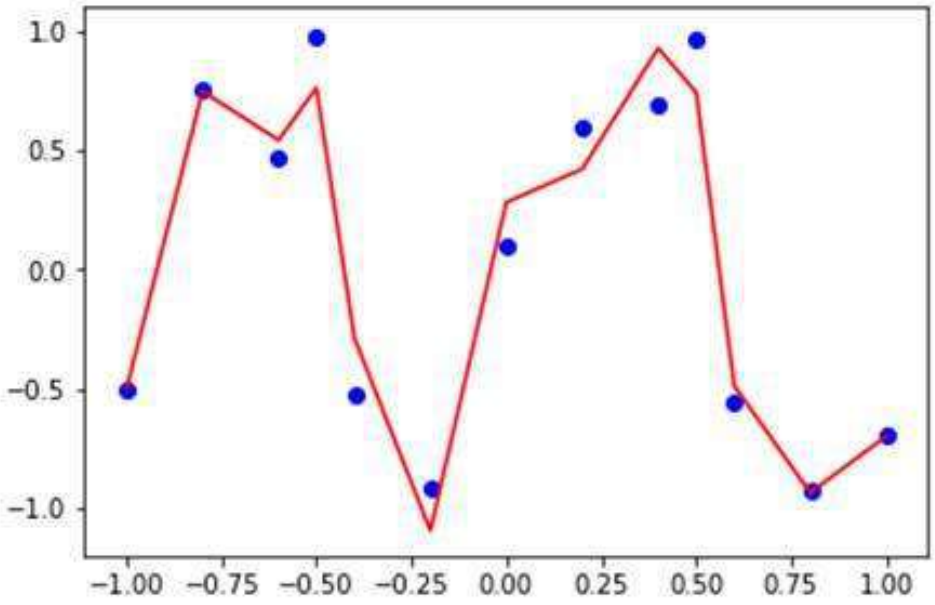
M = 3

- You can see that the model does not fit the given dataset properly. This problem is called underfitting.

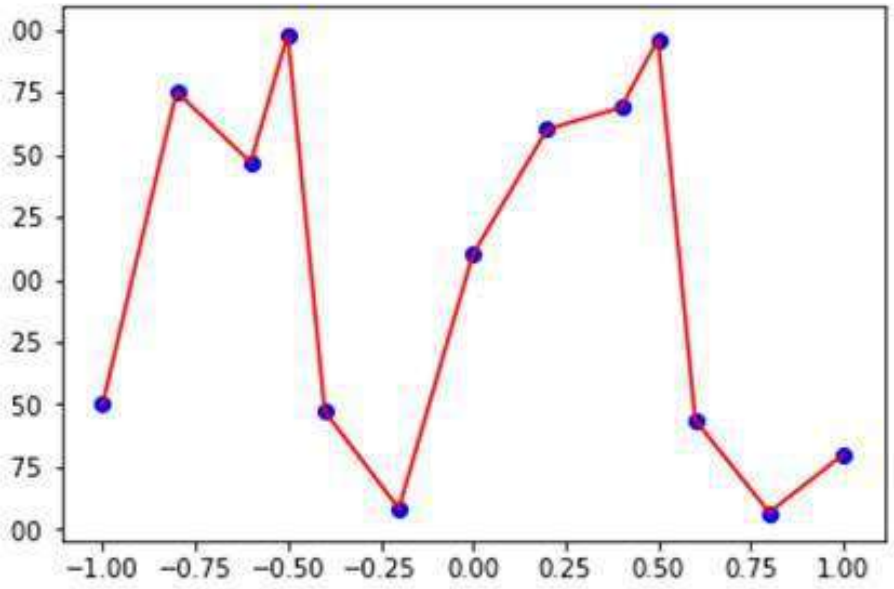
Overfitting and Underfitting

- **If M is very high (Overfitting)**
- On the other hand, if M is very high, then the following will be true:
  - the model will fit the dataset very closely or even match every point in the dataset
  - the model's complexity will increase
  - the error also becomes high (it decreases, then increases again)
  - You can see how the polynomial looks for values of M = 11 and M = 12 as shown in the next slide.

Overfitting and Underfitting



M = 11



M = 12

- In this case of M = 11 and M =12, you can see that the model fits the data very closely. There are two problems with this:
  - first, the model becomes too complex
  - second, the model is not able to generalize (predict new values) properly
  - So, this is the problem of overfitting.

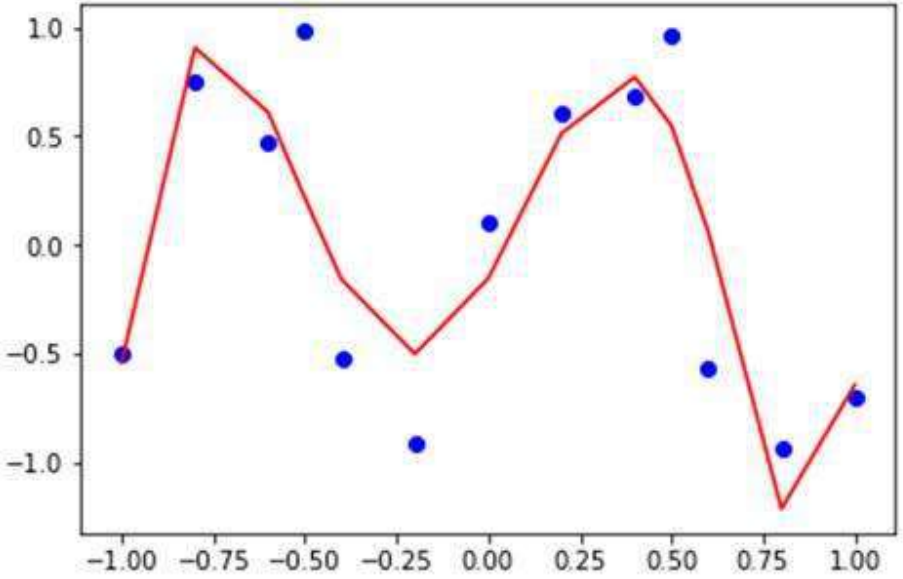
Overfitting and Underfitting

- **The Trade-off**
- Therefore, we need to find a trade-off between the too extremes we just discussed. This trade-off has a special name in Machine Learning. It is called **Bias-Variance Trade-off.**

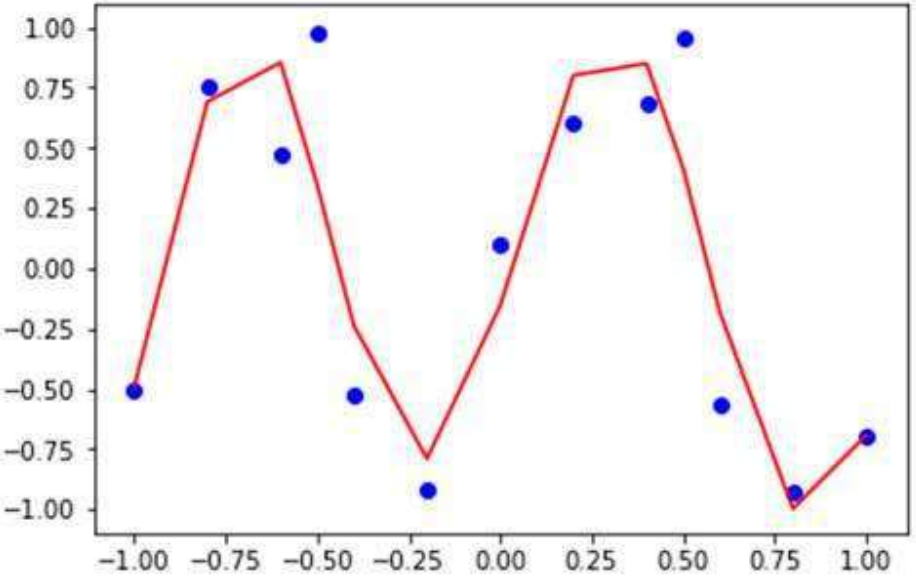
- At the trade-off point, the error is minimum.

- The plot for values of M = 6 and M = 7 is shown in the next slide:

Overfitting and Underfitting



M = 6



M = 7

Bias-Variance Trade-off

- **Definition of Bias-Variance Trade-off**

- First, let's take a simple definition. Bias-Variance Trade-off refers to the property of a machine learning model such that as the bias of the model increased, the variance reduces and as the bias reduces, the variance increases.

- Therefore the problem is to determine the amount of bias and variance to make the model optimal.

Bias-Variance Trade-off

- **Sources of Error**
- We recall the problem of underfitting and overfitting when trying to fit a regression line through a set of data points.
- In case of underfitting, the bias is an error from a faulty assumption in the learning algorithm. This is such that when the bias is too large, the algorithm would be able to correctly model the relationship between the features and the target outputs.
- In case of overfitting, variance is an error resulting from fluctuations in the training dataset. A high value for variance would cause the algorithm may capture the most data points put would not be generalized enough to capture new data points. This is overfitting.

Bias-Variance Trade-off

- **Sources of Error**
- The trade-off, means that a model would be chosen carefully to both correctly capture that regularities in the training data and at the same time be generalized enough to correctly categorize new observation

Bias-Variance Trade-off

- **Bias-Variance Decomposition of Squared Error**
- Considering the squared loss function and the conditional distribution of the training data set, we could summarize the formula for the expected loss to be:
- ***Expected Loss = (bias)^2 + variance + noise***
- Now assuming $y = f(x)$ representing the true relationship between the variables in the training data set
- Also let function $f'(x)$ which is an approximation of $f(x)$ through the learning process

Bias-Variance Trade-off

- **Bias-Variance Decomposition of Squared Error**
- Then we measure the mean squared error between y and $f'(x)$ which is given as: $(y - f'(x))^2$.
  This error is expected to be minimal.
- We than then write the original expected loss equation as:
- $E[(y - f'(x))^2] = Bias[f'(x)]^2 + Var[f'(x)] + \sigma^2$
- where:
  $Bias[f'(x)] = E[(f'(x) - f(x)]$
- and
  $Var[f'(x)] = E[f'(x)^2] - E[f'(x)]^2$
- and
  σ^2 represents the noise term in the equation

Bias-Variance Trade-off

- **Derivation of the Equation**
- We have independent variables x that affect the value of a dependent variable y. Function f denotes the true relationship between x and y.
- In real life problems it is very hard to know this relationship. y is given by this formula along with some noise which is represented by the random variable ε with zero mean and variance σ_ε^2:

$$y = f(x) + \epsilon$$

$$\mathbb{E}[\epsilon] = 0, \mathrm{var}(\epsilon) = \mathbb{E}[\epsilon^2] = \sigma_\epsilon^2$$

Bias-Variance Trade-off

- Now, when we try to model the underlying real-life problem, we try to find a function $\hat{f}$ that can accurately predict the true relationship f.
- The goal is to bring the prediction as close as possible to the actual value ($y \approx \hat{f}(x)$) to minimise the error.

$$\mathbb{E}[(y - \hat{f}(x))^2] = \text{bias}[\hat{f}(x)]^2 + \text{var}(\hat{f}(x)) + \sigma_\epsilon^2$$

$$\text{bias}[\hat{f}(x)] = \mathbb{E}[\hat{f}(x)] - f(x)$$

$$\text{var}(\hat{f}(x)) = \mathbb{E}[(\hat{f}(x) - \mathbb{E}[\hat{f}(x)])^2]$$

$$
\begin{aligned}
\mathbb{E}[(y - \hat{f}(x))^2] &= \mathbb{E}[(f(x) + \epsilon - \hat{f}(x))^2] \\
&= \mathbb{E}[(f(x) - \hat{f}(x))^2] + \mathbb{E}[\epsilon^2] + 2\mathbb{E}[(f(x) - \hat{f}(x))\epsilon] \\
&= \mathbb{E}[(f(x) - \hat{f}(x))^2] + \underbrace{\mathbb{E}[\epsilon^2]}_{=\sigma_\epsilon^2} + 2\mathbb{E}[(f(x) - \hat{f}(x))]\,\underbrace{\mathbb{E}[\epsilon]}_{=0} \\
&= \mathbb{E}[(f(x) - \hat{f}(x))^2] + \sigma_\epsilon^2
\end{aligned}
$$

$$
\begin{aligned}
\mathbb{E}[(f(x)-\hat{f}(x))^2] &= \mathbb{E}\left[\left((f(x)-\mathbb{E}[\hat{f}(x)])-(\hat{f}(x)-\mathbb{E}[\hat{f}(x)])\right)^2\right] \\
&= \mathbb{E}\left[\left(\mathbb{E}[\hat{f}(x)]-f(x)\right)^2\right]+\mathbb{E}\left[\left(\hat{f}(x)-\mathbb{E}[\hat{f}(x)]\right)^2\right] \\
&\quad -2\mathbb{E}\left[\left(f(x)-\mathbb{E}[\hat{f}(x)]\right)\left(\hat{f}(x)-\mathbb{E}[\hat{f}(x)]\right)\right] \\
&= (\underbrace{\mathbb{E}[\hat{f}(x)]-f(x)}_{=\mathrm{bias}[\hat{f}(x)]})^2+\underbrace{\mathbb{E}\left[\left(\hat{f}(x)-\mathbb{E}[\hat{f}(x)]\right)^2\right]}_{=\mathrm{var}(\hat{f}(x))} \\
&\quad -2\left(f(x)-\mathbb{E}[\hat{f}(x)]\right)\mathbb{E}\left[\left(\hat{f}(x)-\mathbb{E}[\hat{f}(x)]\right)\right] \\
&= \mathrm{bias}[\hat{f}(x)]^2+\mathrm{var}(\hat{f}(x)) \\
&\quad -2\left(f(x)-\mathbb{E}[\hat{f}(x)]\right)\left(\mathbb{E}[\hat{f}(x)]-\mathbb{E}[\hat{f}(x)]\right) \\
&= \mathrm{bias}[\hat{f}(x)]^2+\mathrm{var}(\hat{f}(x))
\end{aligned}
$$

$$
\mathbb{E}[(y-\hat{f}(x))^2] = \mathrm{bias}[\hat{f}(x)]^2+\mathrm{var}(\hat{f}(x))+\sigma_\epsilon^2
$$

Bias-Variance Trade-off

- **The Bias/Variance Tradeoff**
- The objective is to reduce the error E to the minimum. This can be done by modifying the terms of the mean square error.
- From the equation, we see that we could only modify the bias and the variance terms.
- Bias arises when we generalize relationships using a function, while variance arises when there are multiple samples or input.
- One way to reduce the error is to reduce the bias and the variance terms.
- However, we cannot reduce both terms simultaneously, since reducing one term leads to increase in the other term. This is the idea of bias variance trade/off.

Bias-Variance Trade-off

- **Relationship with Underfitting and Overfitting**
- A good model should do one of two things
  - Capture the patterns in the given training data set
  - Correctly compute the output for a new instance
- The model should be complete enough to represent the data, but the more complex the model, the better it represents the training data.
- However, there is a limit to how complex the model can get.
- If the model is too complex, then it will pick up specific random features (noise or example) in the training data set.
- If the model is not complex enough, then it might miss out on important dynamics of the data given.

Bias-Variance Trade-off

- **Illustration of Bias-Variance Trade-off**
- Assuming you have several training data sets for the same population:
  - Training Data 1
  - Training Data 2
  - Training Data 3

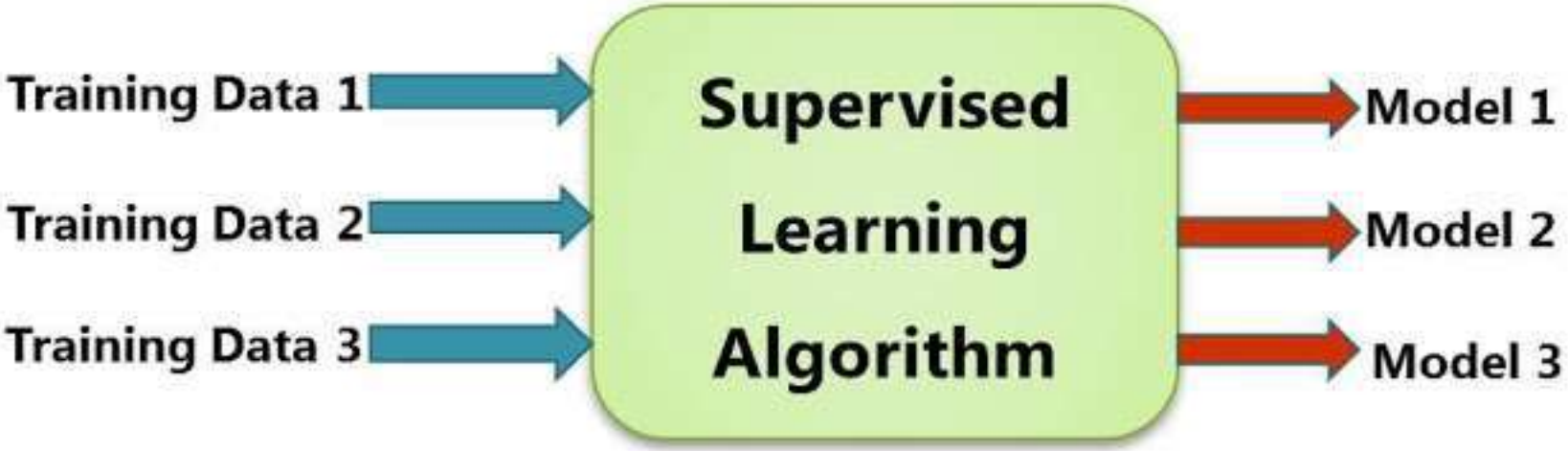


Figure 1: Supervised Learning algorithm

Bias-Variance Trade-off

- **Illustration of Bias-Variance Trade-off**
- These three data sets are passed through the same supervised learning algorithm which produces three models.
  - Model 1
  - Model 2
  - Model 3
- Now, let say we want to predict the output of a new input x,
- The three models should be able to produce the same output for the same new instance. But when you pass x into each of the models, instead of getting the same output, you get a different output(y_1, y_2 and y_3) for the same x. This is illustrated in Figure 2 in the next slide.

Bias-Variance Trade-off

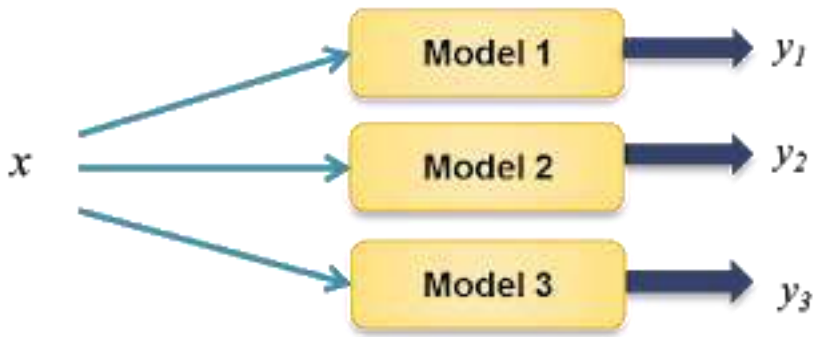


Figure 2: High Variance Error

- The problem here is that that the model have become too specific that it cannot capture the correct output for a new value for x.
- In this case, the algorithm is said to have *high-variance error*. Which results in a problem of overfitting.

Bias-Variance Trade-off

- **Illustration of Bias-Variance Trade-off**
- Let's also assume that, you pass different values of x *(*x_1, x_2 *and* x_3*)* into the same model.
- Instead of getting different outputs, you get the same output y.
- In this case, the algorithm is said to have *high bias error*, which results in a problem of underfitting. This is illustrated in Figure 3 in the next slide:

Bias-Variance Trade-off

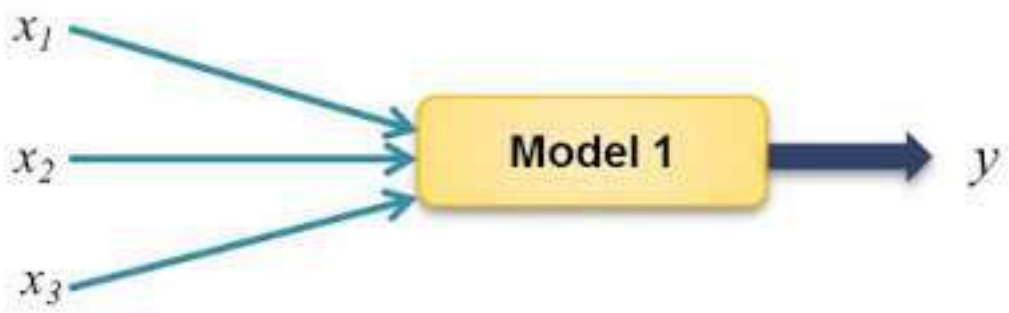


Figure 3: High Bias Error

- *High variance* means that the algorithm have become too specific.
- *High bias* means that the algorithm have failed to understand the pattern in the input data.
- It's generally not possible to minimize both errors simultaneously, since high bias would always means low variance, whereas low bias would always mean high variance.
- Finding a trade-off between the two extremes is known as *Bias/Variance Tradeoff*.

