

PCA: Theory Q&A (Text Only)

Built from your handwritten class notes and your LDA notebook. This version has no formulas, no matrices and no numerical examples. It is for understanding the ideas and writing theory answers in words. The marks shown on each question tell you how long the answer should be.

Part A: Theory in plain words

Why PCA exists

PCA is used a lot in business intelligence and business analytics roles such as marketing, HR and fintech. It comes up in interviews as: what is PCA and why use it? The core problem it solves is **multicollinearity**. Regression assumes the predictor variables are not correlated with each other. When they are, the usual statistical fix is to drop one of the correlated variables, but in industry that rarely works because every variable carries useful information. So instead you use PCA.

PCA builds **new variables called principal components**. Each one is a linear combination (a weighted mix) of the original variables, and the components are **uncorrelated** with each other. The first component is one mix of the variables, the second is a different mix of the same variables, and so on. The regression is then run on the components instead of the original variables.

Hedge fund example

Take three highly correlated stocks, for example three banks or three IT companies. When news hits, they can move in three different ways:

Movement	What happens	Matches
Parallel shift	All the stocks move in the same direction	First component
Curvature change	Two move one way and one moves very little	Second component
Slope change	Two move in one direction and the third moves the opposite way	Third component

Each movement is a different weighted combination of the stocks. If the first component is dominated by one particular stock, that stock has the most influence. So instead of tracking every stock and guessing which pattern is dominating, you simply look at which component is active.

Geometric intuition

Imagine a company surveying customers on two things, comfort and look. Plotting the ratings gives a tilted oval cloud of points. None of the original axes line up with the real direction of spread, so PCA **rotates the coordinate system**. The first principal component lies along the long direction of the cloud, where the points are most spread out. The second lies perpendicular to it. The first component alone preserves most of the story. It tells the company the dimension along which customer opinions genuinely differ, and that is the dimension worth investing research and development into.

When to use PCA

- Use it when the variables are **highly correlated**. Only then does the scatter plot stretch along one dominant direction.
- If the variables are not correlated, the plot is a **circular blob**. Rotating the axes gives nothing useful because no direction is better than another, so PCA has no benefit.
- **Hedging example:** hedging means protecting yourself from loss by betting on both sides. Gold and silver move together, so if you do not know the future movement you buy one and sell the other. Hedge funds need to know what ratio of the two to hold so the position is balanced. PCA is used to find that

ratio.

Covariance and total variance

Covariance shows how much two variables vary together. The covariance matrix collects the spread of each variable (on the diagonal) and how each pair varies together (off the diagonal). The total variance, also called total risk, is simply the combined spread of all the variables.

Eigenvectors and eigenvalues

A vector is a point with a direction and a size. Multiplying a matrix with a vector either **rotates** it, which changes its direction, or **stretches** it, which changes its length. If you multiply the covariance matrix with a random vector, and keep multiplying the result again and again, the vectors drift toward the same general direction: the direction of maximum variability, the long dominant axis. Eventually you reach a vector that is no longer rotated by the multiplication. It is only stretched. That vector is the **eigenvector**, which is a line passing through the origin. The amount by which it gets stretched is the **eigenvalue**.

In PCA, the eigenvectors of the covariance matrix are the directions of the principal components, and each eigenvalue tells you how much variance lies along its direction.

Unit vector

A unit vector is a vector whose length is exactly one. We need it because the eigen relationship holds for any scalar multiple of the vector (double it, triple it, halve it), so we standardise the vector to a fixed length. To get a unit vector you divide the vector by its own length, and the length comes from the Pythagoras theorem.

Variance is redistributed, not lost

The total variability in the system never changes. Rotating the axes only changes how that variability is split across the axes. Before the rotation the spread may be shared almost evenly between the original variables. After rotating to line up with the eigenvector directions, the first new axis captures most of the spread and the second captures only a little. That is what makes **dimensionality reduction** possible: throw away the weak axes and keep the strong ones. With many variables you get as many eigenvalues as variables, and often a handful of the largest ones already capture most of the variability.

