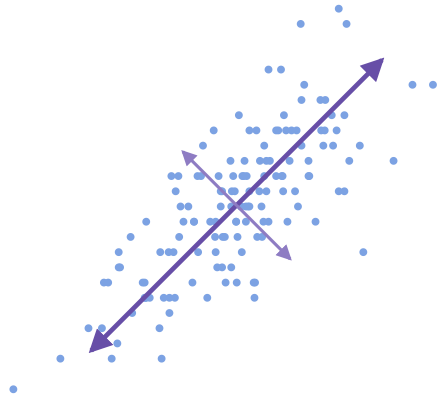


UNIT 02 · CHAPTER 1

Principal Component Analysis

CSE 30124 · Introduction to Artificial Intelligence
William Theisen · University of Notre Dame

Dimensionality Expansion

With the SVM _____ we saw a way of implicitly increasing the dimensionality of our data set.

This allowed us to fit **nonlinear problems**.

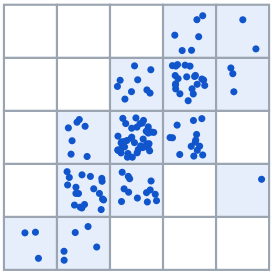
The Curse of Dimensionality

As we add more _____ to our dataset, our data becomes more _____.

As our number of dimensions increases, the _____ of the feature space grows _____.



total yards: 5 cells, all 5 used



total yards and first downs: 25 cells, 13 used

d = 80?

Dimensionality Reduction (exercise)

Dimensionality Reduction is an **unsupervised learning** technique that helps us **project our data from a high dimensional space to a lower dimensional space**.

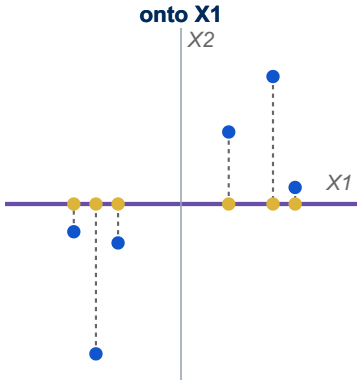
See if you can project the cubes into the lower dimensions.

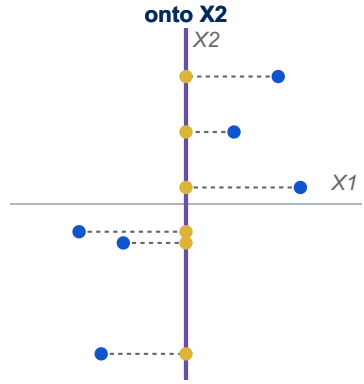
Are All Features Equally Useful?

- X_1 = Sq. Ftg.
- X_2 = # of doors
- **y** = Sale price

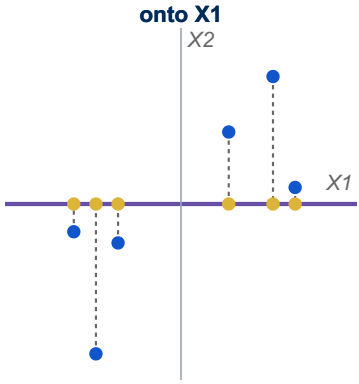
We may as well not even use X_2, there'd likely be no difference in our ability to solve the problem, but **we'd get to do less work**.

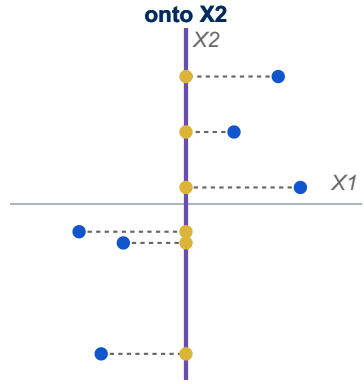
Projecting Data onto an Axis





Variance of Data on an Axis





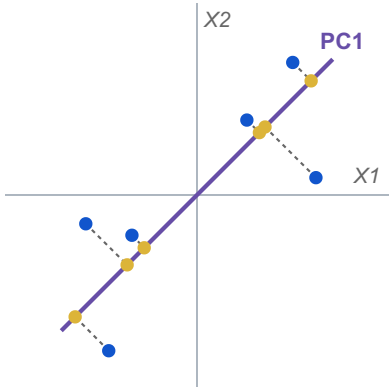
Why Do We Care About Variance

- X_1 = Sq. Ftg., X_2 = # of doors, **y** = Sale price

The more variance we observe in X_i, **the more precisely we can estimate its regression coefficient** (ω_i), all else equal.