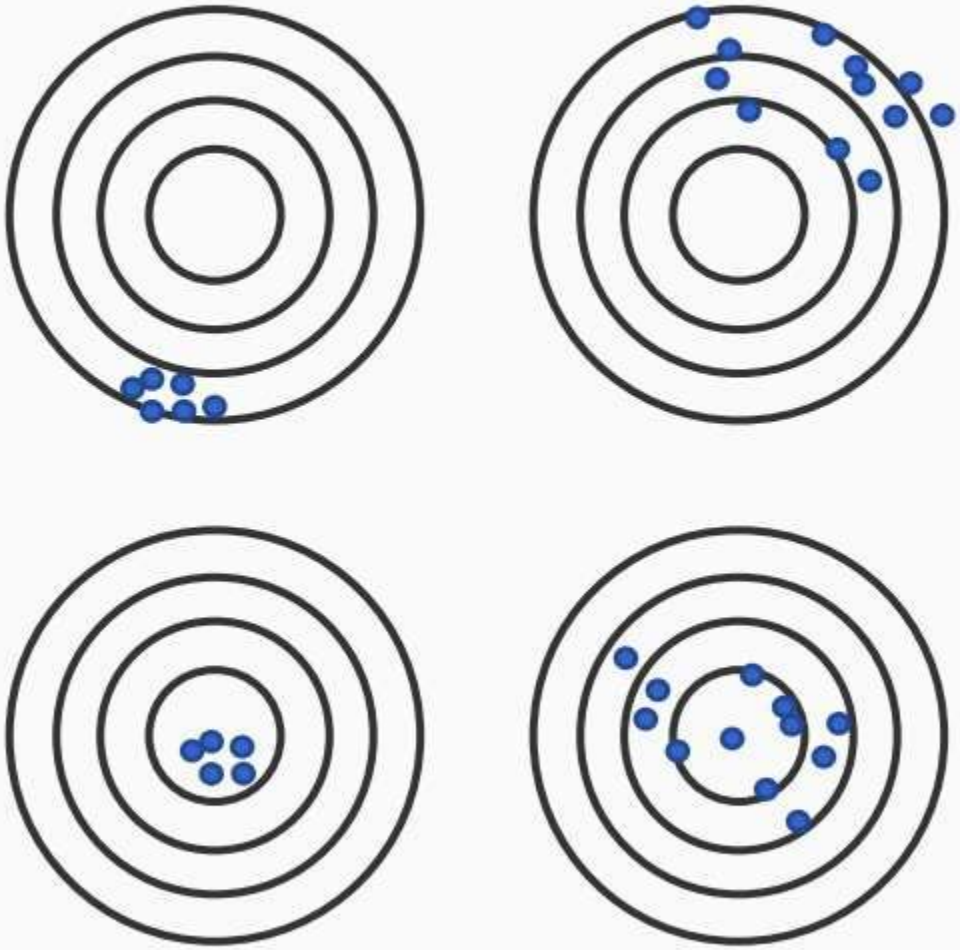
Let ’s play darts

Suppose the true concept is the center

Each dot is a model that is learned from a different dataset

High bias

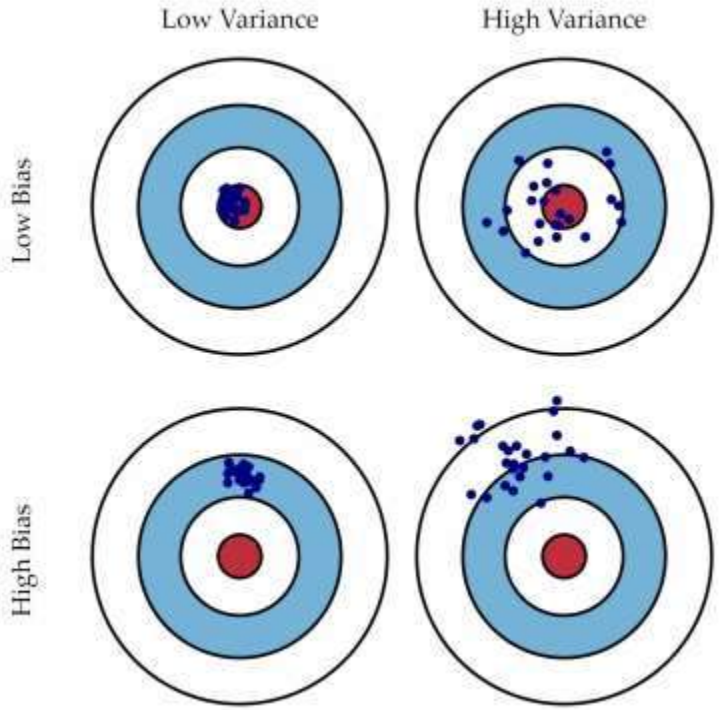
Low bias



Low variance

High variance

Bias-Variance-Tradeoff: Crossvalidation & Learning Curves

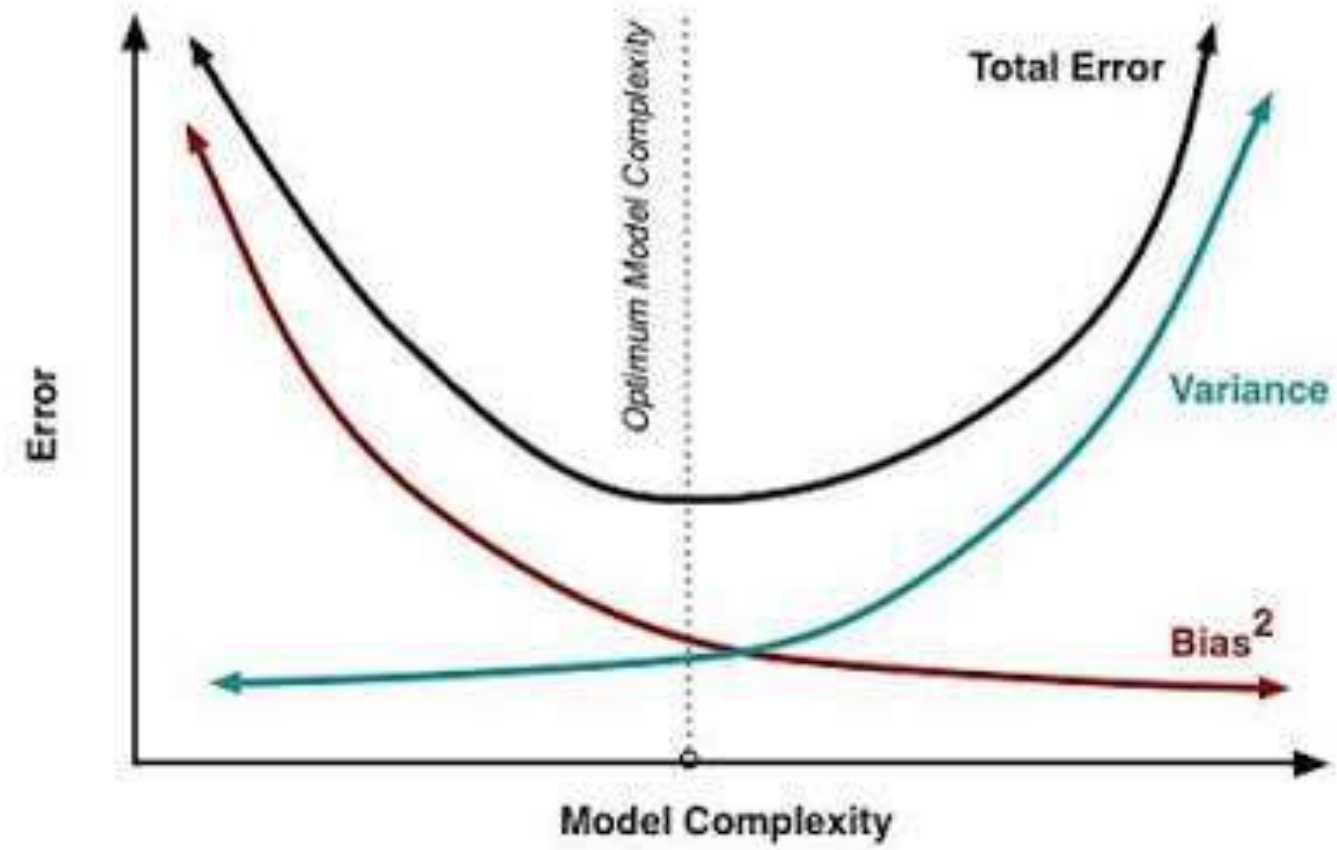


$$\mathrm{Bias}\left[\hat{f}(x)\right] = \mathrm{E}\left[\hat{f}(x) - f(x)\right]$$

$$\mathrm{Var}\left[\hat{f}(x)\right] = \mathrm{E}\left[\left(\hat{f}(x) - \mathrm{E}[\hat{f}(x)]\right)^2\right]$$

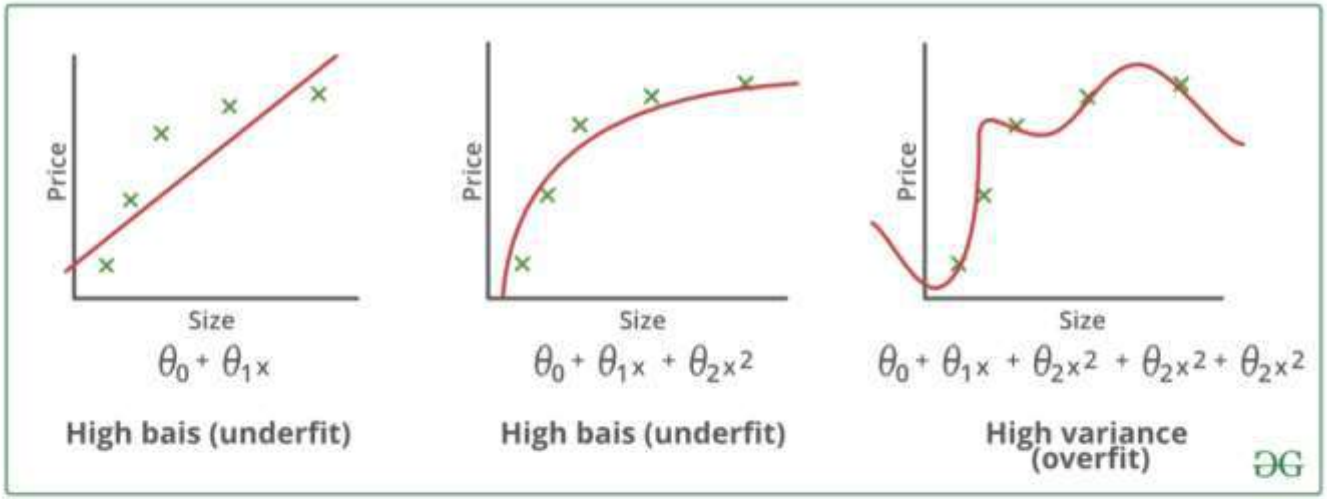
Bias-Variance Trade-off

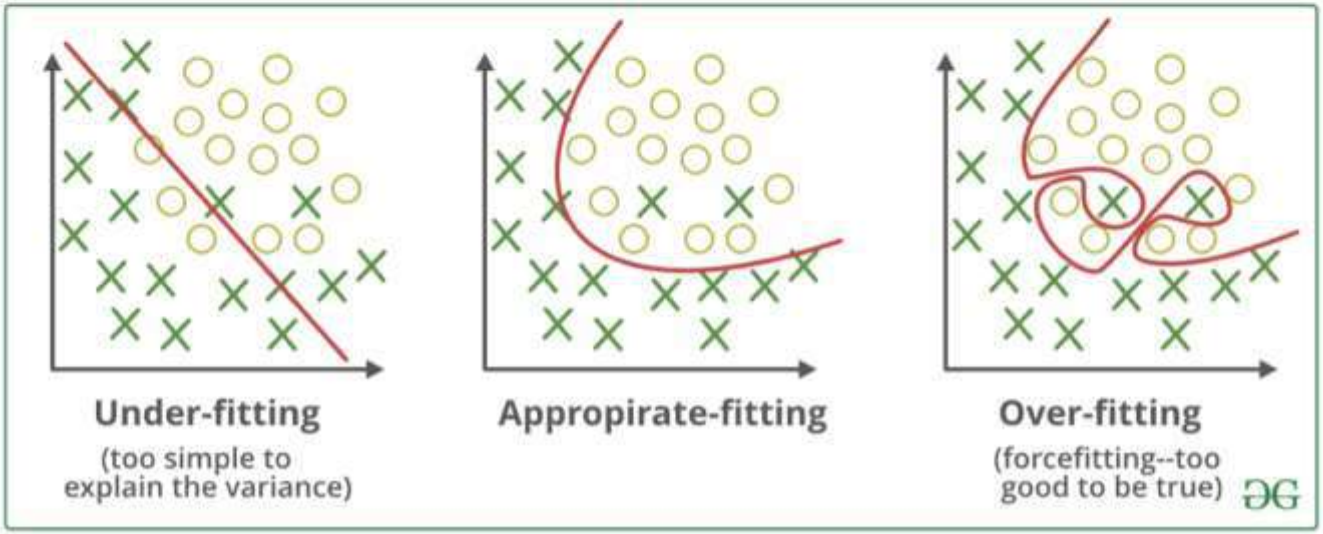
- The graph in Figure 3 is a typical plot of the bias/variance trade-off

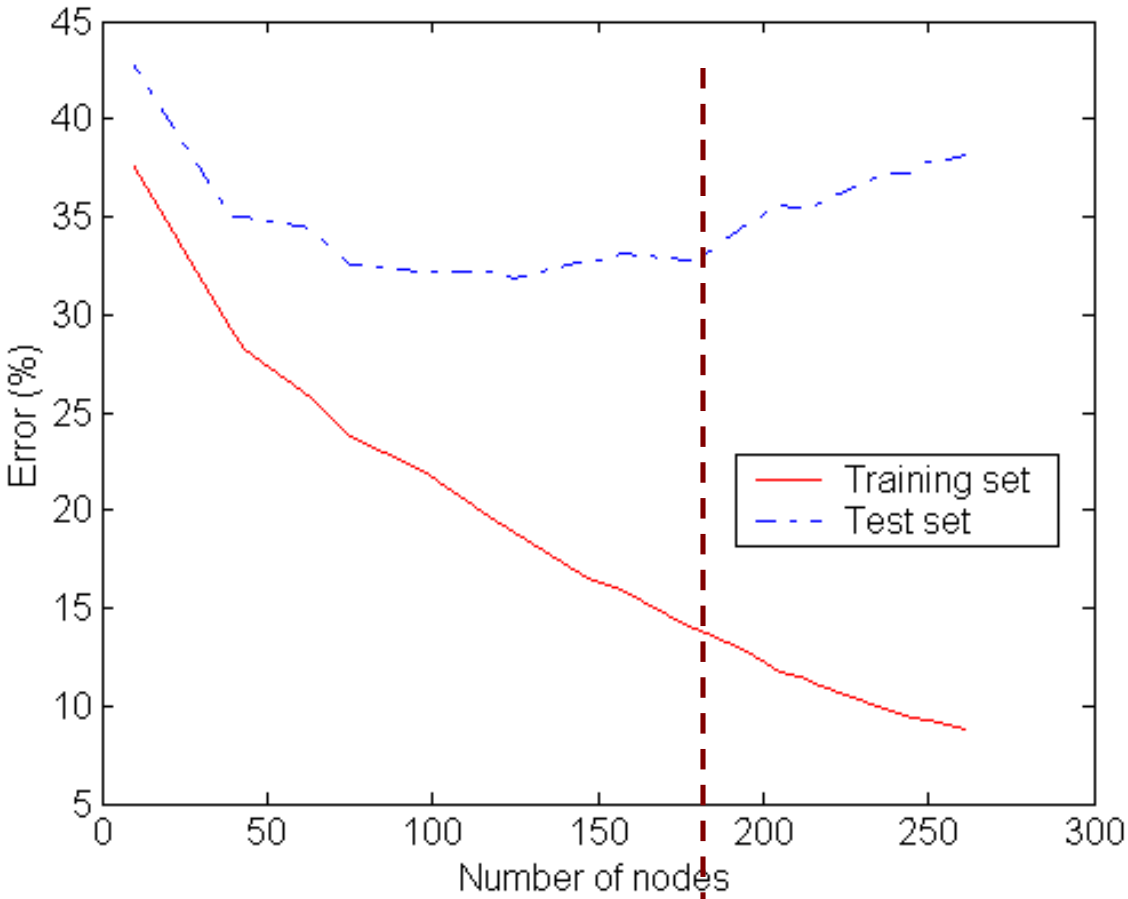


Bias-Variance Trade-off

- The bias/variance graph shows a plot of Error against Model Complexity. It also shows:
- *Relationship of variance and Model Complexity:* As we increase the variance, the model complex increases.
- *Relationship of bias and Model Complexity*: As the bias increase, the model complexity reduces
- *Relationship of variance and Error*: As the variance increases, the error increases.
- *Relationship of bias and Error*: As the bias increases, the error increases.


Price
Size
$\theta_0 + \theta_1 x$
High bais (underfit)
Price
Size
$\theta_0 + \theta_1 x + \theta_2 x^2$
High bais (underfit)
Price
Size
$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_2 x^2 + \theta_2 x^2$
High variance
(overfit)


Under-fitting
(too simple to
explain the variance)
Appropirate-fitting
Over-fitting
(forcefitting--too
good to be true)



Complexity of a Decision Tree := number of nodes It uses

Complexity of the Used Model

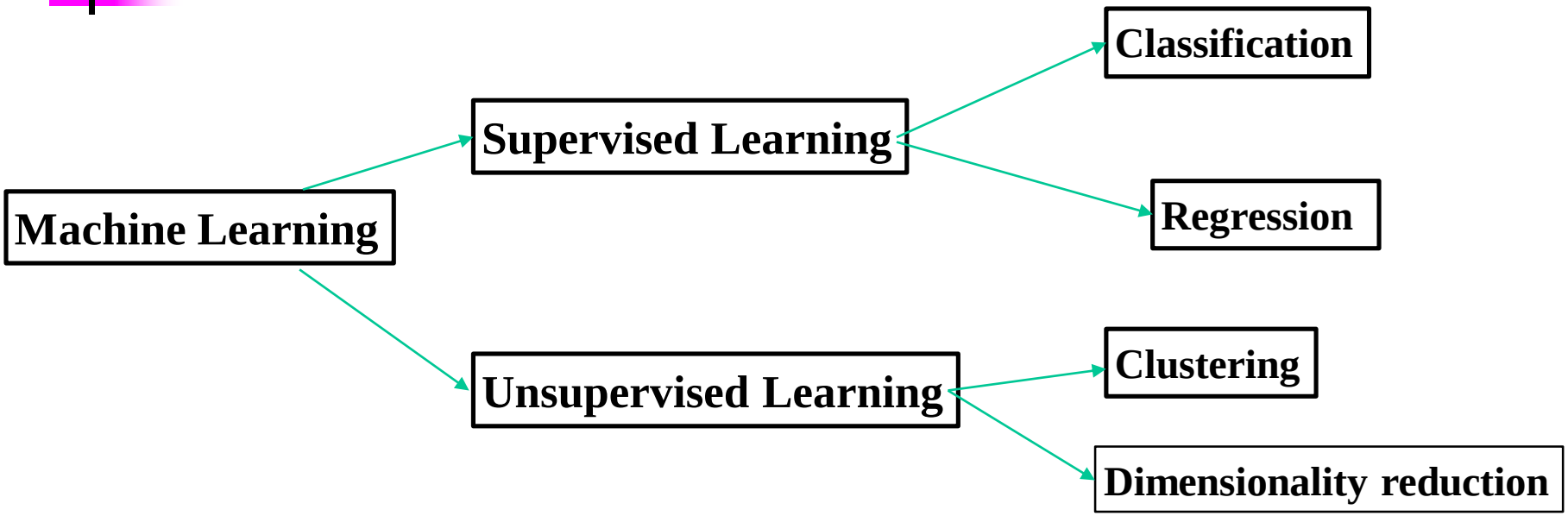
Underfitting: when model is too simple, both training and test errors are large

Overfitting: when model is too complex and test errors are large although training errors are small.

Managing of bias and variance

- Ensemble methods reduce variance
  - Multiple classifiers are combined
  - Eg: Bagging, boosting
- Decision trees of a given depth
  - Increasing depth decreases bias, increases variance
- SVMs
  - Higher degree polynomial kernels decreases bias, increases variance
  - Stronger regularization increases bias, decreases variance
- Neural networks
  - Deeper models can increase variance, but decrease bias
- K nearest neighbors
  - Increasing k generally increases bias, reduces variance

Types of Machine Learning Problems



- Supervised Learning: develop predictive models from labelled data (i.e. data with classes or targets)
- Unsupervised learning: describe hidden structure of unlabelled data
  - Clustering: Group similar data into categories (clusters) based only on input data
  - Dimensionality reduction: Reduce input variables of a dataset to a smaller set of variables (structure of dataset)

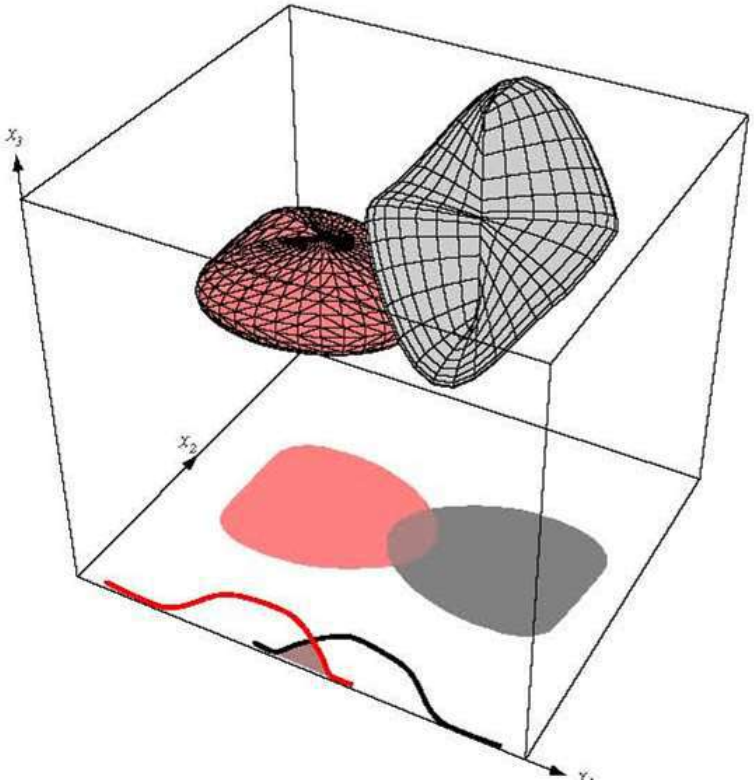
Dimensionality Reduction

Data Dimensionality

- From a theoretical point of view, increasing the number of features should lead to better performance.

- In practice, the inclusion of more features leads to worse performance (i.e., curse of dimensionality).

- The number of training examples required increases exponentially with dimensionality.



Why Reduce Dimensionality?

1. Reduces time complexity: Less computation
2. Reduces space complexity: Less parameters
3. Saves the cost of aquiring the feature
4. Simpler models are more robust
5. Easier to interpret; simpler explanation
6. Data visualization (structure, groups, outliers, etc) if plotted in 2 or 3 dimensions

Dimensionality Reduction

- Significant improvements can be achieved by first mapping (projecting) the data into a *lower-dimensional* space.

$$x = \begin{bmatrix} a_1 \\ a_2 \\ \dots \\ a_N \end{bmatrix} --> reduce\ dimensionality --> y = \begin{bmatrix} b_1 \\ b_2 \\ \dots \\ b_K \end{bmatrix} \ (K << N)$$

- Dimensionality can be reduced by:
  - Combining features using a linear or non-linear transformations.
  - Selecting a subset of features (i.e., feature selection).

Dimensionality Reduction (cont'd)

- Linear combinations are particularly attractive because they are simpler to compute and analytically tractable.