Rotation using trigonometry

When the axes are rotated by some angle, each new coordinate is the **shadow (projection)** of the old coordinates on the new axis. The shadow of each old coordinate on a new axis comes from the cosine of the angle between them (adjacent side over hypotenuse). The two shadows are then added together to give the new coordinate. Doing this for both new axes gives the full rotation, which can be written compactly as a matrix. The columns of that matrix are unit vectors along the new axes. Multiplying that matrix with its own transpose gives the identity matrix, which proves the new axes are perpendicular to each other and of unit length, so they form a valid set of axes.

Length of a projection (the linear algebra view)

To find how far a point reaches along an axis, take the unit vector along that axis and multiply it with the point. The result is the **length of the projection**. To get the actual position of the shadow, multiply that length by the unit vector. If you use a vector that is not a unit vector, the answer is inflated by the length of that vector, so the number you get is not the true length. That is why the axis vector must be a unit vector. The trigonometry method and the linear algebra method always give the same answer.

PCA needs centred data

The rotation happens about the origin, so each variable must have its mean subtracted first. This places the centre of the cloud at the origin, and the new axes then pass through the middle of the data.

PCA on images (eigenfaces)

- **Flatten** each image into one long column of pixel values. Place the columns of all the training images side by side to form the data matrix. The number of pixels is the number of features and the number of images is the number of training examples.
- **Mean image:** take the average of every pixel position across all the images. This gives one average face. **Subtract** it from each image to centre the data.
- There are two ways to build the covariance matrix: image-by-image (small, size set by the number of images) or pixel-by-pixel (huge, size set by the number of pixels).
- **The trick:** when there are far fewer images than pixels, find the eigenvectors of the small matrix, then multiply the data matrix by them to get eigenvectors of the large one. The eigenvalues stay the same. This saves a huge amount of computation.
- The resulting eigenvectors, when displayed as images, look like ghostly faces and are called **eigenfaces**.
- **Weights:** to describe a face, project it on the eigenfaces. The resulting set of weights is that face's compact signature, and faces can be compared or recognised using these weights.

LDA in one paragraph

LDA is the supervised partner of PCA. PCA ignores class labels and looks for the direction of maximum spread. LDA uses the labels and looks for the direction that pushes the class averages as far apart as possible while keeping each class tightly bunched. It builds a between-class scatter and a within-class scatter, and the eigenvectors of the within-class inverse times the between-class matrix give the discriminant directions. The eigenvector with the largest eigenvalue is the best direction. For two classes there is only one useful direction.

Part B: Questions and Answers

Write the answer to about this length: two marks is a few sentences, three marks is the idea, the reasoning and one example.

Q1. What is PCA and why is it used?

[2 marks]

PCA, or Principal Component Analysis, creates new variables called principal components. They are weighted combinations of the original variables and they are uncorrelated with each other. It is used to handle multicollinearity and to reduce the number of variables while keeping most of the variability. It is widely used in business analytics roles such as marketing, HR and fintech.

Q2. Which problem does PCA solve? Explain multicollinearity.

[2 marks]

Regression assumes the predictors are not correlated with each other. When they are correlated, that is multicollinearity. The usual statistical fix is to drop one of the correlated variables, but in industry that does not work because every variable holds useful information. PCA replaces the correlated variables with uncorrelated principal components, and the regression is then built on those components.

Q3. Explain PCA using the hedge fund example.

[3 marks]

Take several highly correlated stocks. When news hits they can move in three ways. In a parallel shift all of them move in the same direction. In a curvature change two move one way and one moves very little. In a slope change two move in one direction and the third moves the opposite way.

These three movements correspond to the first, second and third principal components. Each is a different weighted combination of the stocks. If the first component is dominated by one stock, that stock has the most influence. So instead of tracking every stock and guessing which pattern dominates, you only check which component is active.

Q4. Explain the geometric intuition of PCA.

[2 marks]

Suppose customers rate comfort and look in a survey. The points form a tilted oval. None of the original axes line up with the real spread, so we rotate the coordinate system. The first component lies along the direction of maximum spread and the second is perpendicular to it. The first component alone keeps most of the story and shows the dimension along which customer opinion truly differs, which is worth investing research and development into.

Q5. When should PCA be used and when does it fail to help?