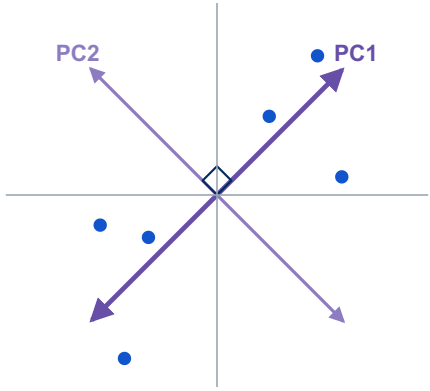
Projecting Data onto a New Axis

What if we could find a **new axis with higher variance**?



Principal Component Analysis (PCA)

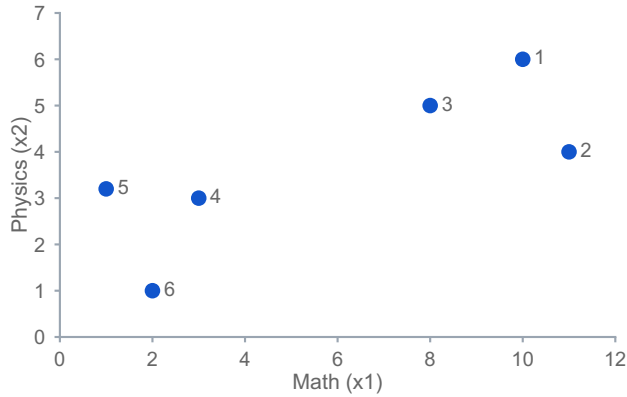
PCA works by discovering the directions (_____)
along which the data _____ .



Exploring PCA

Say we have the following dataset of students' scores in both Math (X_1) and Physics (X_2):

Student ID (S)	Math (x_1)	Physics (x_2)
901797635	10.0	6.0
2	11.0	4.0
3	8.0	5.0
4	3.0	3.0
5	1.0	3.2
6	2.0	1.0



Finding the Principal Components

1. Standardize our data
2. Calculate the _____ matrix
3. Compute the _____
4. Sort the eigenvalues and vectors from largest to smallest
5. Choose how many _____ to use
6. Project our points onto the new principal components

1.) Standardize the Data

We first calculate the **average** score for math and for physics.

We then shift the data so it is **centered and scaled around the origin** by subtracting the average and dividing by the **standard deviation**.

Q25. What does it mean to standardize your data? Why do you need to standardize your data before performing PCA?

2.) Calculate the Covariance Matrix

From the **covariance matrix** we can see the **relationships between our features**.

ND ID (S)	Math (x_1)	Physics (x_2)
901797635	1.05	1.45
2	1.30	0.19
3	0.54	0.82
4	−0.71	−0.44
5	−1.22	−0.32
6	−0.96	−1.70

[1.3] the students' covariance matrix

$$C = \begin{bmatrix} 1 & 0.7584 \\ 0.7584 & 1 \end{bmatrix}$$

Interpreting Covariance

- A **positive covariance** indicates that when one increases, the other increases (here we have Cov = `0.7584`)
- A **negative covariance** indicates that when one increases, the other decreases
- A **covariance close to 0** indicates that there's no real relationship between the two variables

Covariance is a measure of how our data relates to each other.

We could also think of covariance as a measure of how much **"informational" overlap** our features have.

Interpreting the Covariance Matrix

The total amount of variance in our data is:

$$\text{Total Var} = \text{Var}(X_1) + \text{Var}(X_2) = 1 + 1 = 2$$

The percentage of total variance per feature is:

$$\%X_1 = \frac{\text{Var}(X_1)}{\text{Total Var}} = \frac{1}{2} = 50\% \qquad \%X_2 = \frac{1}{2} = 50\%$$

So X_1 holds **50%** of **what separates our data** and X_2 has **50%** of what separates our data.

We can also calculate **shared linear variation** by:

$$(0.7584)^2 = 0.5752$$

So **57.52%** of the variation in Physics **can be explained using only Math**.

3.) Find the Eigenvalues

The **eigenvalues** represent the amount of _____ contained by each _____ we find.

The eigenvalue equation is:

[1.4] solve for λ

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

covariance matrix and identity matrix

$$C = \begin{bmatrix} 1 & 0.7584 \\ 0.7584 & 1 \end{bmatrix} \qquad I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Using the identity matrix means we're solving for **covariance being 0** (or an informational overlap of 0).

3.) Compute the Eigenvalues

3.) Compute the Eigenvectors

While the eigenvalues capture how much variance is explained by each principal component, the _____ show us the _____ along which the principal components point.