- Given $x \in R^N$, the goal is to find an N x K matrix U such that:

$$y = U^T x \in R^K \text{ where } K << N \text{ (projection)}$$

$$x = \begin{bmatrix} a_1 \\ a_2 \\ \dots \\ a_N \end{bmatrix} \xrightarrow{U^T} \text{reduce dimensionality} \longrightarrow y = \begin{bmatrix} b_1 \\ b_2 \\ \dots \\ b_K \end{bmatrix} \quad (K << N)$$

Dimensionality Reduction (cont'd)

- <u>Idea:</u> represent data in terms of basis vectors in a lower dimensional space which is embedded within the original space.

  (1) Higher-dimensional space representation:

$$x = a_1 v_1 + a_2 v_2 + \cdots + a_N v_N \qquad x = \begin{bmatrix} a_1 \\ a_2 \\ \cdots \\ a_N \end{bmatrix}$$

$v_1, v_2, \ldots, v_N$ is a basis of the N-dimensional space

  (2) Lower-dimensional sub-space representation:

$$\hat{x} = b_1 u_1 + b_2 u_2 + \cdots + b_K u_K \qquad y = \begin{bmatrix} b_1 \\ b_2 \\ \cdots \\ b_K \end{bmatrix}$$

$u_1, u_2, \ldots, u_K$ is a basis of the K-dimensional space

Dimensionality Reduction (cont'd)

- Classical approaches for finding an optimal linear transformation:
  - Principal Components Analysis (PCA): Seeks a projection that preserves as much **information** in the data as possible (in a least-squares sense).
  - Linear Discriminant Analysis (LDA): Seeks a projection that best **separates** the data (in a least-squares sense).

Principal Component Analysis (PCA)

- Dimensionality reduction implies information loss; PCA preserves as much information as possible by minimizing the "reconstruction" error:

$$\|x - \hat{x}\|$$

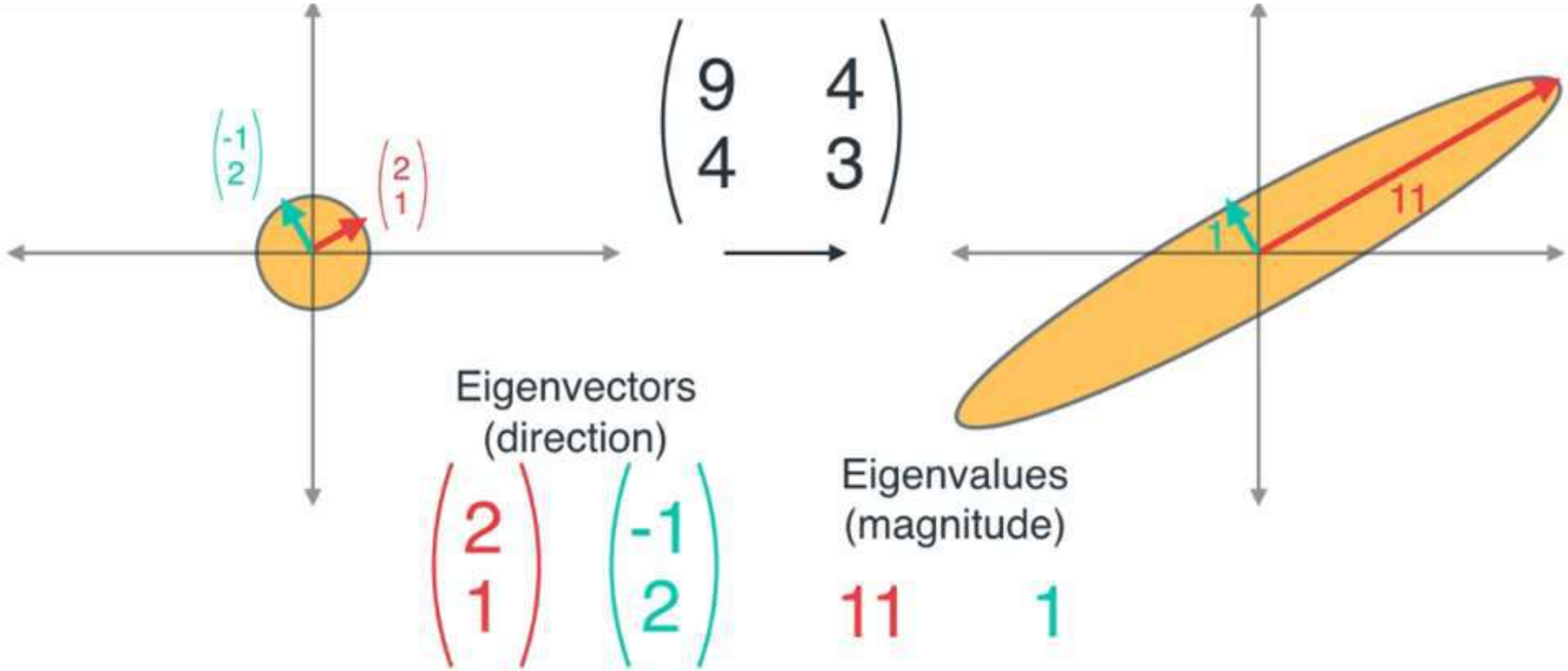
$$x = a_1 v_1 + a_2 v_2 + \cdots + a_N v_N$$

$$\hat{x} = b_1 u_1 + b_2 u_2 + \cdots + b_K u_K$$

- How should we determine the "best" lower dimensional space (i.e., basis u_1, u_2, …,u_k)?

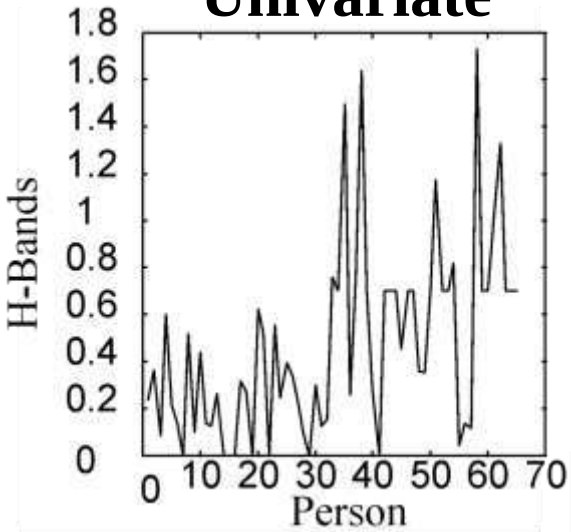
By the "best" eigenvectors of the covariance matrix of the data (i.e., corresponding to the "largest" eigenvalues – also called "principal components")

Linear Transformation

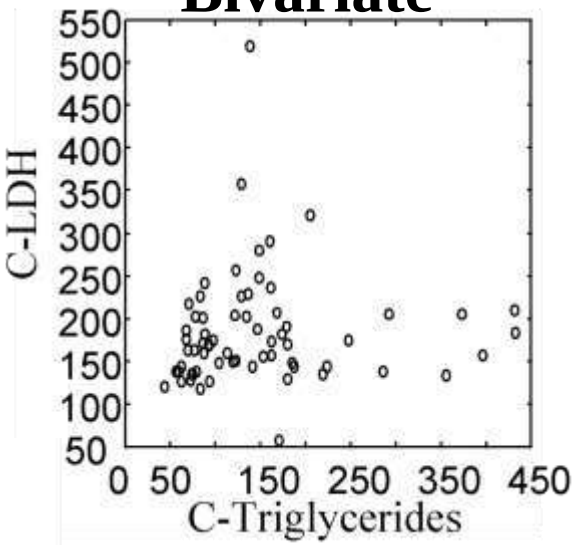


Data Presentation

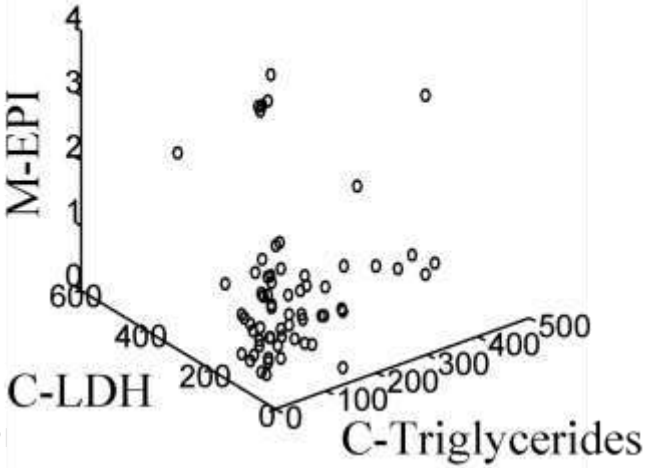
Univariate



Bivariate



Trivariate



PCA: *General*

From k original variables: $x_1, x_2, \ldots, x_k$:

Produce k new variables: $y_1, y_2, \ldots, y_k$:

$$y_1 = u_{11}x_1 + u_{12}x_2 + \ldots + u_{1k}x_k$$

$$y_2 = u_{21}x_1 + u_{22}x_2 + \ldots + u_{2k}x_k$$

$$\ldots$$

$$y_k = u_{k1}x_1 + u_{k2}x_2 + \ldots + u_{kk}x_k$$

PCA: *General*

From k original variables: $x_1, x_2, \ldots, x_k$:

Produce k new variables: $y_1, y_2, \ldots, y_k$:

$y_1 = u_{11}x_1 + u_{12}x_2 + \ldots + u_{1k}x_k$

$y_2 = u_{21}x_1 + u_{22}x_2 + \ldots + u_{2k}x_k$

...

$y_k = u_{k1}x_1 + u_{k2}x_2 + \ldots + u_{kk}x_k$

such that:

y_k's are uncorrelated (orthogonal)

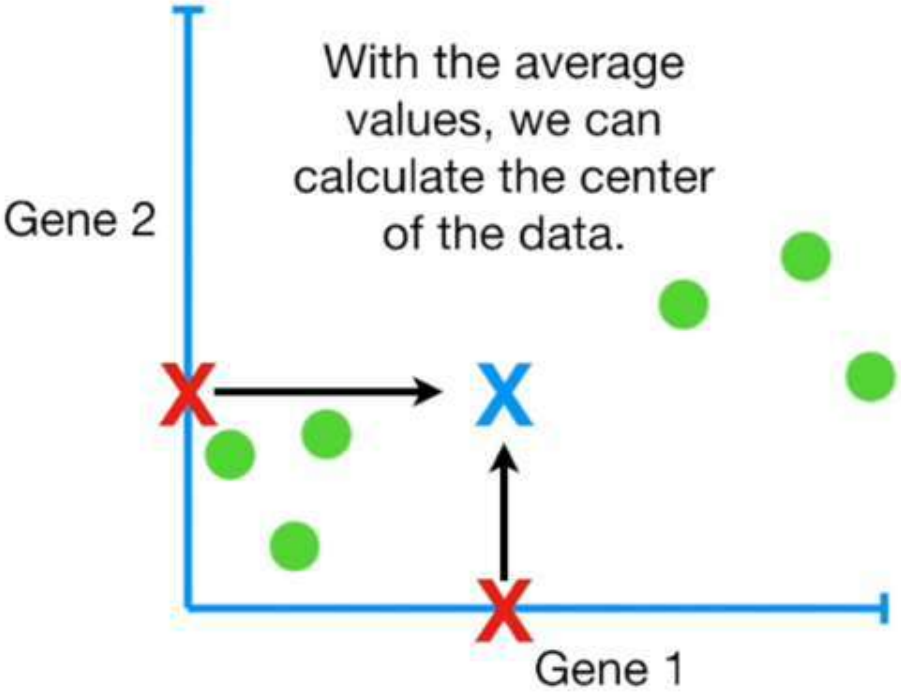
y_1 explains as much as possible of original variance in data set

y_2 explains as much as possible of remaining variance

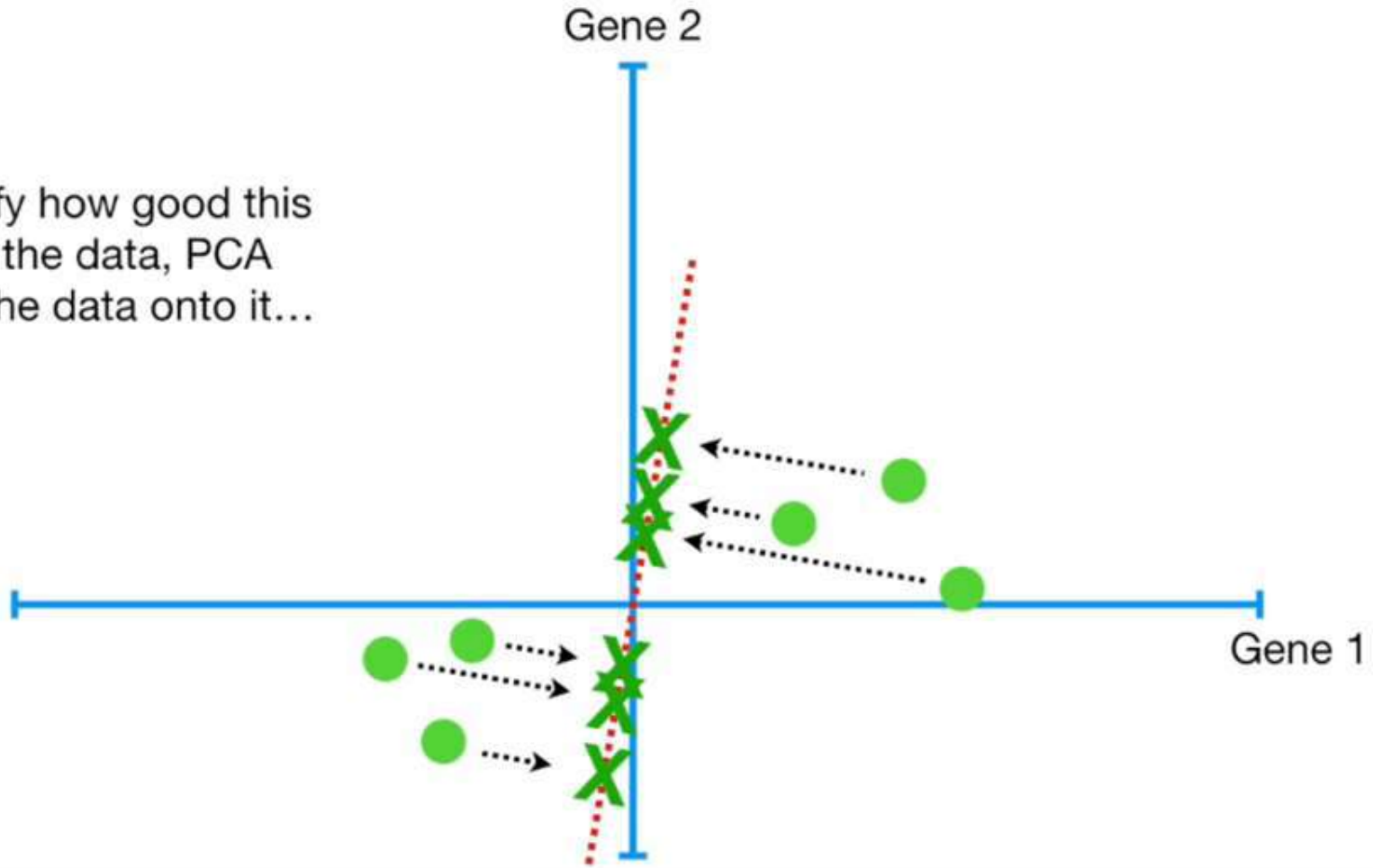
etc.

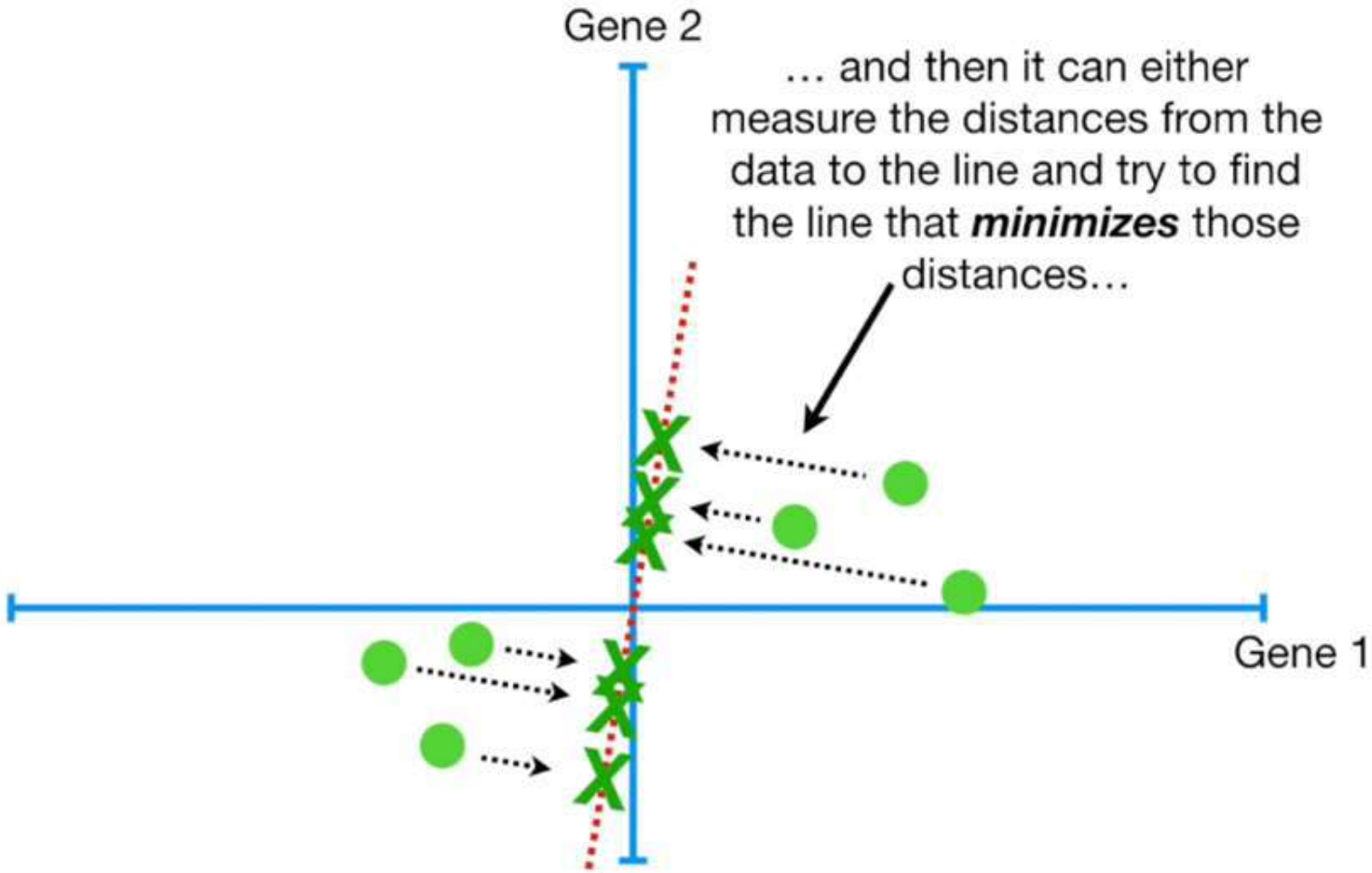
Example Data Set

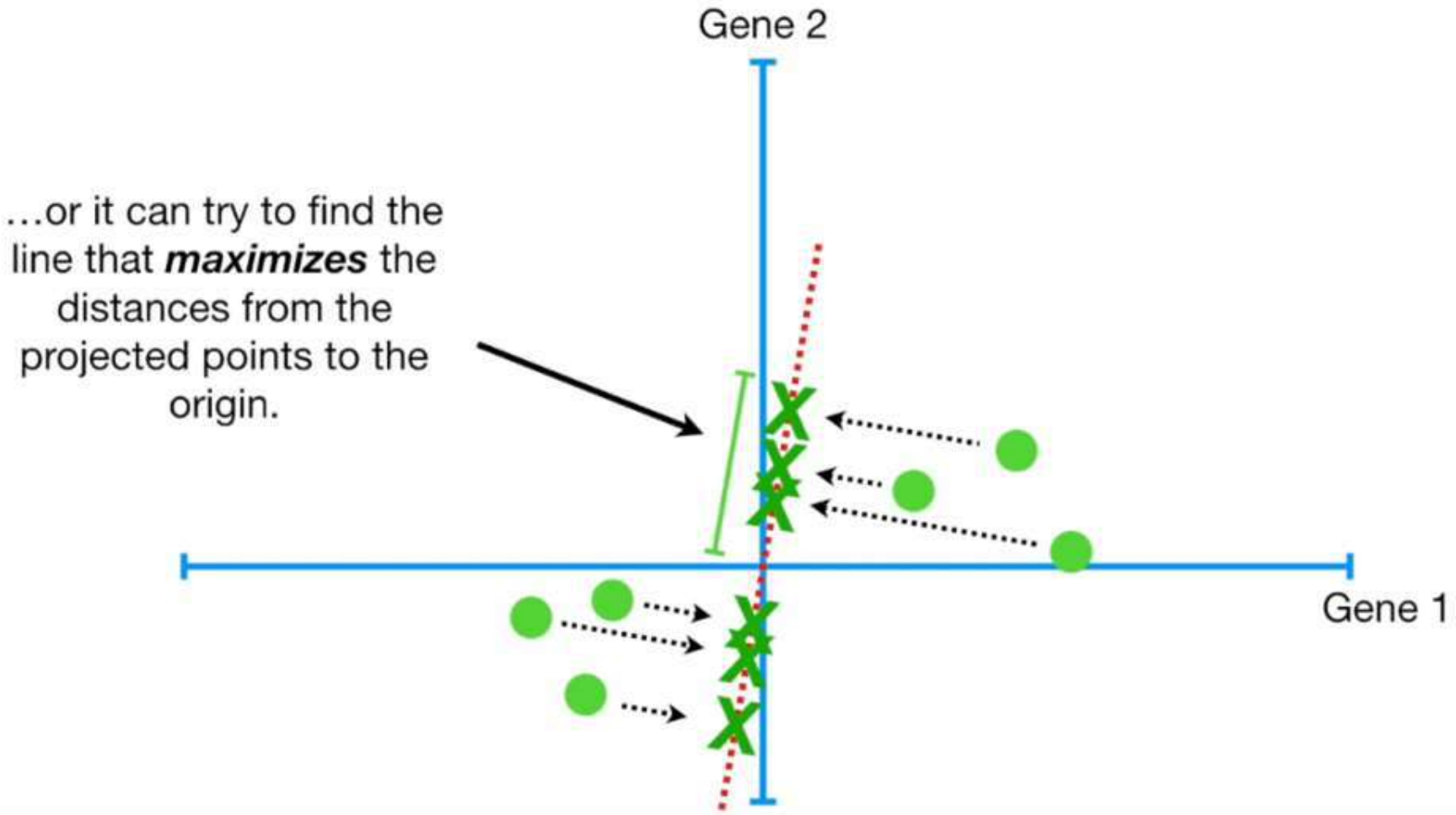
	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	10	11	8	3	2	1
Gene 2	6	4	5	3	2.8	1



