DR_PCA

May 22, 2024

0.0.1 Ý tưởng chính và bài toán PCA

1. **PCA là gì và tại sao nó được sử dụng?** PCA (Principal Component Analysis) là một kỹ thuật giảm chiều dữ liệu. Nó được sử dụng để chuyển đổi một tập dữ liệu với nhiều biến liên quan thành một tập dữ liệu mới với ít biến hơn, trong khi vẫn giữ lại phần lớn thông tin quan trọng. Điều này giúp giảm thiểu độ phức tạp của dữ liệu, làm tăng hiệu quả của các thuật toán học máy và phân tích dữ liệu.
2. **Những bài toán nào thường được giải quyết bằng PCA?** PCA thường được sử dụng trong các bài toán giảm chiều dữ liệu, trực quan hóa dữ liệu, phát hiện dữ liệu ngoại lai, và trước khi áp dụng các thuật toán học máy để cải thiện hiệu suất và tốc độ xử lý.
3. **PCA có thể giúp giảm chiều dữ liệu như thế nào?** PCA giảm chiều dữ liệu bằng cách tìm ra các hướng (principal components) trong không gian dữ liệu mà ở đó dữ liệu có sự biến thiên lớn nhất. Các principal components này là các tổ hợp tuyến tính của các biến gốc, và số lượng các components được giữ lại sẽ ít hơn số lượng biến gốc ban đầu.
4. **Ý tưởng chính của PCA là gì?** Ý tưởng chính của PCA là tìm ra các trục chính trong không gian dữ liệu mà trên đó dữ liệu có sự biến thiên lớn nhất. Các trục này được gọi là principal components và chúng là các tổ hợp tuyến tính của các biến gốc. PCA cố gắng tối đa hóa phương sai trên các trục này để giữ lại thông tin quan trọng nhất.
5. **Tại sao PCA được coi là một kỹ thuật giảm chiều phi tuyến tính?** Thực tế, PCA là một kỹ thuật giảm chiều tuyến tính. Nó tuyến tính ở chỗ nó tìm các tổ hợp tuyến tính của các biến gốc để tạo ra các principal components.
6. **Trong PCA, "principal component" có nghĩa là gì?** Principal component là các hướng trong không gian dữ liệu mà ở đó dữ liệu có sự biến thiên lớn nhất. Mỗi principal component là một tổ hợp tuyến tính của các biến gốc và chúng được sắp xếp theo thứ tự giảm dần của phương sai.
7. **Làm thế nào PCA có thể giúp trong việc phát hiện các đặc trưng chính của dữ liệu?** PCA giúp phát hiện các đặc trưng chính của dữ liệu bằng cách tìm ra các hướng trong không gian dữ liệu mà ở đó dữ liệu có sự biến thiên lớn nhất. Những hướng này chứa phần lớn thông tin quan trọng của dữ liệu, do đó, giữ lại các principal components tương ứng sẽ giúp ta tập trung vào các đặc trưng chính.

0.0.2 Input và Output của PCA

8. **Dữ liệu đầu vào cho PCA yêu cầu những gì?** Dữ liệu đầu vào cho PCA yêu cầu phải là một ma trận dữ liệu, trong đó các hàng là các mẫu và các cột là các biến. Dữ liệu thường

cần được chuẩn hóa hoặc chuẩn hóa để đảm bảo các biến có cùng đơn vị đo lường và tránh việc một biến có ảnh hưởng quá lớn.

9. **Dữ liệu đầu vào có cần phải chuẩn hóa không? Tại sao?** Có, dữ liệu đầu vào thường cần được chuẩn hóa (z-score normalization) để đảm bảo rằng tất cả các biến có cùng mức độ quan trọng. Nếu không chuẩn hóa, các biến có đơn vị đo lường lớn hơn sẽ chiếm ưu thế trong quá trình tính toán principal components.

10. **PCA tạo ra những output gì từ dữ liệu đầu vào?** PCA tạo ra các principal components (các tổ hợp tuyến tính của các biến gốc) và các eigenvalues tương ứng (đại diện cho phương sai dọc theo mỗi principal component). Ngoài ra, PCA còn tạo ra ma trận các coefficients để chuyển đổi dữ liệu gốc sang không gian của các principal components.

11. **Làm thế nào để chọn số lượng principal components cần giữ lại?** Số lượng principal components cần giữ lại thường được chọn dựa trên "explained variance". Các eigenvalues được sắp xếp theo thứ tự giảm dần, và ta chọn số lượng components sao cho tổng phương sai được giữ lại vượt qua một ngưỡng nhất định (thường là 90-95%).