[2 marks]

PCA should be used when variables are highly correlated, because then the scatter plot stretches along one dominant direction. If the variables are not correlated the plot is a circular blob. Rotating the axes then gives nothing useful because no direction is better than any other, so PCA has no benefit.

Q6. How is PCA used in hedging?

[2 marks]

Hedging means protecting yourself from loss by betting on both sides. Gold and silver are correlated, so when one rises the other usually does too. Without knowing the future movement, you buy one and sell the other. Hedge funds need the right ratio of the two to hold so the position is balanced. PCA is used to find that ratio.

Q7. What is the covariance matrix and what is total variance?**[2 marks]**

Covariance shows how much two variables vary together. The covariance matrix puts each variable's own spread on the diagonal and the shared variation between each pair in the remaining places. Total variance, or total risk, is the combined spread of all the variables, which is the sum of the individual variances.

Q8. What are eigenvectors and eigenvalues? Explain the intuition.**[3 marks]**

Multiplying a matrix with a vector rotates it or stretches it. If you multiply the covariance matrix with a vector and keep repeating the multiplication on the new vectors, they drift toward the direction of maximum variability, the long dominant axis. You eventually reach a vector that is not rotated any more, only stretched. That vector is the eigenvector, a line through the origin, and the stretching factor is the eigenvalue.

In PCA, the eigenvectors of the covariance matrix give the directions of the principal components, and each eigenvalue is the variance along its direction.

Q9. What is a unit vector, why is it needed and how is it found?**[2 marks]**

A unit vector has a length of exactly one. It is needed because the eigen relationship holds for any scalar multiple of the vector, so we standardise the vector to a fixed length. It is found by dividing the vector by its own length, where the length comes from the Pythagoras theorem.

Q10. Why does rotation not change total variance, and how does this lead to dimensionality reduction?**[3 marks]**

Total variability is a property of the data, not of the axes. Rotation only redistributes how it is shared among the axes. Before rotation the spread may be split almost evenly between the original variables. After aligning with the eigenvector directions the first new axis carries most of it and the second carries little.

Since the weak axes carry little information, we can discard them and keep only the strong ones. That is dimensionality reduction. With many variables, a handful of the largest eigenvalues can already hold most of the variability.

Q11. Explain how the new coordinates are obtained after rotating the axes, using trigonometry.**[3 marks]**

Each new coordinate is the shadow, or projection, of the old coordinates on the new axis. The shadow of an old coordinate on a new axis is found from the cosine of the angle between them, which is adjacent side over hypotenuse. For one new axis the angle to the first old axis is the rotation angle itself, and the angle to the second old axis is the complement of it, so cosine turns into sine. For the other new axis the angle to the first old axis is the rotation angle plus a right angle, which gives a negative sine, and the second old axis gives a cosine.

Adding the two shadows for each new axis gives the new coordinates. The same result can be written compactly as a matrix applied to the old coordinates.

Q12. Why is the rotation matrix made of unit vectors and what does multiplying it with its transpose show?**[2 marks]**

Each column of the matrix is a unit vector pointing along one of the new axes. Multiplying the matrix by its transpose gives the identity matrix. This shows that each column has unit length and that the columns are perpendicular to each other, so the new axes form a valid set of axes and the rotation does not stretch or distort the data.

Q13. How do you find the length of the projection of a point on an axis?

[3 marks]

Take the unit vector along the axis and multiply it with the point. The result is the length of the projection, which is how far the point's shadow lies along that axis. To get the position of the shadow itself, multiply that length by the unit vector. The same answer is obtained by the trigonometry method using the cosine and sine of the angle, so the two views agree.

Q14. Why must the axis vector be a unit vector?

[2 marks]

If you multiply with a vector that is not of unit length, the answer is inflated by the length of that vector. The number you get is then larger than the true length of the shadow. A unit vector has length one, so it adds no extra scaling and the result is the true length of the projection.

Q15. Why does PCA work only on centred data?

[2 marks]

The rotation happens about the origin. Subtracting the mean of each variable places the centre of the cloud at the origin, so the new axes pass through the middle of the data and the variance along each axis is measured correctly.

Q16. Explain how PCA is applied to face images to get eigenfaces.