The eigenvector equation is:

[1.6] one equation for each eigenvalue

$$(C - \lambda I)\,V = 0$$

$$\lambda_1 = 1.7584 \quad \lambda_2 = 0.2416$$

Interpreting the Eigenvalues

You can think of the **eigenvalues** as the values in the **covariance matrix for our two new axes** (principal components).

before: math and physics

$$C = \begin{bmatrix} 1 & 0.7584 \\ 0.7584 & 1 \end{bmatrix}$$

after: PC1 and PC2

$$C = \begin{bmatrix} 1.7584 & 0.0 \\ 0.0 & 0.2416 \end{bmatrix}$$

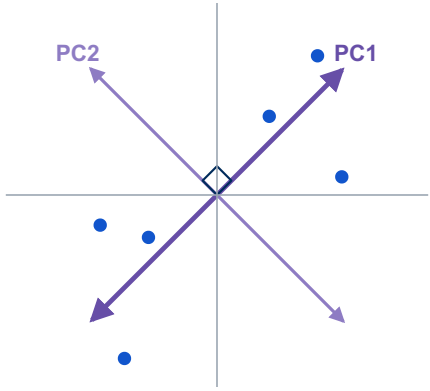
The total variance in our data is still `2` but now the first axis (PC1) contains **`87.92%` of the total variance**, compared to X_1 with **`50.00%`**.

3.) Compute the Eigenvectors

Notice that V_1 and V_2 are **orthogonal** (perpendicular) to each other.

Orthogonality of Principal Components

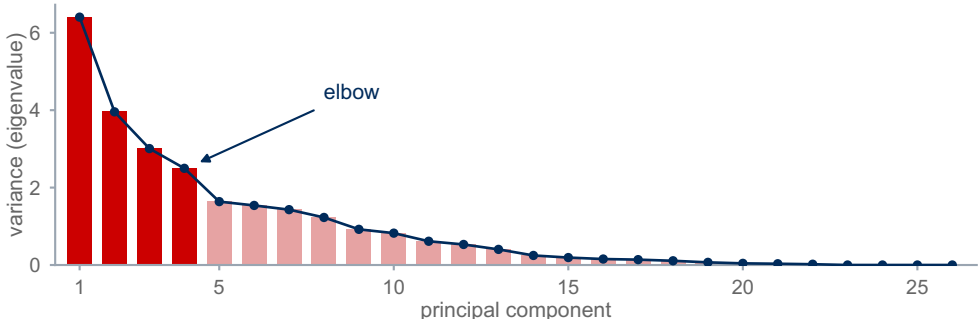
PCA can be seen as a **rotation of the original data axes** to a new set of axes. If the principal components weren't orthogonal, there would be overlap between the axes, meaning that some of the variance would be captured in multiple directions, resulting in **less efficient dimensionality reduction**.



Imagine non-orthogonal axes. Moving the X_1 of one of the points would inherently move the X_2 value of that same point! This would be **extremely high covariance**, and have large "informational" overlap between features.

4. and 5.) Sorting and Picking PCAs

Using **PCA** we find a number of principal components **equal to the number of original features** in our dataset, but we typically aren't interested in all of them. Instead we want a **minimum set that can explain the majority of the variance** in our data.



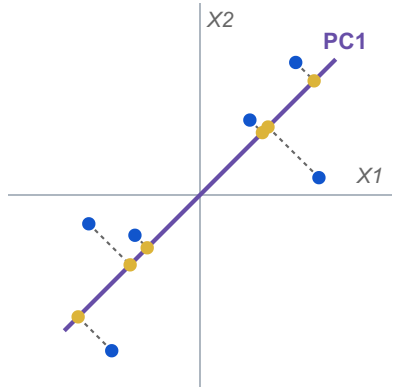
Why Use PCA

When using **PCA**, we can often get **equivalent accuracy with lower training time* and memory requirements**!

In many datasets, **not all features are equally useful**!

6. Project Our Data into the PCA Space

The final step is to use the eigenvalues and eigenvectors of our selected principal components to **transform our original data into our new "PC space"**.



Using PCA in Code

Like always it's easy.

```python
import numpy as np
from sklearn.decomposition import PCA
from sklearn.preprocessing import StandardScaler

# math and physics scores for our six students
X = np.array([[10, 6], [11, 4], [8, 5],
              [3, 3], [1, 3.2], [2, 1]])

Z = StandardScaler().fit_transform(X)  # step 1
pca = PCA(n_components=1).fit(Z)       # steps 2 to 5
print(pca.components_.round(4))
print(pca.explained_variance_ratio_.round(4))
print(pca.transform(Z).round(2).ravel())  # step 6
```

OUTPUT

```
[[0.7071 0.7071]]
[0.8792]
[ 1.77  1.05  0.96 -0.82 -1.08 -1.89]
```

Input Space vs. Feature Space

So far we've worked primarily in **input space**. Input space is _____.