To quantify how good this line fits the data, PCA projects the data onto it…



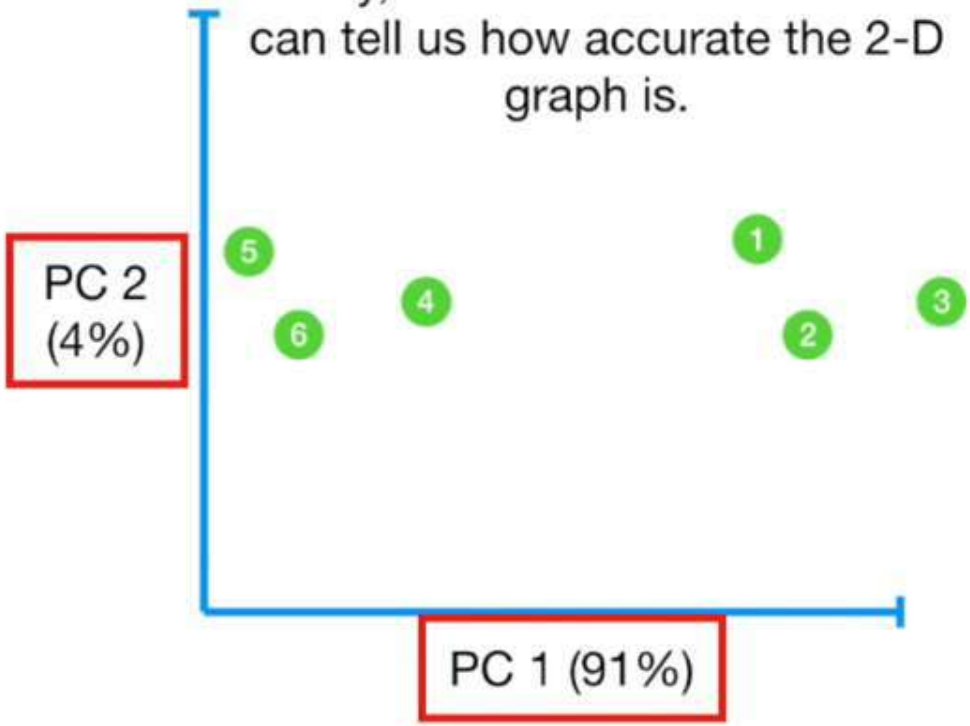

Gene 2
… and then it can either measure the distances from the data to the line and try to find the line that ***minimizes*** those distances…
Gene 1


Gene 2
…or it can try to find the line that ***maximizes*** the distances from the projected points to the origin.
Gene 1

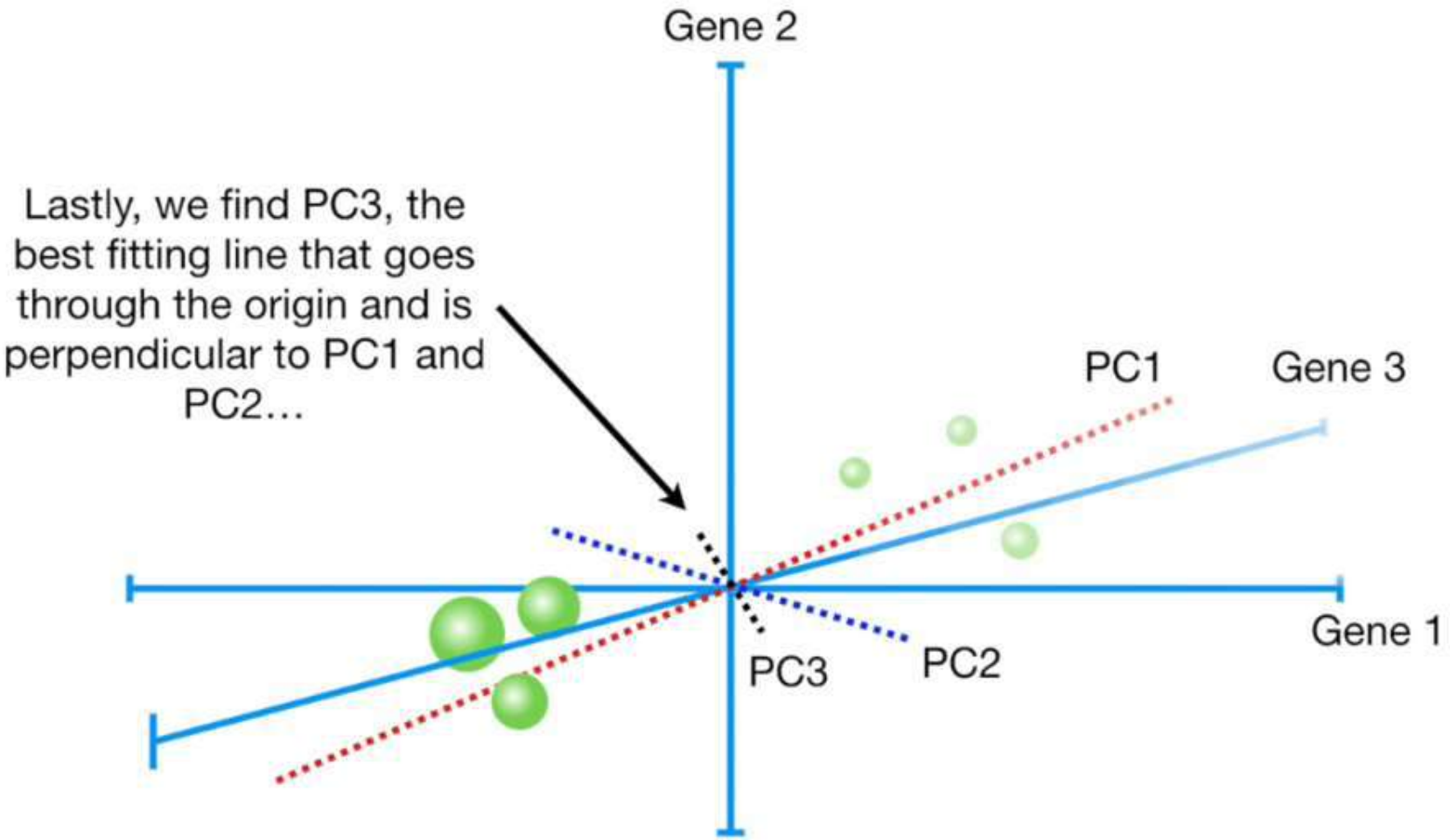
Data Presentation and PCA

	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	10	11	8	3	2	1
Gene 2	6	4	5	3	2.8	1
Gene 3	12	9	10	2.5	1.3	2
Gene 4	5	7	6	2	4	7

Lastly, we'll talk about how PCA can tell us how accurate the 2-D graph is.



Data Presentation



PCA - Steps

- Suppose $x_1, x_2, \ldots, x_N$ are $N \times 1$ vectors

Step 1: $\bar{x} = \frac{1}{N} \sum_{i=1}^{N} x_i$

Step 2: subtract the mean: $\Phi_i = x_i - \bar{x}$ (i.e., center at zero)

Step 3: form the matrix $A = [\Phi_1 \ \Phi_2 \ \cdots \ \Phi_N]$ ($N \times N$ matrix), then compute:

$$C = \frac{1}{N} \sum_{n=1}^{N} \Phi_n \Phi_n^T = \frac{1}{M} A A^T$$

(sample **covariance** matrix, $N \times N$, characterizes the *scatter* of the data)

Step 4: compute the eigenvalues of C: $\boldsymbol{\lambda}_1 > \boldsymbol{\lambda}_2 > \cdots > \boldsymbol{\lambda}_N$

Step 5: compute the eigenvectors of C: $u_1, u_2, \ldots, u_N$

PCA – Steps (cont'd)

an orthogonal basis

- Since C is symmetric, $u_1, u_2, \ldots, u_N$ form ~~a basis~~, (i.e., any vector x or actually $(x - \bar{x})$, can be written as a linear combination of the eigenvectors):

$$x - \bar{x} = b_1 u_1 + b_2 u_2 + \cdots + b_N u_N = \sum_{i=1}^{N} b_i u_i \qquad \text{where } b_i = \frac{(x - \bar{x}).u_i}{(u_i.u_i)}$$

Step 6: **(dimensionality reduction step)** keep only the terms corresponding to the K largest eigenvalues:

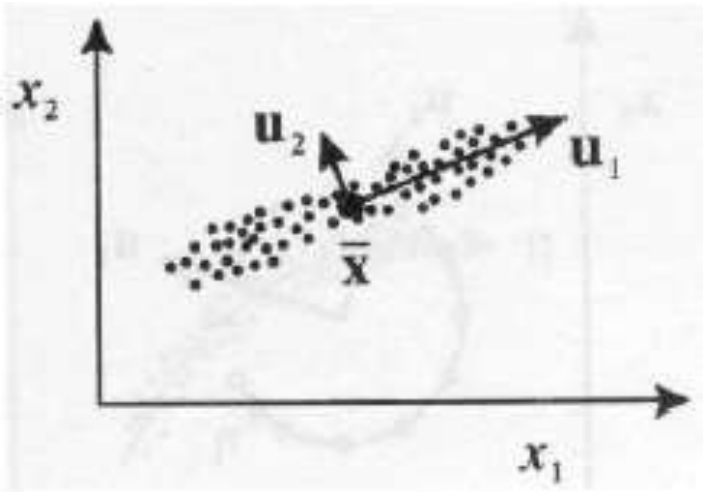
$$\hat{x} - \bar{x} = \sum_{i=1}^{K} b_i u_i \text{ where } K << N$$

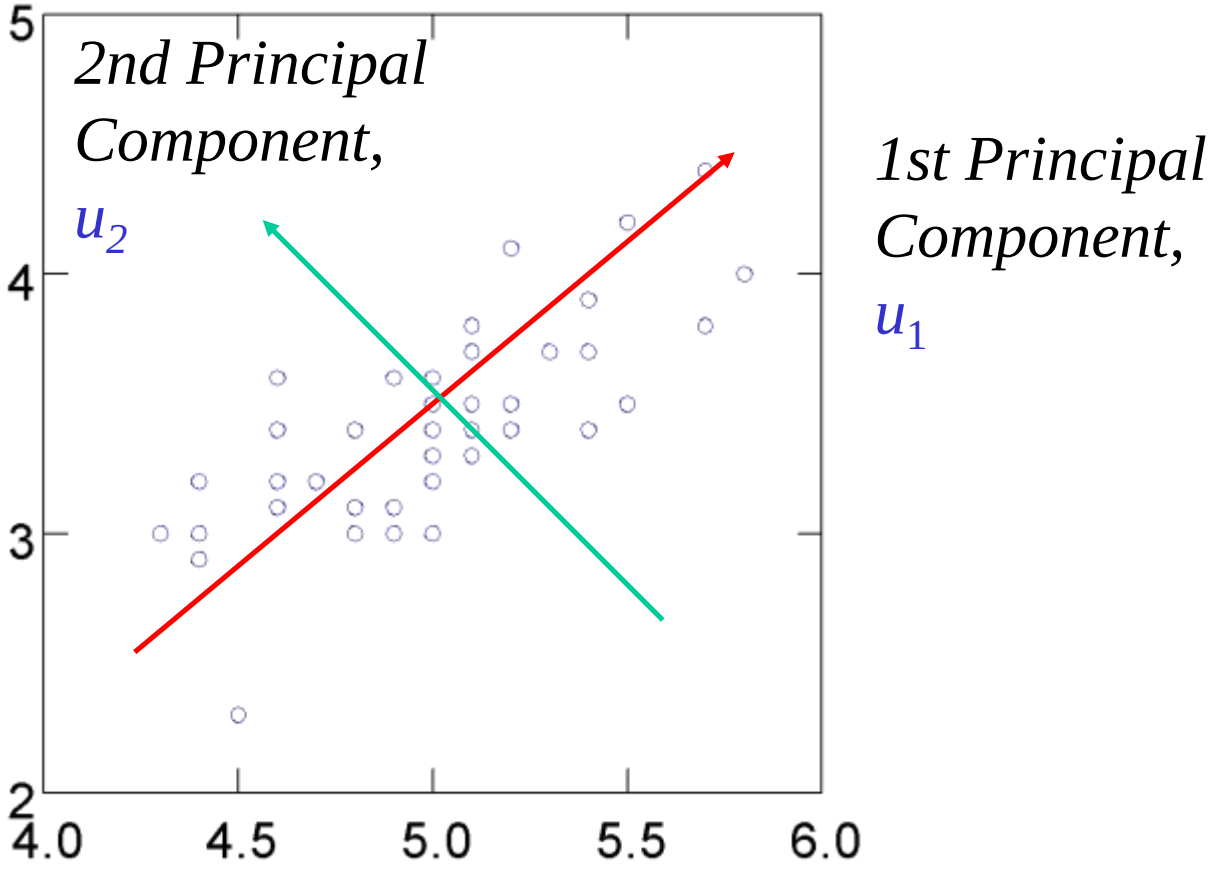
- The representation of $\hat{x} - \bar{x}$ into the basis $u_1, u_2, \ldots, u_K$ is thus

$$\begin{bmatrix} b_1 \\ b_2 \\ \ldots \\ b_K \end{bmatrix}$$

Geometric interpretation

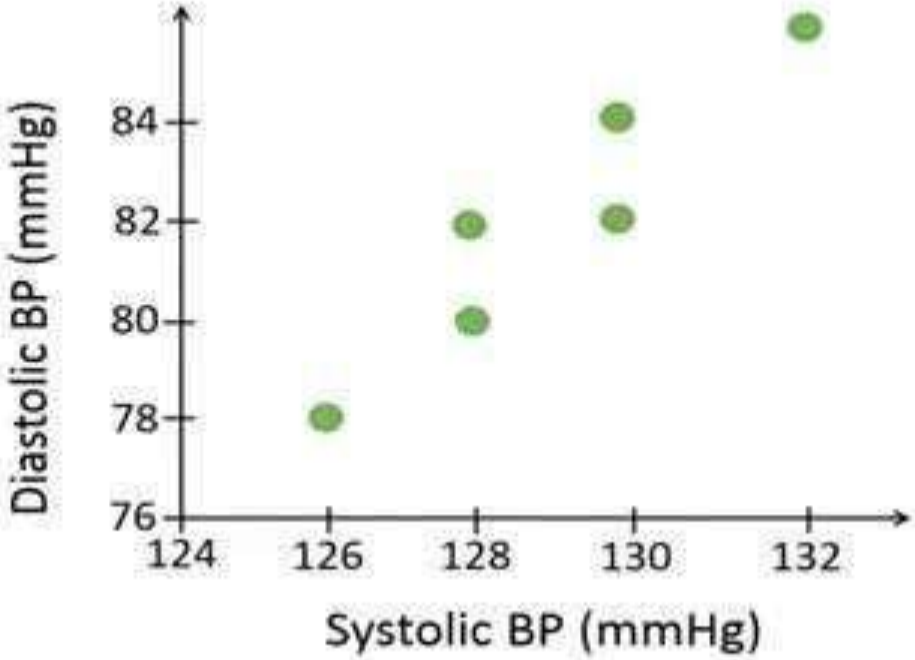
- PCA projects the data along the directions where the data varies most.
- These directions are determined by the eigenvectors of the covariance matrix corresponding to the largest eigenvalues.
- The magnitude of the eigenvalues corresponds to the variance of the data along the eigenvector directions.





u1 explains as much as possible of original variance in data set
u2 explains as much as possible of remaining variance

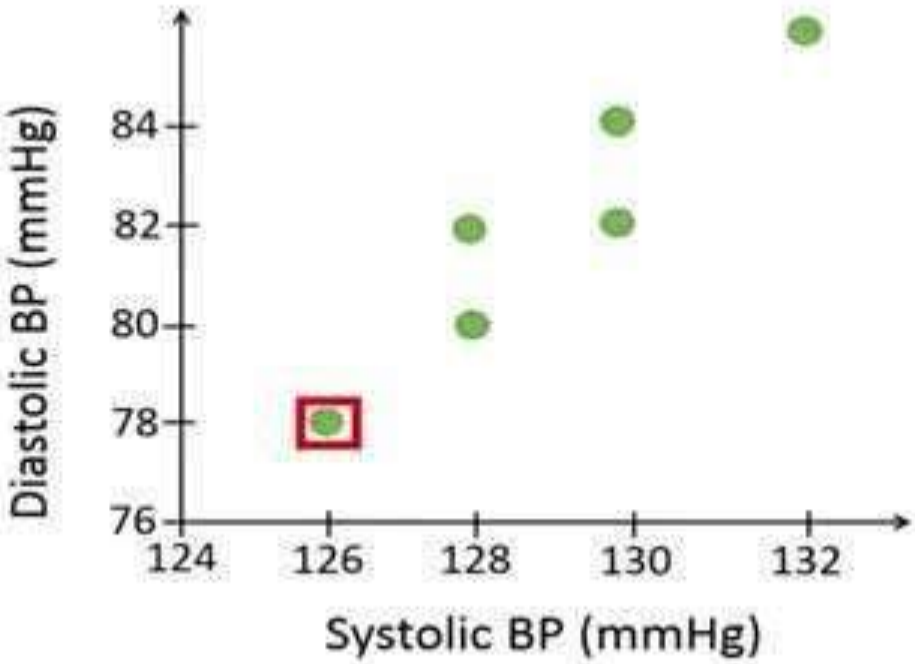
Example data



Systolic BP	Diastolic BP
126	78
128	80
128	82
130	82
130	84
132	86

To explain how the PCA works, we will use the following example data. We will use PCA to combine the two blood pressure variables into just one variable based on data from six individuals.

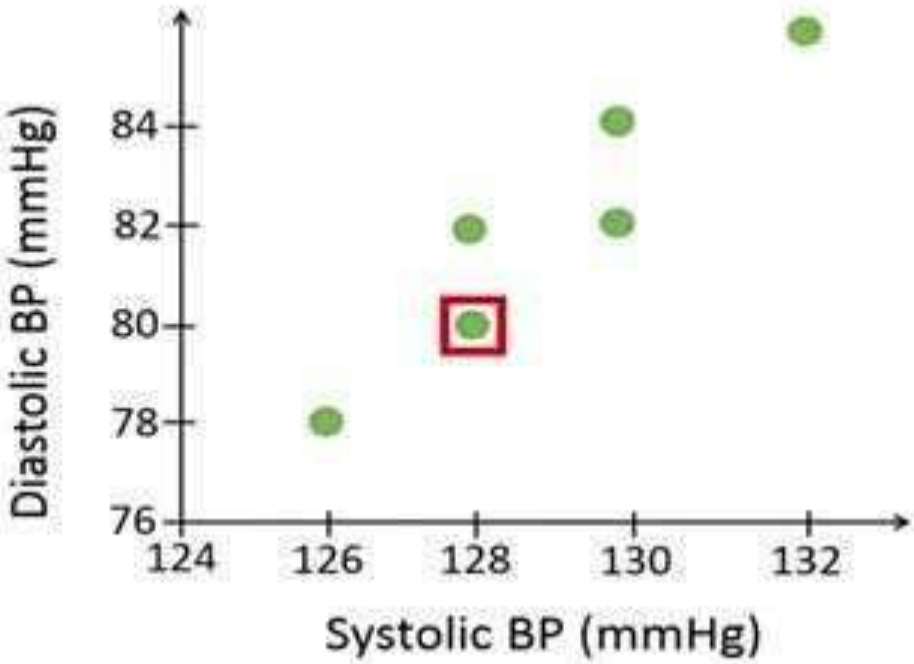
Example data



Systolic BP	Diastolic BP
126	78
128	80
128	82
130	82
130	84
132	86

For example, person number one has a diastolic blood pressure of 78 and a systolic blood pressure of 126.

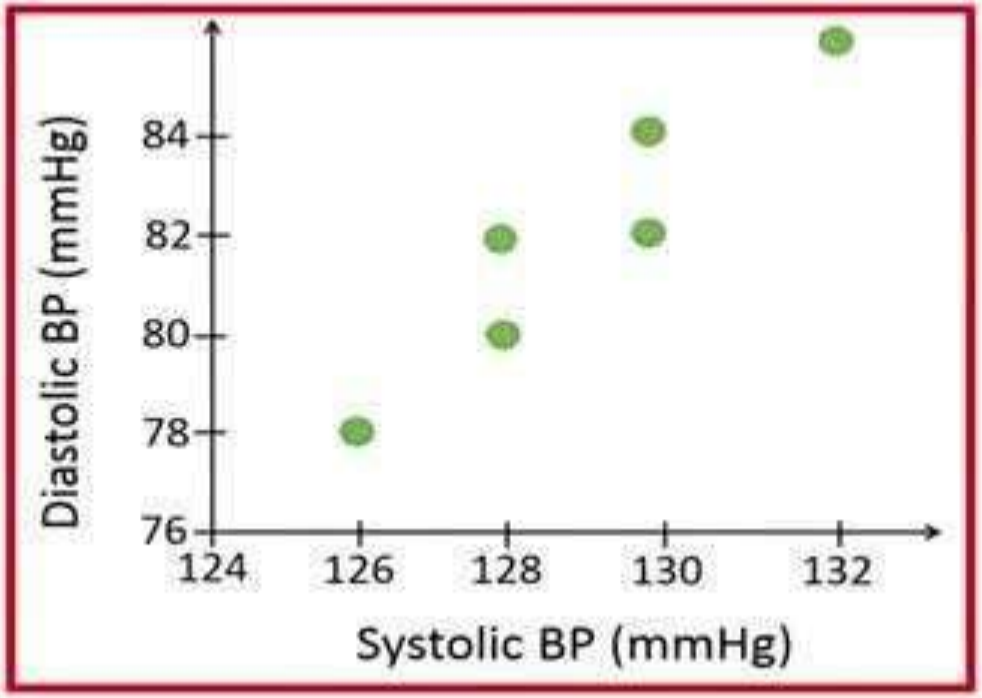
Example data



Systolic BP	Diastolic BP
126	78
128	80
128	82
130	82
130	84
132	86

whereas person number two has a diastolic blood pressure of 80 and a systolic blood pressure of 128, and so on.