[3 marks]

- Flatten every image into one long column of pixel values and place the columns side by side to form the data matrix.
- Compute the mean image by averaging each pixel position over all images, then subtract it from every image to centre the data.
- Build the covariance matrix and find its eigenvalues and eigenvectors. The eigenvectors, shown as images, are the eigenfaces.
- Project each face on the eigenfaces to get its weights. Keep only the top eigenfaces that hold enough of the variance.

Q17. Why is the small covariance matrix used for images instead of the large one?

How are the eigenvectors related?

[3 marks]

A picture has a very large number of pixels, while the number of training images is much smaller. The covariance built pixel by pixel is huge and would need a very expensive eigen computation. The covariance built image by image is small. We find the eigenvectors of the small matrix, then multiply the data matrix by them. The result is an eigenvector of the large matrix with the same eigenvalue. This is shown by multiplying the small matrix's eigen relationship by the data matrix on the left.

Q18. What are the weights in the eigenface method and what are they used for?

[2 marks]

The weights are the coordinates of a face when it is projected on the eigenfaces. They are obtained by multiplying the transpose of the eigenface matrix with the centred face. They act as a compact signature of the face, much shorter than the full image, and are used to compare, classify or recognise faces.

Q19. What is the mean image and why is it subtracted?**[2 marks]**

The mean image is the average of all training images, calculated pixel by pixel, which gives one average face. Subtracting it from every face centres the data, which PCA requires. The eigenfaces then capture how each face differs from the average face instead of what all faces have in common.

Q20. Compare PCA and LDA.**[3 marks]**

	PCA	LDA
Type	Unsupervised, ignores class labels	Supervised, uses class labels
Goal	Direction of maximum variance	Direction that best separates the classes
Matrix used	Covariance matrix	Within-class inverse times between-class matrix
What eigenvectors give	Principal components	Discriminant directions
Number of components	Up to the number of features	At most the number of classes minus one

Q21. Analyse how eigenvectors and eigenvalues are used in PCA and LDA.**[2 marks]**

In PCA the eigenvectors of the covariance matrix are the principal component directions, which are the new rotated axes. The eigenvalue of each one is the variance along it, so the largest eigenvalue gives the first component. The share of variance explained by a component is its eigenvalue divided by the total of all eigenvalues, and the weakest components are dropped for dimensionality reduction.

In LDA the eigenvectors of the within-class inverse times the between-class matrix are the discriminant directions. The eigenvalue measures how well the classes are separated along that direction, and the eigenvector with the largest eigenvalue is the optimal discriminant direction. With two classes only one eigenvalue is meaningfully non-zero, and its eigenvector is the line onto which the data is projected.

Q22. How do you decide how many principal components to keep?**[2 marks]**

Work out the variance explained by each component, which is its eigenvalue divided by the sum of all eigenvalues. Add these up from the strongest component downward and keep the smallest number of components whose combined share reaches the target, for example a high percentage such as the one a question specifies. In code this is done by giving PCA a fraction instead of a count, and it keeps just enough components to preserve that share of the variance.

Q23. What is the difference between PCA and simply dropping a correlated variable?**[2 marks]**

Dropping a variable throws away all the information it carries. PCA keeps the information from every variable by blending them into new uncorrelated components, and then lets you discard only the components that carry little variance. So the loss of information is small and controlled.

Quick revision sheet

Idea	One line
PCA	Rotates the axes to the direction of maximum variance. New variables are uncorrelated weighted mixes of the old ones.
When to use	When variables are highly correlated. A circular blob means PCA is useless.
Eigenvector	A direction that is only stretched, never rotated, by the covariance matrix.
Eigenvalue	How much it is stretched, which equals the variance along that direction.
Unit vector	Length of one. Needed because any multiple of the eigenvector also works.
Total variance	Not changed by rotation, only redistributed. This is why dropping weak axes is safe.
Projection	Multiply the unit vector with the point for the length. Multiply that length by the unit vector for the position.
Centring	PCA only works on centred data.
Eigenfaces	Flatten, subtract the mean image, use the small covariance matrix, convert the eigenvectors, project to get weights.
PCA versus LDA	PCA ignores labels and maximises spread. LDA uses labels and maximises class separation.