However PCA finds us new dimensions. When we plot our original points along these new dimensions they could be considered to now be in **feature/latent space**.

Much of **Deep Learning** is about **transforming our input into latent spaces that we learn**.

Eigenfaces: Turk and Pentland (1991)

Dimensionality Reduction in the Eye

Each eye has well over 100 million light-sensing cells, but only about 1 million fibers in the optic nerve, so the retina **has to** _____ **what we see** before it goes to the brain.

Color is one example. Our three kinds of cone overlap a lot, so their signals are highly correlated, and the retina recombines them into **light–dark, red–green and blue–yellow** channels. Buchsbaum and Gottschalk (1983) showed these are close to the **principal components** of the cone signals.

Oja (1982) showed a single neuron with a Hebbian learning rule ends up finding the first principal component of its inputs.

Review

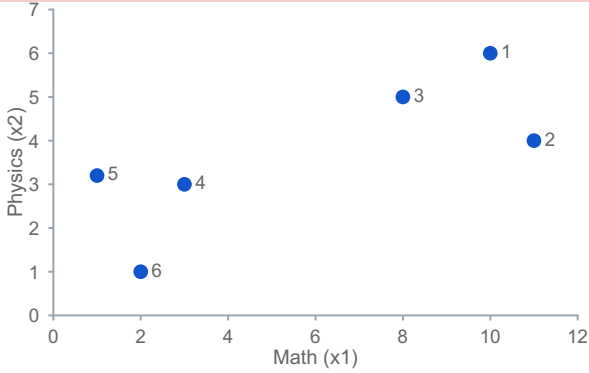
Complete the review: **Eigenvectors** identify _____, **eigenvalues** quantify _____. A **scree plot** helps us choose _____.

Matching: Principal Component Analysis

___ **Principal component**	**A.** Projecting data from a high-dimensional space to a lower-dimensional one
___ **Eigenvalue**	**B.** The space of our original measured features
___ **Eigenvector**	**C.** The direction along which a principal component points
___ **Covariance**	**D.** The amount of variance contained by each principal component
___ **Variance**	**E.** A new axis along which the data varies the most
___ **Scree plot**	**F.** As dimensions grow, the data needed to generalize grows exponentially
___ **Dimensionality reduction**	**G.** How two features vary together, their "informational" overlap
___ **Curse of dimensionality**	**H.** The new dimensions found by a transformation such as PCA
___ **Input space**	**I.** Variance explained per component, used to choose how many to keep
___ **Feature (latent) space**	**J.** How spread out the data is along an axis

Practice Problems

Q20. Draw (approximately) the two principal components on the plot and label them.



Practice Problems, Continued

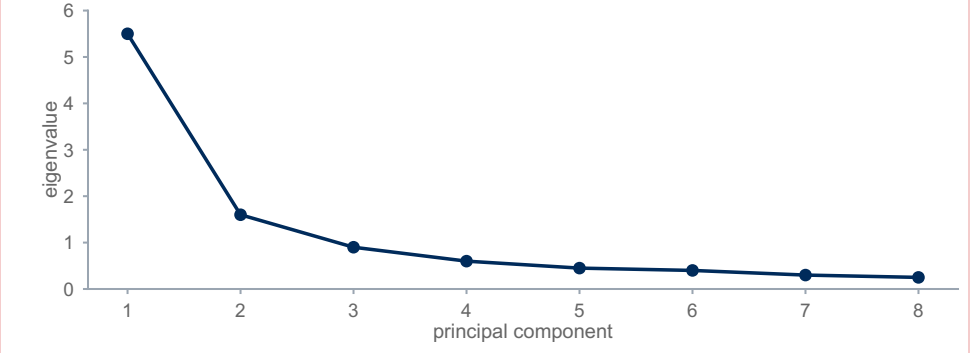
Q21. Why are the principal components always orthogonal to each other? What do we do with the covariance matrix to ensure this happens?

Q22. What is dimensionality reduction, and when may we use it?

Q23. What are the roles of eigenvalues and eigenvectors in PCA?

Q24. If I have a dataset with 5 input features and then transform the 5 features into 5 principal components, did I lose any information in my dataset? Why or why not? If I only select and use the top 2 principal components did I lose any information?

Q48. When we use PCA, we often get more principal components back than we'd realistically use. Given the scree plot below, indicate how many principal components you'd choose to use. Why don't we use every component?



Q56. In Homework03, why did we use SVD during PCA instead of the covariance matrix?