Example data



Systolic BP	Diastolic BP
126	78
128	80
128	82
130	82
130	84
132	86

For this data set, it seems to be a strong positive correlation between the two variables.

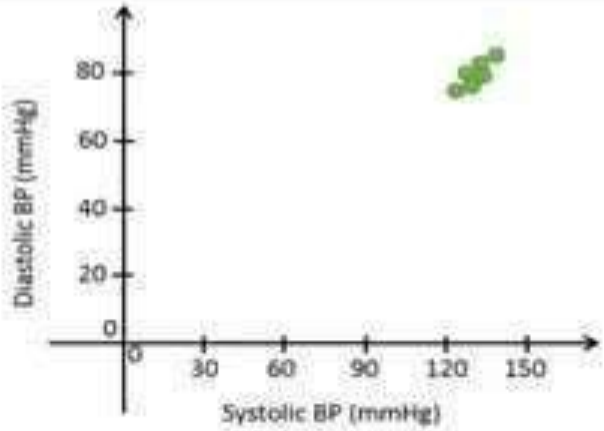
PCA

1. Center the data
2. Calculate the covariance matrix
3. Calculate eigenvalues of the covariance matrix
4. Calculate eigenvectors of the covariance matrix
5. Order the eigenvectors
6. Calculate the principal components

To compute a PCA, we can perform the following steps,

1. Center the data

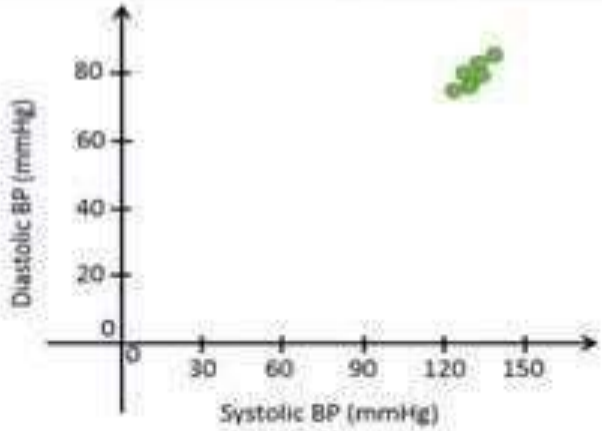
Systolic BP	Diastolic BP
126	78
128	80
128	82
130	82
130	84
132	86



Usually, one starts to center or standardize the data in the first step of the PCA analysis. In this case, we will only center the data, which means that we subtract all the values for each variable by its corresponding mean.

1. Center the data

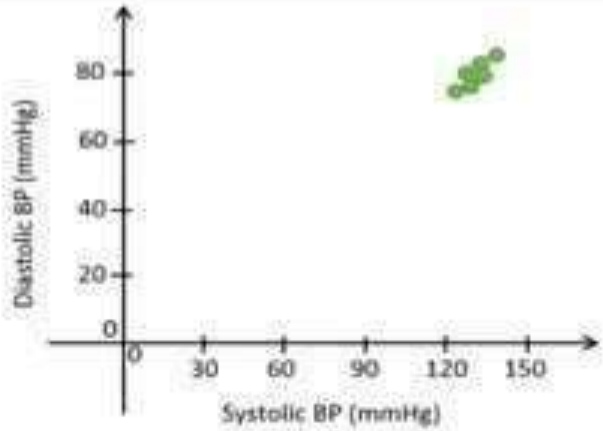
Systolic BP	Diastolic BP
126 -129 = -3	78
128 -129 = -1	80
128 -129 = -1	82
130 -129 = 1	82
130 -129 = 1	84
132 -129 = 3	86



We therefore subtract the mean systolic blood pressure,

1. Center the data

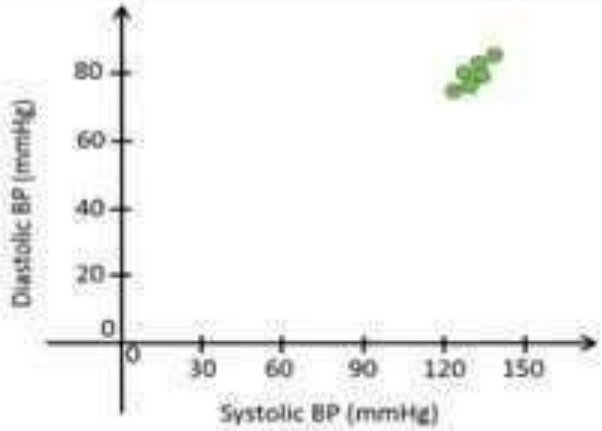
Systolic BP	Diastolic BP
126 -129 = -3	78 -82 = -4
128 -129 = -1	80 -82 = -2
128 -129 = -1	82 -82 = 0
130 -129 = 1	82 -82 = 0
130 -129 = 1	84 -82 = 2
132 -129 = 3	86 -82 = 4



We then do the same calculations for the diastolic blood pressure, which has a mean value of 82.

1. Center the data

Systolic BP	Diastolic BP
126 -129 = -3	78 -82 = -4
128 -129 = -1	80 -82 = -2
128 -129 = -1	82 -82 = 0
130 -129 = 1	82 -82 = 0
130 -129 = 1	84 -82 = 2
132 -129 = 3	86 -82 = 4



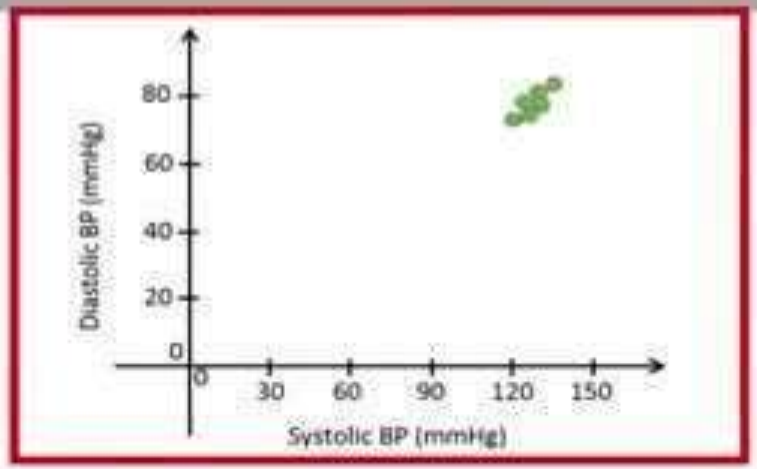
Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4

We can summarize the centered data in the following table.

1. Center the data

Systolic BP	Diastolic BP
126 -129 = -3	78 -82 = -4
128 -129 = -1	80 -82 = -2
128 -129 = -1	82 -82 = 0
130 -129 = 1	82 -82 = 0
130 -129 = 1	84 -82 = 2
132 -129 = 3	86 -82 = 4

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4

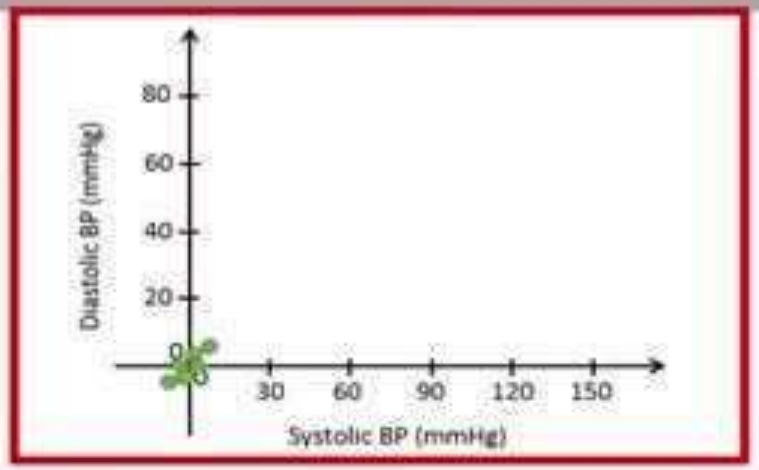


When we center the data, it means that we center the data points around the origin. Centering the data around the origin will help us later when we will rotate the data.

1. Center the data

Systolic BP	Diastolic BP
126 -129 = -3	78 -82 = -4
128 -129 = -1	80 -82 = -2
128 -129 = -1	82 -82 = 0
130 -129 = 1	82 -82 = 0
130 -129 = 1	84 -82 = 2
132 -129 = 3	86 -82 = 4

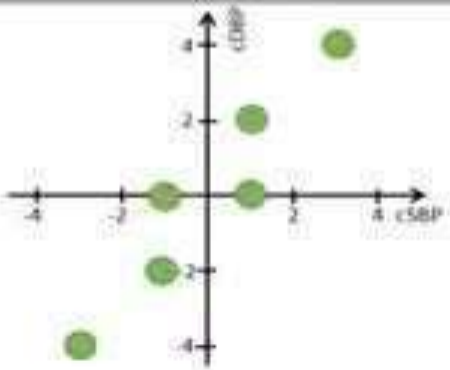
Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4



When we center the data, it means that we center the data points around the origin. Centering the data around the origin will help us later when we will rotate the data.

1. Center the data

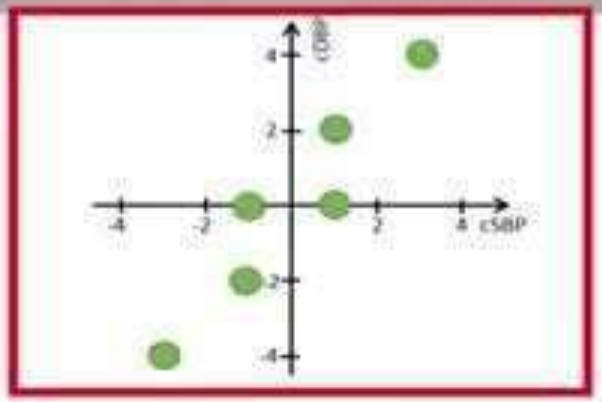
Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4



After we have centered the data, we will have the following values,

1. Center the data

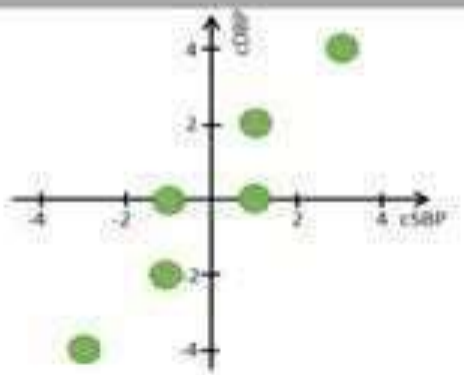
Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4



which can be plotted like this, where the x-axis now represents the centered systolic blood pressure, whereas the y-axis represents the centered diastolic blood pressure.

2. Calculate the covariance matrix

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4

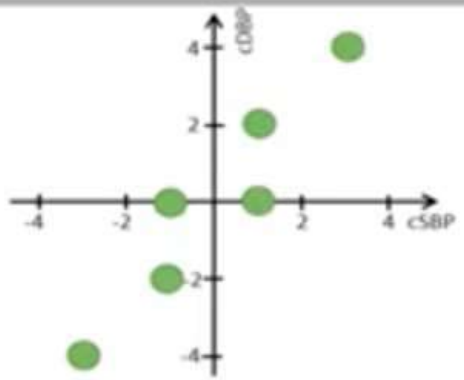


	SBP	DBP
SBP	4.4	5.6
DBP	5.6	8.0

Next, we calculate the covariance matrix based on the centered data. Note that we would have got the same values in the covariance matrix if we instead would have used the original data since the variance does not change when we center the data.

2. Calculate the covariance matrix

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4

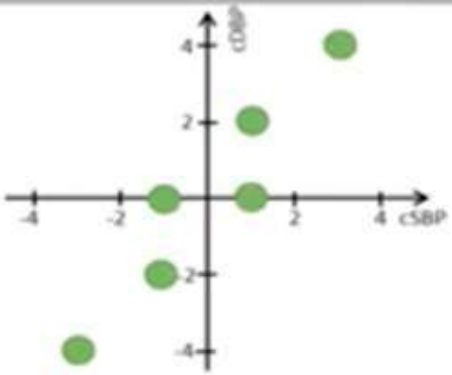


	SBP	DBP
SBP	4.4	5.6
DBP	5.6	8.0

Remember that the main diagonal of the covariance matrix includes the variance of each variable.

2. Calculate the covariance matrix

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4



	SBP	DBP
SBP	4.4	5.6
DBP	5.6	8.0

$$\mathrm{var}(\mathrm{cSBP}) = \frac{1}{n-1}\sum_{i=1}^{n}(\mathrm{cSBP}_i - \overline{\mathrm{cSBP}})^2 = ((-3)^2 + (-1)^2 + (-1)^2 + 1^2 + 1^2 + 3^2)/(6-1) = 22/5 = 4.4$$

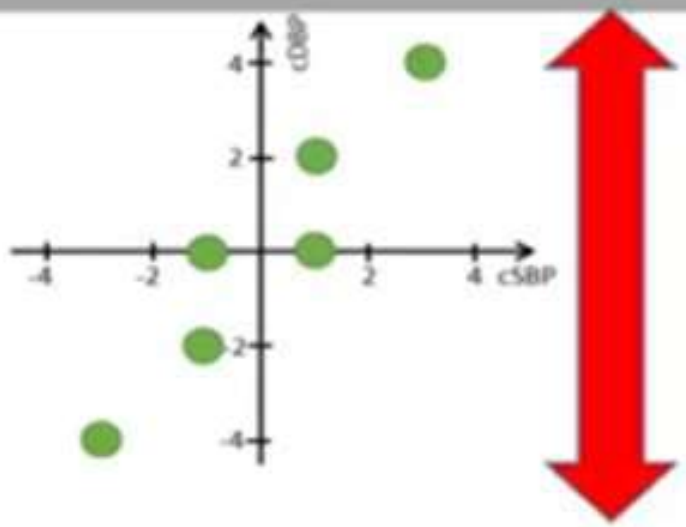
$$\mathrm{var}(\mathrm{cDBP}) = \frac{1}{n-1}\sum_{i=1}^{n}(\mathrm{cDBP}_i - \overline{\mathrm{cDBP}})^2 = ((-4)^2 + (-2)^2 + 0^2 + 0^2 + 2^2 + 4^2)/(6-1) = 40/5 = 8$$

$$\mathrm{cov}(\mathrm{cSBP},\mathrm{cDBP}) = \frac{1}{n-1}\sum(\mathrm{cSBP}_i - \overline{\mathrm{cSBP}})\cdot(\mathrm{cDBP}_i - \overline{\mathrm{cDBP}}) = ((-3)\cdot(-4) + (-1)\cdot(-2) + (-1)\cdot 0 + 1\cdot 0 + 1\cdot 2 + 3\cdot 4)/(6-1) = 28/5 = 5.6$$

Finally, we calculate the covariance, which is a measure of how much the two variables spread together.

2. Calculate the covariance matrix

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4

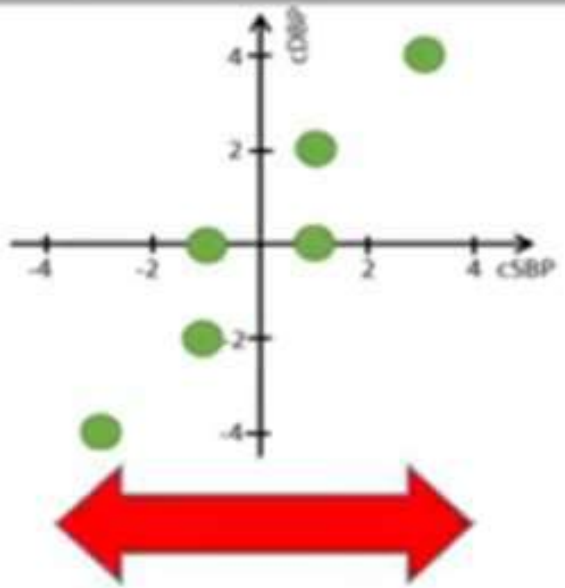


	SBP	DBP
SBP	4.4	5.6
DBP	5.6	8.0

We see that the spread in the diastolic blood pressure is a bit higher compared to

2. Calculate the covariance matrix

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4

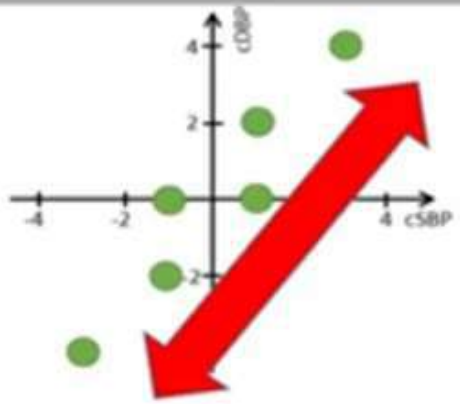


	SBP	DBP
SBP	4.4	5.6
DBP	5.6	8.0

the spread in the systolic blood pressure.

2. Calculate the covariance matrix

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4

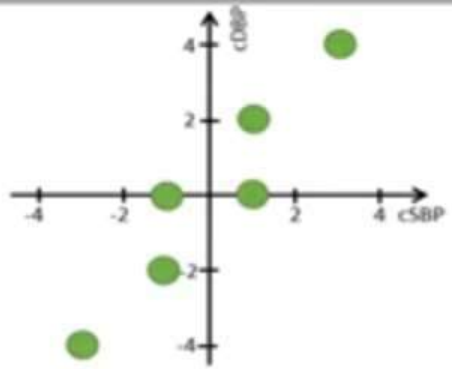


	SBP	DBP
SBP	4.4	5.6
DBP	5.6	8.0

The covariance is somewhere between these two values.

3. Calculate the eigenvalues of the covariance matrix

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4



	SBP	DBP
SBP	4.4	5.6
DBP	5.6	8.0

$$\det|A - \lambda I| = 0$$

$$\det\left|\begin{bmatrix} (4.4-\lambda) & 5.6 \\ 5.6 & (8.0-\lambda) \end{bmatrix}\right| = 0$$

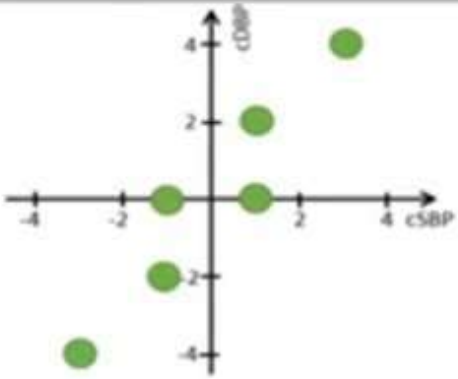
$$(4.4-\lambda)(8.0-\lambda) - 5.6 \cdot 5.6 = 0$$

$$\boxed{3.84 - 12.4\lambda + \lambda^2 = 0}$$

After some simplifications, we have the following quadratic equation. Quadratic equations like this can be solved in different ways, which will not be discussed here.

4. Calculate the eigenvectors of the covariance matrix

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4



$$A \cdot v = \lambda \cdot v$$

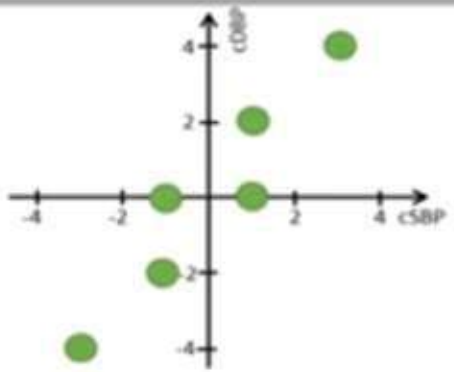
	SBP	DBP
SBP	4.4	5.6
DBP	5.6	8.0

$\lambda_1 = 0.32$ $\boxed{\lambda_2 = 12.08}$

Next, we calculate the corresponding eigenvectors to these two eigenvalues. We will start by calculating the eigenvector of the covariance matrix with the corresponding eigenvalue 12.08.

4. Calculate the eigenvectors of the covariance matrix

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4



	SBP	DBP
SBP	4.4	5.6
DBP	5.6	8.0

$$A \cdot v = \lambda \cdot v$$

$$\begin{bmatrix} 4.4 & 5.6 \\ 5.6 & 8.0 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} = 12.08 \cdot \begin{bmatrix} x \\ y \end{bmatrix}$$

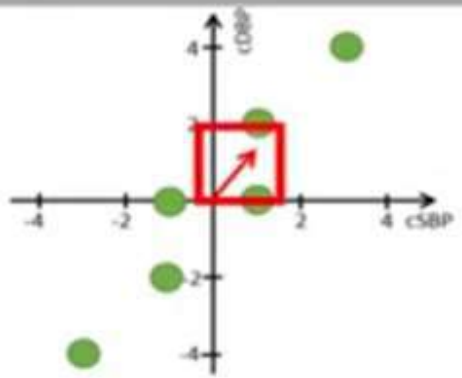
$$4.4x + 5.6y = 12.08x$$
$$5.6x + 8.0y = 12.08y$$

$$5.6y = 7.68x$$
$$5.6x = 4.08y$$
$$\boxed{\begin{aligned} y &= 1.37x \\ 1.37x &= y \end{aligned}}$$

Solving for y in the two equations, results in that y is equal to 1.37 x.

4. Calculate the eigenvectors of the covariance matrix

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4



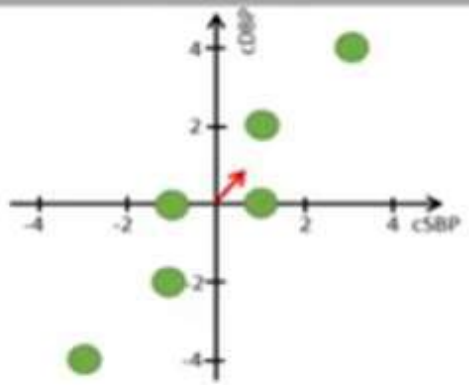
$$v_2 = \begin{bmatrix} 1 \\ 1.37 \end{bmatrix}$$

	SBP	DBP
SBP	4.4	5.6
DBP	5.6	8.0

We will now normalize this vector to unit length, which means that it should have a length of one. Watch the lecture about the eigenvectors and eigenvalues to see how one can normalize the eigenvector to unit length.

4. Calculate the eigenvectors of the covariance matrix

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4



$$v_2 = \begin{bmatrix} 0.59 \\ 0.81 \end{bmatrix} \quad \lambda_2 = 12.08$$

$$\lambda_1 = 0.32$$

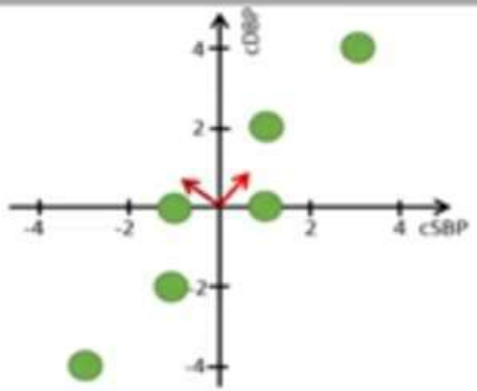
	SBP	DBP
SBP	4.4	5.6
DBP	5.6	8.0

$$\begin{bmatrix} 4.4 & 5.6 \\ 5.6 & 8.0 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} = \boxed{0.32} \begin{bmatrix} x \\ y \end{bmatrix}$$

To find the second eigenvector, we do the same calculations as before based on the second eigenvalue.

4. Calculate the eigenvectors of the covariance matrix

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4



$$v_2 = \begin{bmatrix} 0.59 \\ 0.81 \end{bmatrix} \quad \lambda_2 = 12.08$$

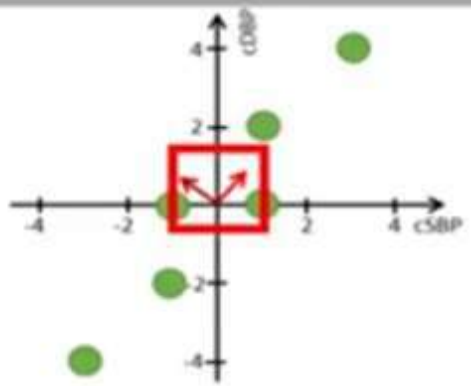
$$\boxed{v_1 = \begin{bmatrix} -0.81 \\ 0.59 \end{bmatrix}} \quad \lambda_1 = 0.32$$

	SBP	DBP
SBP	4.4	5.6
DBP	5.6	8.0

After some calculations, this vector represents our second eigenvector with unit length.

4. Calculate the eigenvectors of the covariance matrix

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4



$$v_2 = \begin{bmatrix} 0.59 \\ 0.81 \end{bmatrix} \quad \lambda_2 = 12.08$$

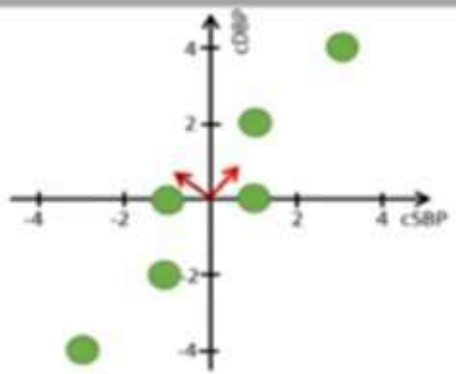
$$v_1 = \begin{bmatrix} -0.81 \\ 0.59 \end{bmatrix} \quad \lambda_1 = 0.32$$

	SBP	DBP
SBP	4.4	5.6
DBP	5.6	8.0

Since the covariance matrix is a symmetric matrix, the eigenvectors will be orthogonal, which means that the angle between them is 90 degrees.

5. Order the eigenvectors

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4



$$\boxed{v_1 = \begin{bmatrix} 0.59 \\ 0.81 \end{bmatrix} \quad \lambda_1 = 12.08}$$

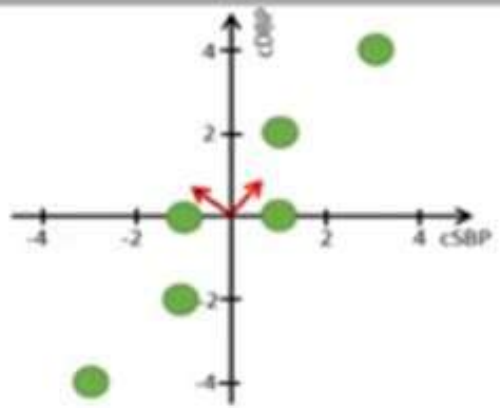
$$v_2 = \begin{bmatrix} -0.81 \\ 0.59 \end{bmatrix} \quad \lambda_2 = 0.32$$

	SBP	DBP
SBP	4.4	5.6
DBP	5.6	8.0

Since this eigenvector has the largest eigenvalue, it will represent our first eigenvector. We therefore rename this vector so that it is called v1 instead of v2.

5. Order the eigenvectors

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4



$$v_1 = \begin{bmatrix} 0.59 \\ 0.81 \end{bmatrix} \quad \lambda_1 = 12.08$$

$$v_2 = \begin{bmatrix} -0.81 \\ 0.59 \end{bmatrix} \quad \lambda_2 = 0.32$$

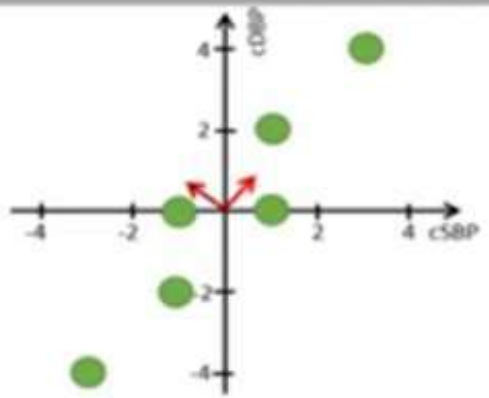
$$\boxed{V = \begin{bmatrix} 0.59 & -0.81 \\ 0.81 & 0.59 \end{bmatrix}}$$

	SBP	DBP
SBP	4.4	5.6
DBP	5.6	8.0

Let's put these two eigenvectors together into a matrix that we call V,

6. Calculate the principal components

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4

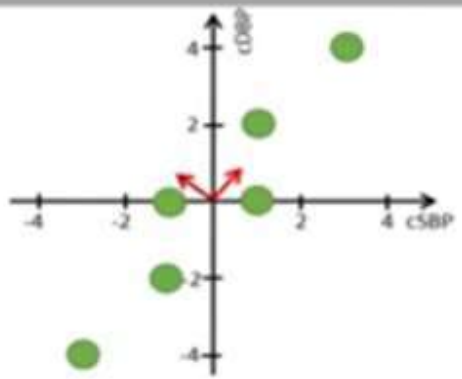


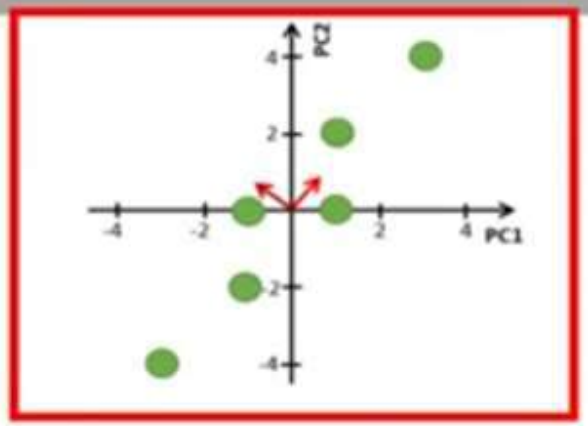
$$V = \begin{bmatrix} 0.59 & -0.81 \\ 0.81 & 0.59 \end{bmatrix}$$

We will now use this matrix to transform our original centered data so that the two variables are completely uncorrelated.

6. Calculate the principal components

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4



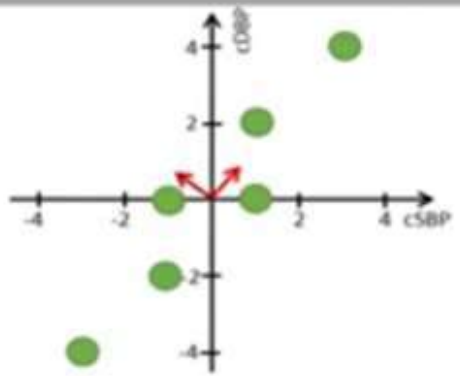


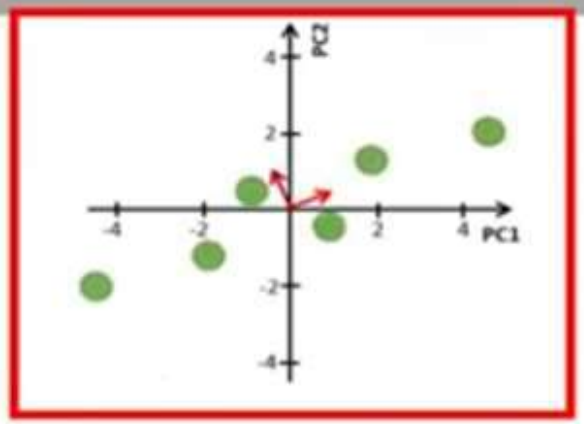
$$V = \begin{bmatrix} 0.59 & -0.81 \\ 0.81 & 0.59 \end{bmatrix} \qquad D = \begin{bmatrix} -3 & -4 \\ -1 & -2 \\ -1 & 0 \\ 1 & 0 \\ 1 & 2 \\ 3 & 4 \end{bmatrix} \qquad DV = \begin{bmatrix} -3 & -4 \\ -1 & -2 \\ -1 & 0 \\ 1 & 0 \\ 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} 0.59 & -0.81 \\ 0.81 & 0.59 \end{bmatrix} = \begin{bmatrix} -5.0 & 0.1 \\ -2.2 & -0.4 \\ -0.6 & 0.8 \\ 0.6 & -0.8 \\ 2.2 & 0.4 \\ 5.0 & -0.1 \end{bmatrix}$$

When we go from our original data matrix to the transformed data, this can be seen like we rotate the data clockwise until the two eigenvectors point in the same direction as the x and y-axes of the plot.

6. Calculate the principal components

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4



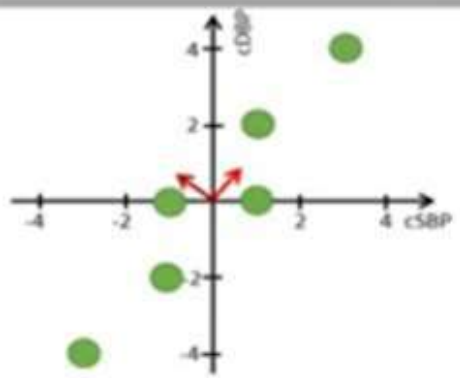


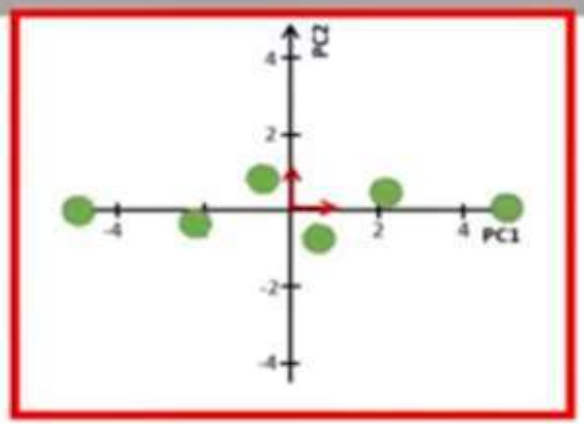
$$V = \begin{bmatrix} 0.59 & -0.81 \\ 0.81 & 0.59 \end{bmatrix} \qquad D = \begin{bmatrix} -3 & -4 \\ -1 & -2 \\ -1 & 0 \\ 1 & 0 \\ 1 & 2 \\ 3 & 4 \end{bmatrix} \qquad DV = \begin{bmatrix} -3 & -4 \\ -1 & -2 \\ -1 & 0 \\ 1 & 0 \\ 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} 0.59 & -0.81 \\ 0.81 & 0.59 \end{bmatrix} = \begin{bmatrix} -5.0 & 0.1 \\ -2.2 & -0.4 \\ -0.6 & 0.8 \\ 0.6 & -0.8 \\ 2.2 & 0.4 \\ 5.0 & -0.1 \end{bmatrix}$$

When we go from our original data matrix to the transformed data, this can be seen like we rotate the data clockwise until the two eigenvectors point in the same direction as the x and y-axes of the plot.

6. Calculate the principal components

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4



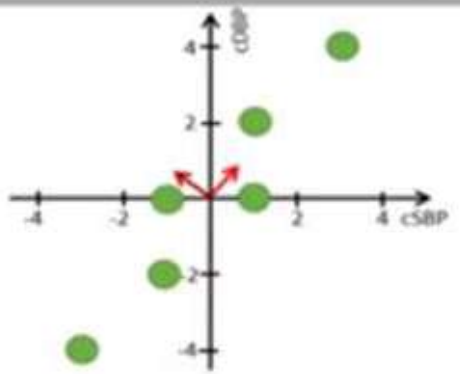


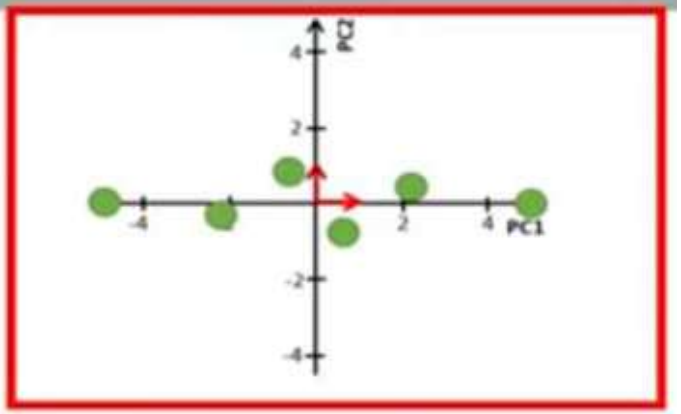
$$V = \begin{bmatrix} 0.59 & -0.81 \\ 0.81 & 0.59 \end{bmatrix} \qquad D = \begin{bmatrix} -3 & -4 \\ -1 & -2 \\ -1 & 0 \\ 1 & 0 \\ 1 & 2 \\ 3 & 4 \end{bmatrix} \qquad DV = \begin{bmatrix} -3 & -4 \\ -1 & -2 \\ -1 & 0 \\ 1 & 0 \\ 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} 0.59 & -0.81 \\ 0.81 & 0.59 \end{bmatrix} = \begin{bmatrix} -5.0 & 0.1 \\ -2.2 & -0.4 \\ -0.6 & 0.8 \\ 0.6 & -0.8 \\ 2.2 & 0.4 \\ 5.0 & -0.1 \end{bmatrix}$$

When we go from our original data matrix to the transformed data, this can be seen like we rotate the data clockwise until the two eigenvectors point in the same direction as the x and y-axes of the plot.

6. Calculate the principal components

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4



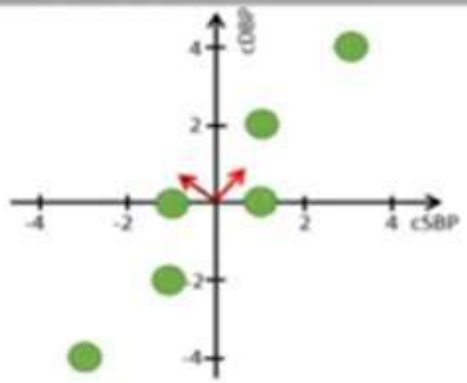


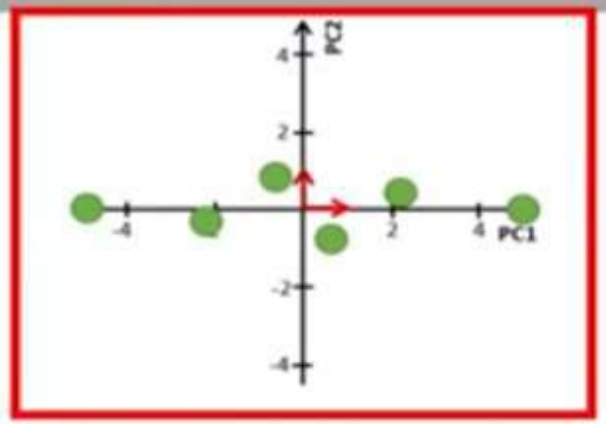
$$V = \begin{bmatrix} 0.59 & -0.81 \\ 0.81 & 0.59 \end{bmatrix} \qquad D = \begin{bmatrix} -3 & -4 \\ -1 & -2 \\ -1 & 0 \\ 1 & 0 \\ 1 & 2 \\ 3 & 4 \end{bmatrix} \qquad DV = \begin{bmatrix} -3 & -4 \\ -1 & -2 \\ -1 & 0 \\ 1 & 0 \\ 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} 0.59 & -0.81 \\ 0.81 & 0.59 \end{bmatrix} = \begin{bmatrix} -5.0 & 0.1 \\ -2.2 & -0.4 \\ -0.6 & 0.8 \\ 0.6 & -0.8 \\ 2.2 & 0.4 \\ 5.0 & -0.1 \end{bmatrix}$$

The rotated data now looks like this. Note that the labels of the axes have now been changed to principal component one and two.

6. Calculate the principal components

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4

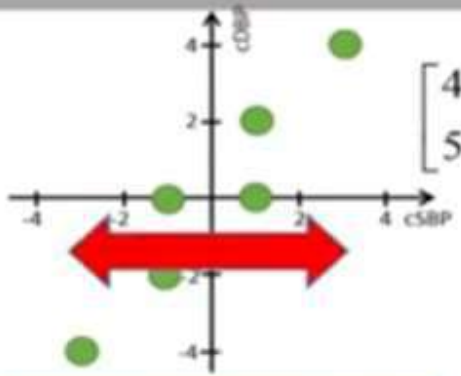




$$V = \begin{bmatrix} 0.59 & -0.81 \\ 0.81 & 0.59 \end{bmatrix} \qquad D = \begin{bmatrix} -3 & -4 \\ -1 & -2 \\ -1 & 0 \\ 1 & 0 \\ 1 & 2 \\ 3 & 4 \end{bmatrix} \qquad DV = \begin{bmatrix} -3 & -4 \\ -1 & -2 \\ -1 & 0 \\ 1 & 0 \\ 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} 0.59 & -0.81 \\ 0.81 & 0.59 \end{bmatrix} = \begin{array}{cc} \textbf{PC1} & \textbf{PC2} \\ \end{array} \begin{bmatrix} -5.0 & 0.1 \\ -2.2 & -0.4 \\ -0.6 & 0.8 \\ 0.6 & -0.8 \\ 2.2 & 0.4 \\ 5.0 & -0.1 \end{bmatrix}$$

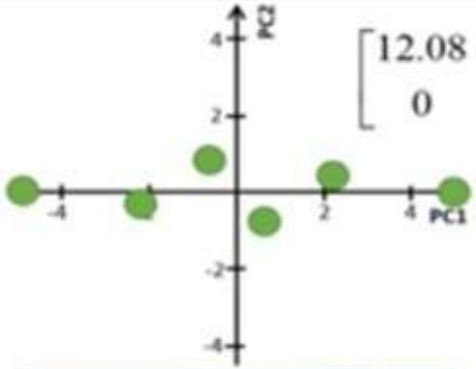
we would get the following plot, which represents the original plot after the rotation. Since we plot the principal component scores, this kind of plot is called a score plot.

Interpret the PCA



$$\begin{bmatrix} 4.4 & 5.6 \\ 5.6 & 8.0 \end{bmatrix}$$

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4
Var=4.4	Var=8.0

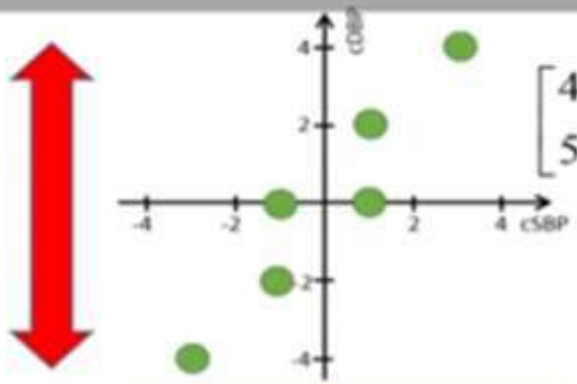


$$\begin{bmatrix} 12.08 & 0 \\ 0 & 0.32 \end{bmatrix}$$

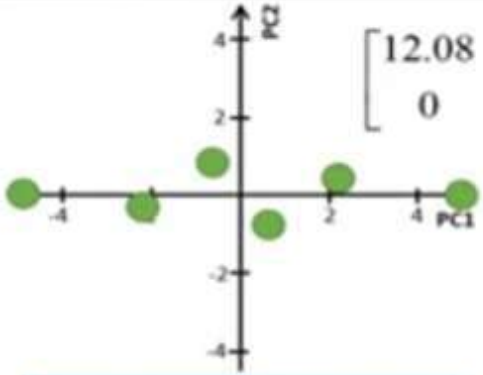
PC1	PC2
-5.0	0.1
-2.2	-0.4
-0.6	0.8
0.6	-0.8
2.2	0.4
5.0	-0.1
Var=12.08	Var=0.32

The variance of the systolic blood pressure is 4.4,

Interpret the PCA



$$\begin{bmatrix} 4.4 & 5.6 \\ 5.6 & 8.0 \end{bmatrix}$$



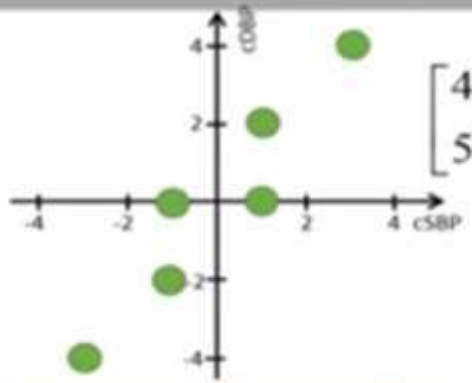
$$\begin{bmatrix} 12.08 & 0 \\ 0 & 0.32 \end{bmatrix}$$

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4
Var=4.4	Var=8.0

PC1	PC2
-5.0	0.1
-2.2	-0.4
-0.6	0.8
0.6	-0.8
2.2	0.4
5.0	-0.1
Var=12.08	Var=0.32

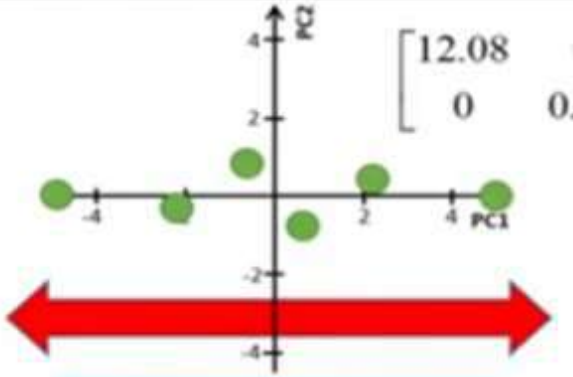
whereas the variance of the diastolic blood pressure is 8.

Interpret the PCA



$$\begin{bmatrix} 4.4 & 5.6 \\ 5.6 & 8.0 \end{bmatrix}$$

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4
Var=4.4	Var=8.0

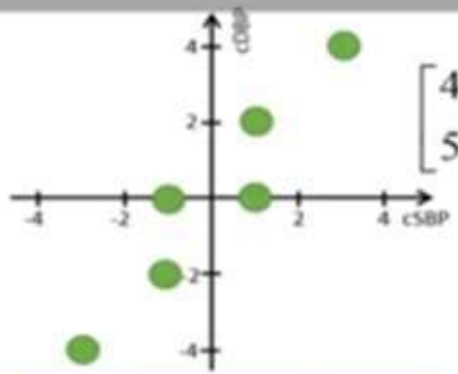


$$\begin{bmatrix} 12.08 & 0 \\ 0 & 0.32 \end{bmatrix}$$

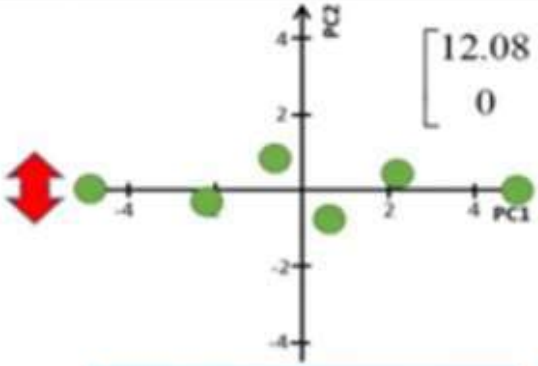
PC1	PC2
-5.0	0.1
-2.2	-0.4
-0.6	0.8
0.6	-0.8
2.2	0.4
5.0	-0.1
Var=12.08	Var=0.32

When we transform the data, using PCA, the first variable called PC1 has a variance of 12.08,

Interpret the PCA



$$\begin{bmatrix} 4.4 & 5.6 \\ 5.6 & 8.0 \end{bmatrix}$$



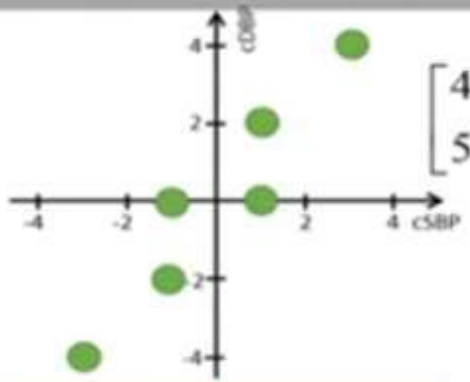
$$\begin{bmatrix} 12.08 & 0 \\ 0 & 0.32 \end{bmatrix}$$

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4
Var=4.4	Var=8.0

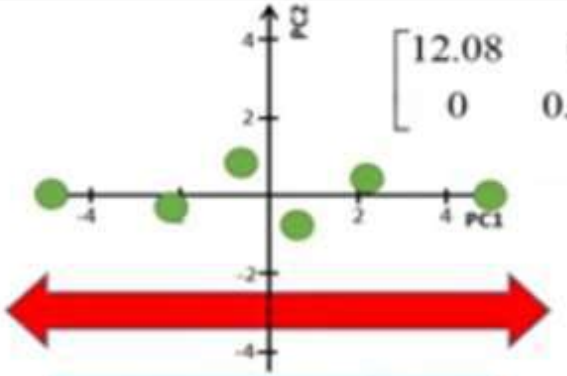
PC1	PC2
-5.0	0.1
-2.2	-0.4
-0.6	0.8
0.6	-0.8
2.2	0.4
5.0	-0.1
Var=12.08	Var=0.32

whereas PC2 has only a variance of 0.32.

Interpret the PCA



$$\begin{bmatrix} 4.4 & 5.6 \\ 5.6 & 8.0 \end{bmatrix}$$



$$\begin{bmatrix} 12.08 & 0 \\ 0 & 0.32 \end{bmatrix}$$

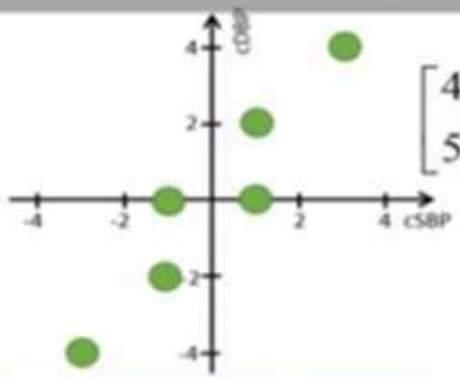
Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4
Var=4.4	Var=8.0

$$\%\,var = \frac{12.08}{\boxed{12.08+0.32}} = 97.4\%$$

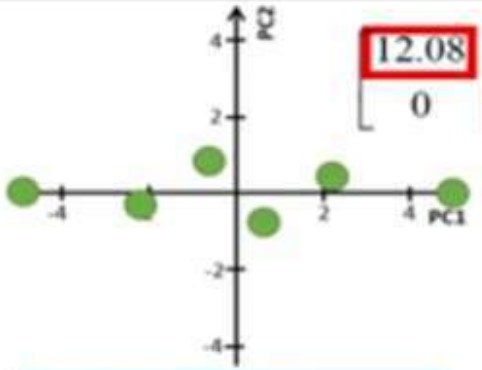
PC1	PC2
-5.0	0.1
-2.2	-0.4
-0.6	0.8
0.6	-0.8
2.2	0.4
5.0	-0.1
Var=12.08	Var=0.32

by the total variance,

Interpret the PCA



$$\begin{bmatrix} 4.4 & 5.6 \\ 5.6 & 8.0 \end{bmatrix}$$



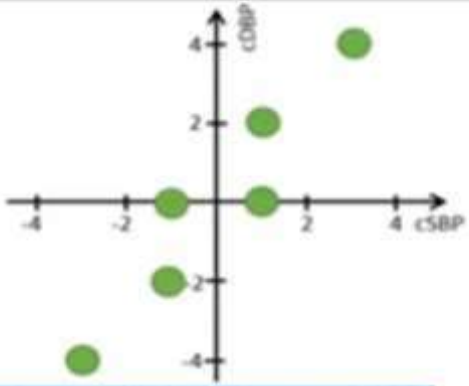
$$\begin{bmatrix} \boxed{12.08} & 0 \\ 0 & \boxed{0.32} \end{bmatrix}$$

Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4
Var=4.4	Var=8.0

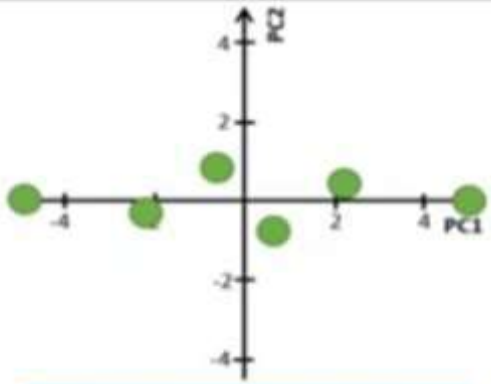
PC1	PC2
-5.0	0.1
-2.2	-0.4
-0.6	0.8
0.6	-0.8
2.2	0.4
5.0	-0.1
Var=12.08	Var=0.32

We also see that the variance of PC1 and PC2, correspond to the eigenvalues associated to the first and the second eigenvector.

Interpret the PCA



$$\begin{bmatrix} -3 & -4 \\ -1 & -2 \\ -1 & 0 \\ 1 & 0 \\ 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} 0.59 & -0.81 \\ 0.81 & 0.59 \end{bmatrix} = \begin{bmatrix} -5.0 & 0.1 \\ -2.2 & -0.4 \\ -0.6 & 0.8 \\ 0.6 & -0.8 \\ 2.2 & 0.4 \\ 5.0 & -0.1 \end{bmatrix}$$



$$PC1 = 0.59 \cdot cSBP + 0.81 \cdot cDBP$$

$$PC2 = -0.81 \cdot cSBP + 0.59 \cdot cDBP$$

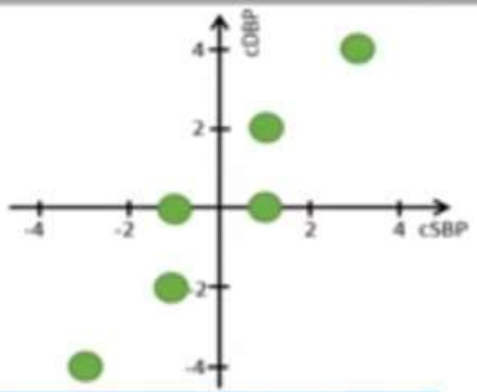
Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4
Var=4.4	Var=8.0

PC1	PC2
-5.0	0.1
-2.2	-0.4
-0.6	0.8
0.6	-0.8
2.2	0.4
5.0	-0.1
Var=12.08	Var=0.32

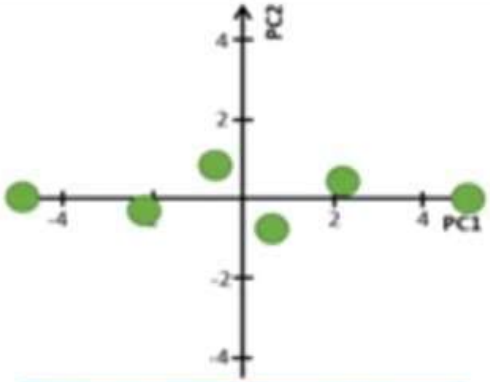
$$PC1_6 = 0.59 \cdot 3 + 0.81 \cdot 4 = 5$$

For example, if we would calculate the corresponding score for person number six,

Interpret the PCA



$$\begin{bmatrix} -3 & -4 \\ -1 & -2 \\ -1 & 0 \\ 1 & 0 \\ 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} 0.59 & -0.81 \\ 0.81 & 0.59 \end{bmatrix} = \begin{bmatrix} -5.0 & 0.1 \\ -2.2 & -0.4 \\ -0.6 & 0.8 \\ 0.6 & -0.8 \\ 2.2 & 0.4 \\ 5.0 & -0.1 \end{bmatrix}$$



Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4
Var=4.4	Var=8.0

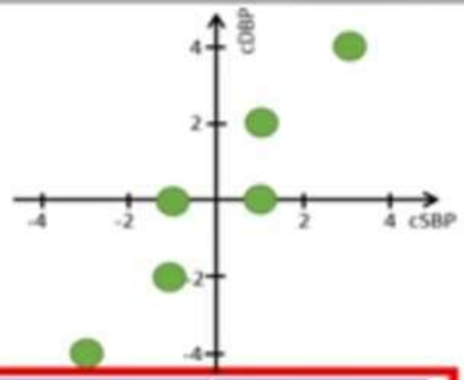
$$PC1 = 0.59 \cdot (SBP - \overline{SBP}) + 0.81 \cdot (DBP - \overline{DBP})$$

$$PC1 = 0.59 \cdot (\boxed{132} - 129) + 0.81 \cdot (\boxed{86} - 82) = 5.0$$

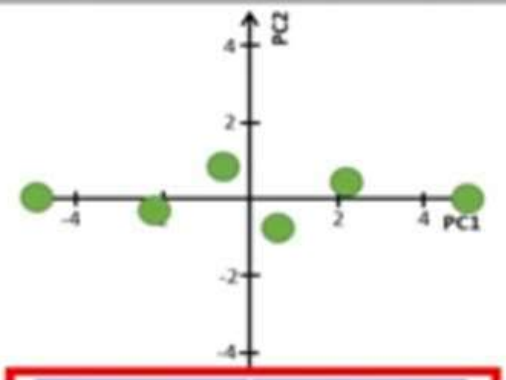
PC1	PC2
-5.0	0.1
-2.2	-0.4
-0.6	0.8
0.6	-0.8
2.2	0.4
5.0	-0.1
Var=12.08	Var=0.32

For example, if we would use the original blood pressure values for person number six,

Interpret the PCA



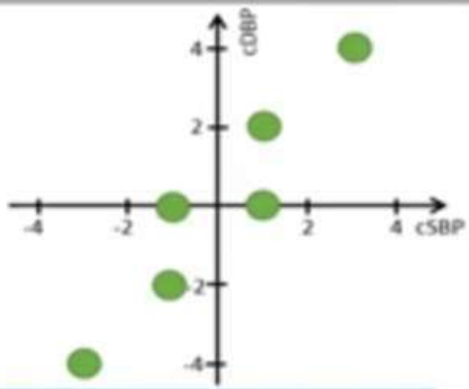
Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4
Var=4.4	Var=8.0



PC1	PC2
-5.0	0.1
-2.2	-0.4
-0.6	0.8
0.6	-0.8
2.2	0.4
5.0	-0.1
Var=12.08	Var=0.32

However, so far, we have not reduced the number of variables since we have the same number of principal components as the number of variables we started with.

Interpret the PCA

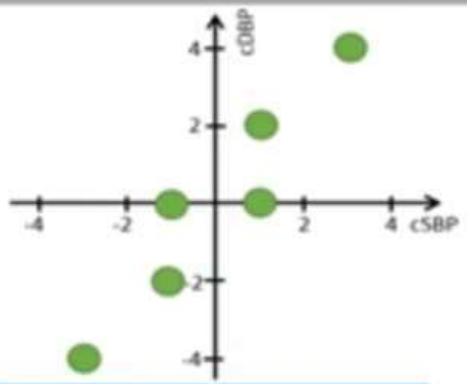


Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4
Var=4.4	Var=8.0

PC1	PC2
-5.0	0.1
-2.2	-0.4
-0.6	[illegible]
0.6	[illegible]
2.2	0.4
5.0	-0.1
Var=12.08	Var=0.32

Since the first principal component captures almost all variance, which can be interpreted as it stores almost all information about the two variables, we can simply delete the second principal component because it includes almost no information.

Interpret the PCA



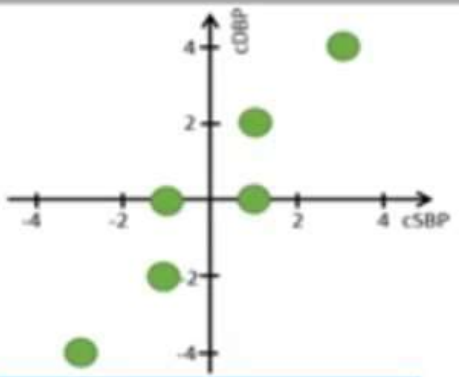
Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4
Var=4.4	Var=8.0

$$PC1 = 0.59 \cdot cSBP + 0.81 \cdot cDBP$$

PC1
-5.0
-2.2
-0.6
0.6
2.2
5.0
Var=12.08

As we have seen previously, by using the following equation, we can combine the two variables into just one variable, in a way that maximize the variance of the linear combination.

Interpret the PCA



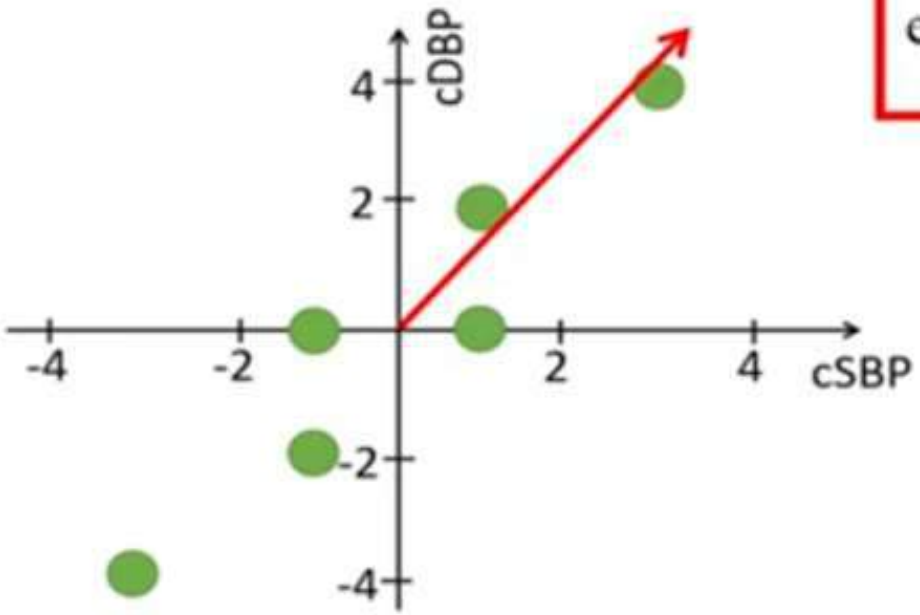
Centered SBP	Centered DBP
-3	-4
-1	-2
-1	0
1	0
1	2
3	4
Var=4.4	Var=8.0

$$PC1 = 0.59 \cdot cSBP + \boxed{0.81}\, cDBP$$

PC1
-5.0
-2.2
-0.6
0.6
2.2
5.0
Var=12.08

Since the absolute value of the weight for the diastolic blood pressure is higher than that of the systolic blood pressure, PCA put more weight on the diastolic blood pressure when the two variables are combined.

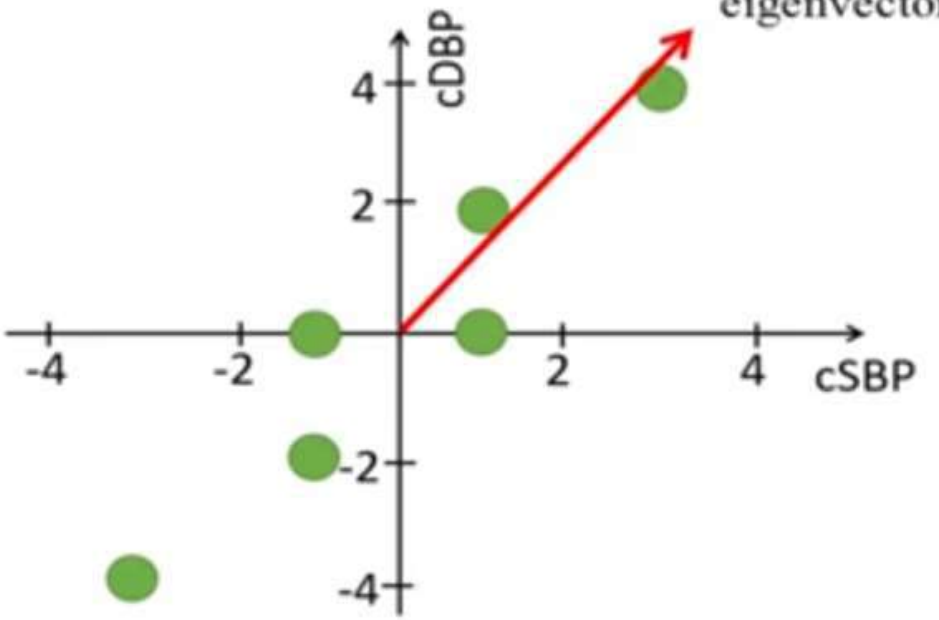
Interpret the eigenvector



$$\text{eigenvector} = \begin{bmatrix} 0.59 \\ 0.81 \end{bmatrix} \cdot 6 = \begin{bmatrix} 3.5 \\ 4.8 \end{bmatrix}$$

This is our previous eigenvector and if we extend it by multiplying by, for example, six,

Interpret the eigenvector

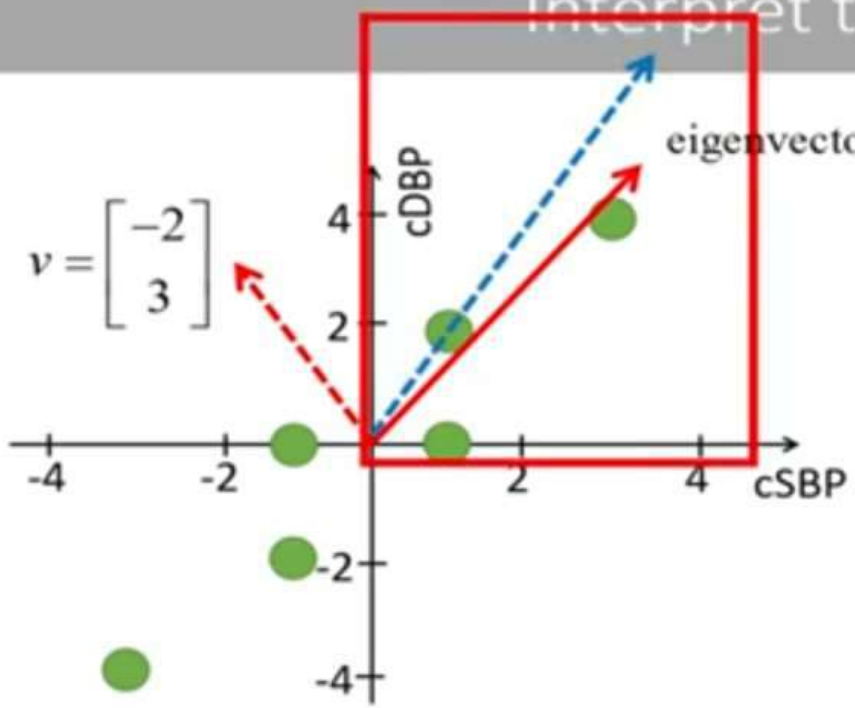


$$\text{eigenvector} = \begin{bmatrix} 3.5 \\ 4.8 \end{bmatrix}$$

$$\boxed{\text{cov}(cSBP, cDBP) = \begin{bmatrix} 4.4 & 5.6 \\ 5.6 & 8.0 \end{bmatrix}}$$

This is our covariance matrix.

Interpret the eigenvector



$$v = \begin{bmatrix} -2 \\ 3 \end{bmatrix}$$

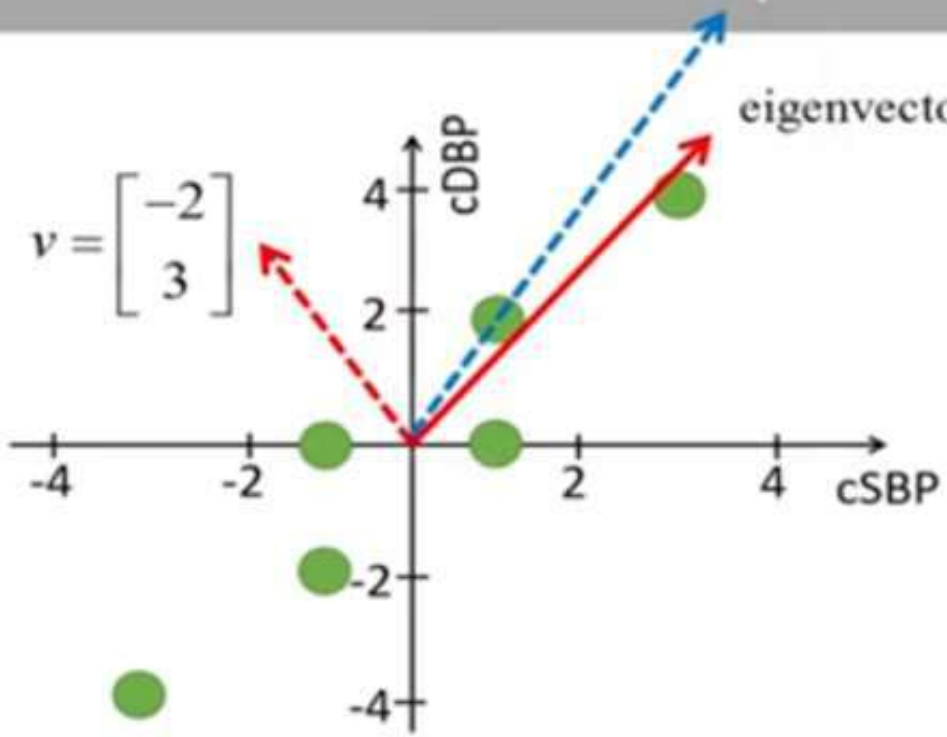
$$\text{eigenvector} = \begin{bmatrix} 3.5 \\ 4.8 \end{bmatrix}$$

$$\text{cov}(cSBP, cDBP) = \begin{bmatrix} 4.4 & 5.6 \\ 5.6 & 8.0 \end{bmatrix}$$

$$\begin{bmatrix} 4.4 & 5.6 \\ 5.6 & 8.0 \end{bmatrix} \cdot \begin{bmatrix} -2 \\ 3 \end{bmatrix} = \begin{bmatrix} 8 \\ 12 \end{bmatrix}$$

We see that the covariance matrix transformed the vector so that it moved in a direction closer to the eigenvector. Note that we here do not plot the full length of the vector since it cannot fit the screen. The importance is its direction.

Interpret the eigenvector



$$v = \begin{bmatrix} -2 \\ 3 \end{bmatrix}$$

$$\text{eigenvector} = \begin{bmatrix} 3.5 \\ 4.8 \end{bmatrix}$$

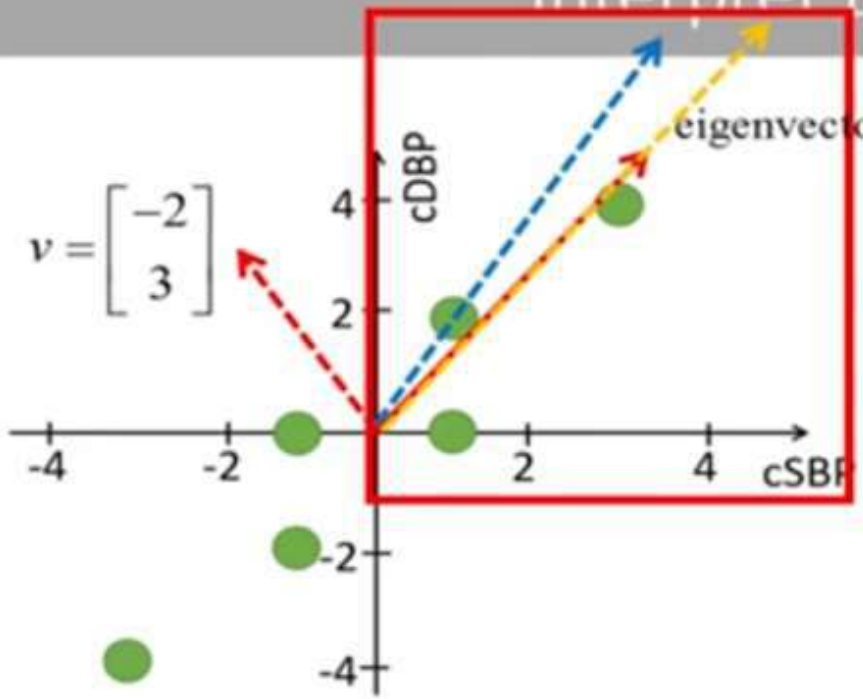
$$\text{cov}(cSBP, cDBP) = \begin{bmatrix} 4.4 & 5.6 \\ 5.6 & 8.0 \end{bmatrix}$$

$$\begin{bmatrix} 4.4 & 5.6 \\ 5.6 & 8.0 \end{bmatrix} \cdot \begin{bmatrix} -2 \\ 3 \end{bmatrix} = \begin{bmatrix} 8 \\ 12 \end{bmatrix}$$

$$\boxed{\begin{bmatrix} 4.4 & 5.6 \\ 5.6 & 8.0 \end{bmatrix} \cdot \begin{bmatrix} 8 \\ 12 \end{bmatrix}} = \begin{bmatrix} 107 \\ 147 \end{bmatrix}$$

If we multiply the covariance matrix by this new vector,

Interpret the eigenvector



$$v = \begin{bmatrix} -2 \\ 3 \end{bmatrix}$$

$$\text{eigenvector} = \begin{bmatrix} 3.5 \\ 4.8 \end{bmatrix}$$

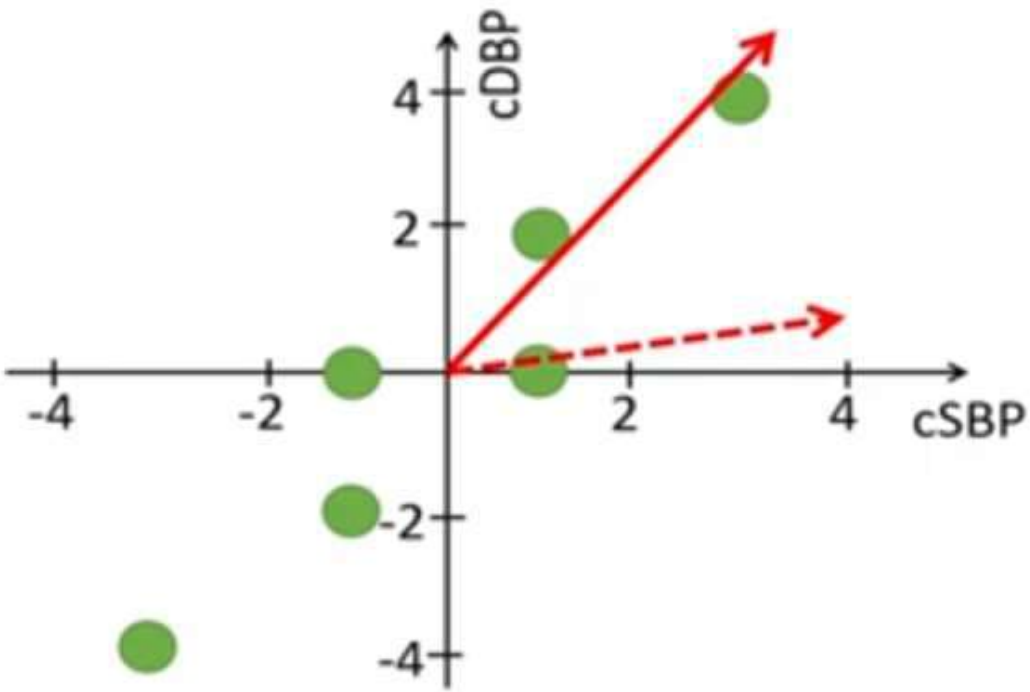
$$\text{cov}(cSBP, cDBP) = \begin{bmatrix} 4.4 & 5.6 \\ 5.6 & 8.0 \end{bmatrix}$$

$$\begin{bmatrix} 4.4 & 5.6 \\ 5.6 & 8.0 \end{bmatrix} \cdot \begin{bmatrix} -2 \\ 3 \end{bmatrix} = \begin{bmatrix} 8 \\ 12 \end{bmatrix}$$

$$\begin{bmatrix} 4.4 & 5.6 \\ 5.6 & 8.0 \end{bmatrix} \cdot \begin{bmatrix} 8 \\ 12 \end{bmatrix} = \begin{bmatrix} 107 \\ 147 \end{bmatrix}$$

This new vector will have more or less the same direction as the eigenvector.

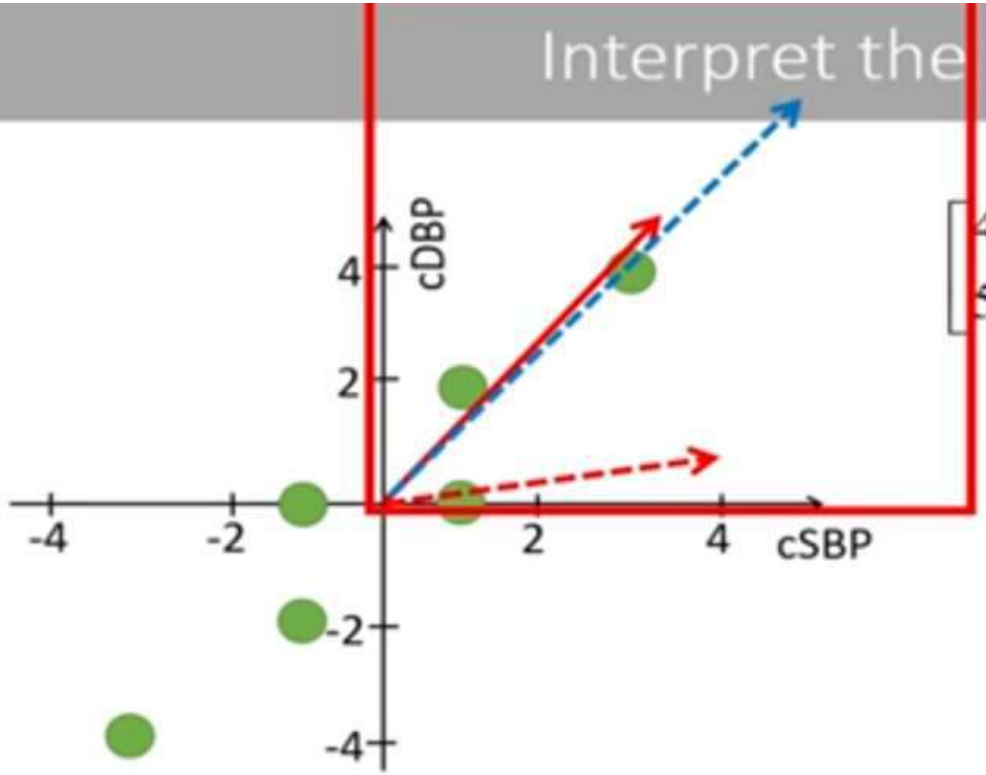
Interpret the eigenvector



$$\begin{bmatrix} 4.4 & 5.6 \\ 5.6 & 8.0 \end{bmatrix} \boxed{\begin{bmatrix} 4 \\ 1 \end{bmatrix}} = \begin{bmatrix} 23 \\ 30 \end{bmatrix}$$

Let's take another example vector, with the coordinate four and one.

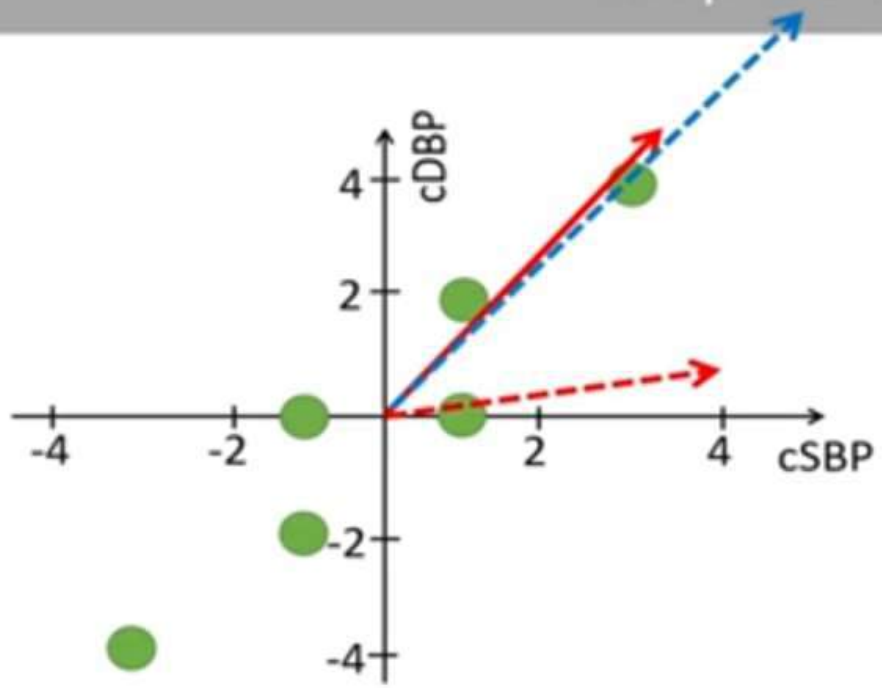
Interpret the eigenvector



$$\begin{bmatrix} 4.4 & 5.6 \\ 5.6 & 8.0 \end{bmatrix} \cdot \begin{bmatrix} 4 \\ 1 \end{bmatrix} = \begin{bmatrix} 23 \\ 30 \end{bmatrix}$$

We see that the covariance matrix will again rotate this vector so that it has a similar direction as the eigenvector.

Interpret the eigenvector



$$\begin{bmatrix} 4.4 & 5.6 \\ 5.6 & 8.0 \end{bmatrix} \begin{bmatrix} 4 \\ 1 \end{bmatrix} = \begin{bmatrix} 23 \\ 30 \end{bmatrix}$$

We can therefore conclude that the values in the covariance matrix rotate vectors towards the eigenvector, which points in a direction where the data has a maximal variance.

Uses of PCA

PCA is mostly used as a tool for **Compression** and **Simplifying** data for **easier learning** in exploratory data analysis and for making predictive models.

1- Better Perspective and less Complexity

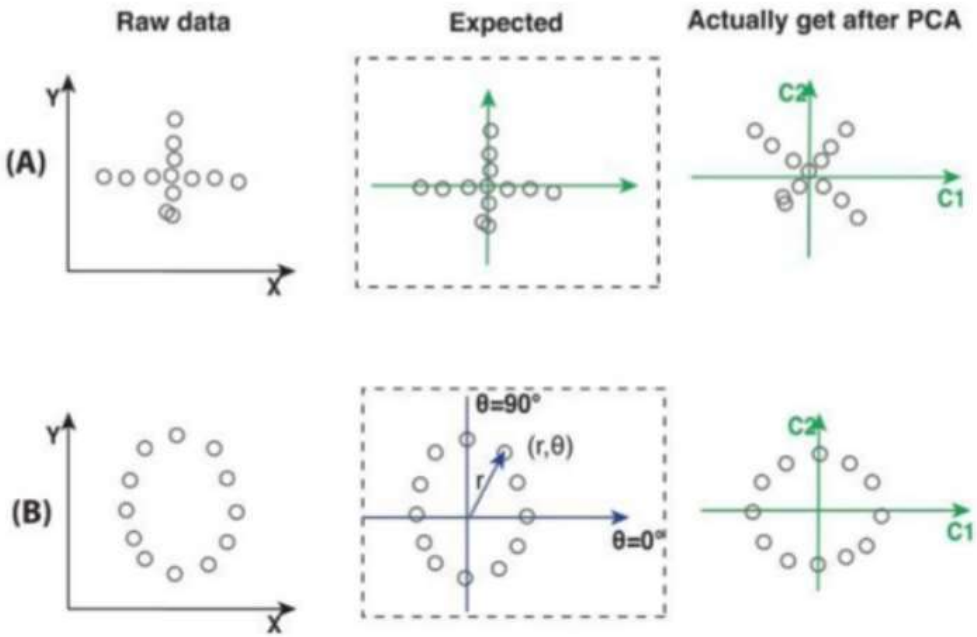
2 - Better visualization

3- Reduce size

4- Different perspective:

Limitation of PCA

If the data does not follow a multidimensional normal (Gaussian) distribution, PCA may not give the best principal components

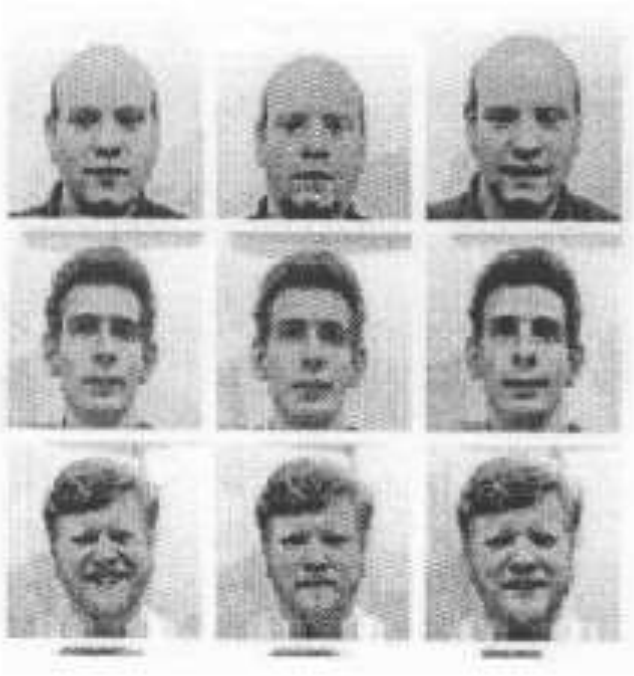


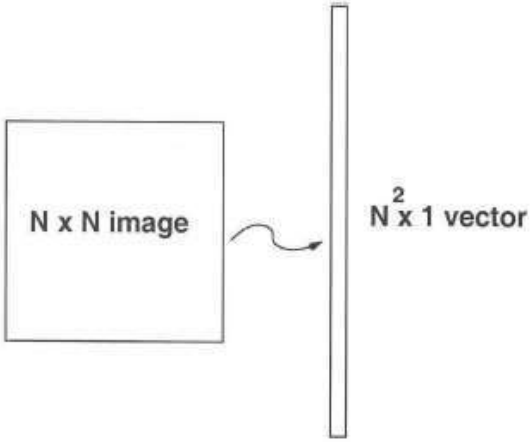
Application to Faces

- Computation of low-dimensional basis (i.e.,eigenfaces):

Step 1: obtain face images $I_1, I_2, ..., I_M$ (training faces)

(**very important:** the face images must be *centered* and of the same *size*)





Step 2: represent every image I_i as a vector Γ_i

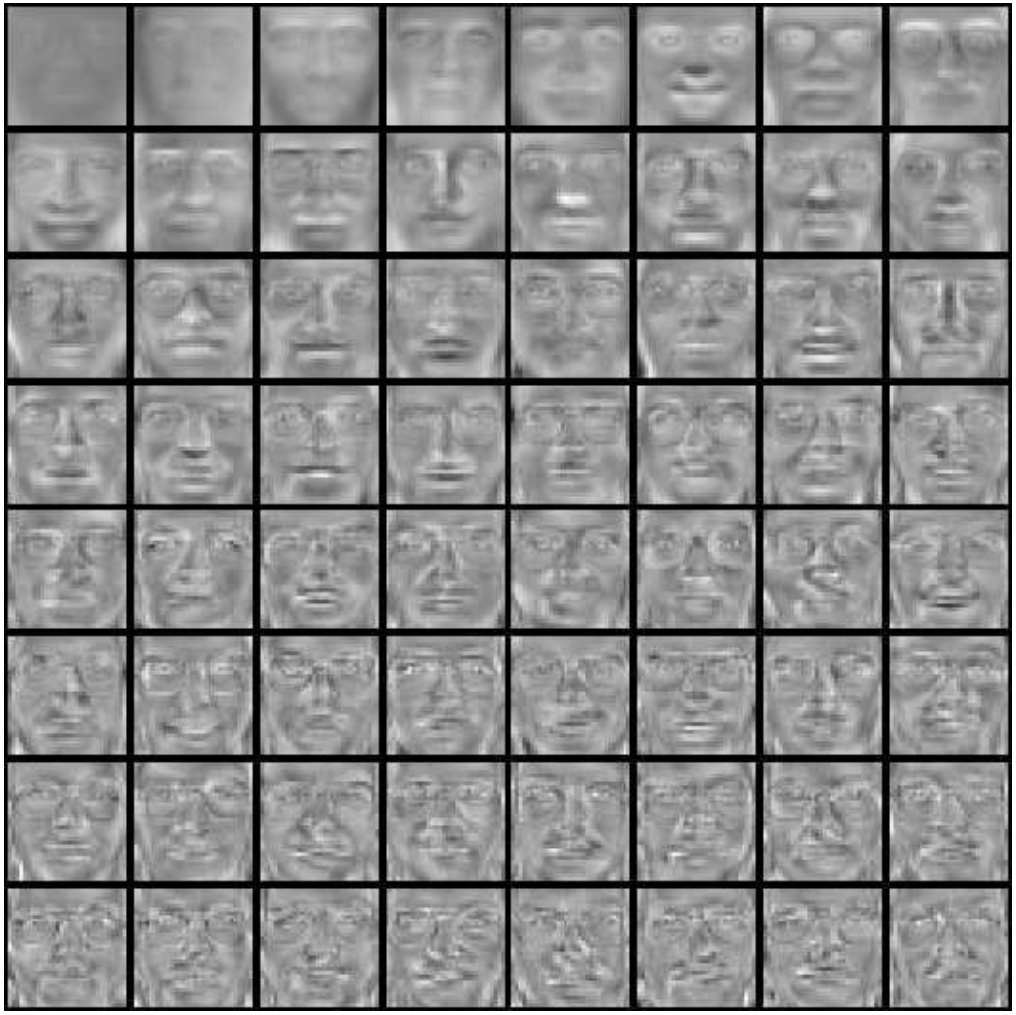
Example

Normalized face images

Example (cont'd)

Mean: μ

Top eigenvectors: $u_1,\ldots u_k$



Application to Faces (cont'd)

- Representing faces onto this basis

- Each face (minus the mean) Φ_i in the training set can be represented as a linear combination of the best K eigenvectors:

$$\hat{\Phi}_i - mean = \sum_{j=1}^{K} w_j u_j, \quad (w_j = u_j^T \Phi_i) \qquad (where \| u_j \| = 1)$$

(we call the u_j's *eigenfaces*)

Face reconstruction:

Thank you for your attention