12. **Biểu đồ Scree plot là gì và nó giúp gì trong PCA?** Scree plot là biểu đồ biểu diễn các eigenvalues theo thứ tự giảm dần. Nó giúp xác định số lượng principal components cần giữ lại bằng cách tìm điểm gấp khúc (elbow point) trên biểu đồ, tại đó các eigenvalues bắt đầu giảm chậm hơn.

13. **Eigenvalues và eigenvectors đóng vai trò gì trong PCA?** Eigenvalues biểu thị lượng phương sai mà mỗi principal component giữ lại, còn eigenvectors đại diện cho các hướng của các principal components trong không gian dữ liệu. Các eigenvectors tương ứng với các eigenvalues lớn nhất sẽ được chọn làm các principal components.

0.0.3 Thuật toán triển khai PCA

14. **Các bước chính của thuật toán PCA là gì?** Các bước chính của thuật toán PCA bao gồm:

    1. Chuẩn hóa dữ liệu.
    2. Tính toán ma trận covariance của dữ liệu chuẩn hóa.
    3. Tính các eigenvectors và eigenvalues của ma trận covariance.
    4. Sắp xếp các eigenvectors theo thứ tự giảm dần của các eigenvalues.
    5. Chọn số lượng principal components cần giữ lại.
    6. Chuyển đổi dữ liệu gốc sang không gian của các principal components.

15. **Tại sao chúng ta cần tính covariance matrix trong PCA?** Covariance matrix cho chúng ta biết mối quan hệ giữa các biến trong dữ liệu. Bằng cách phân tích covariance matrix, chúng ta có thể tìm ra các hướng (principal components) mà ở đó dữ liệu có sự biến thiên lớn nhất.

16. **Làm thế nào để tính toán các eigenvectors và eigenvalues của covariance matrix?** Các eigenvectors và eigenvalues của covariance matrix được tính bằng cách giải bài toán đặc trưng (eigenvalue problem) cho ma trận covariance. Điều này thường được thực hiện bằng cách sử dụng các thuật toán giải tích ma trận có sẵn trong các thư viện phần mềm như NumPy trong Python hoặc Eigen trong C++.

17. **Ý nghĩa của việc sắp xếp các eigenvalues theo thứ tự giảm dần trong PCA là**

gì? Sắp xếp các eigenvalues theo thứ tự giảm dần giúp xác định các principal components theo mức độ quan trọng của chúng. Các eigenvalues lớn nhất tương ứng với các principal components giữ lại phần lớn phương sai của dữ liệu.

18. **Làm thế nào để chuyển đổi dữ liệu gốc sang không gian của các principal components?** Dữ liệu gốc được chuyển đổi sang không gian của các principal components bằng cách nhân dữ liệu gốc đã chuẩn hóa với ma trận các eigenvectors tương ứng với các eigenvalues lớn nhất. Ma trận này là ma trận chuyển đổi từ không gian gốc sang không gian của các principal components.

19. **Khái niệm "explained variance" trong PCA là gì?** Explained variance biểu thị phần trăm tổng phương sai của dữ liệu gốc được giữ lại bởi các principal components đã chọn. Nó giúp đánh giá mức độ thông tin của dữ liệu gốc được giữ lại sau khi giảm chiều.

20. **Làm thế nào để quyết định số lượng components giữ lại dựa trên "explained variance"?** Số lượng components giữ lại được quyết định dựa trên explained variance bằng cách chọn các principal components sao cho tổng phương sai được giữ lại vượt qua một ngưỡng nhất định (thường là 90-95%). Điều này đảm bảo rằng phần lớn thông tin quan trọng của dữ liệu được giữ lại.

21. **Sự khác biệt giữa PCA và Factor Analysis là gì?** PCA tập trung vào việc giảm chiều dữ liệu và giữ lại phương sai lớn nhất, trong khi Factor Analysis tập trung vào việc tìm ra các nhân tố ẩn đằng sau dữ liệu và mô hình hóa cấu trúc liên kết giữa các biến. PCA sử dụng eigenvectors của ma trận covariance, trong khi Factor Analysis sử dụng các phép tính thống kê phức tạp hơn để tìm ra các nhân tố ẩn.

0.0.4 Ưu điểm và nhược điểm của PCA

22. **Những ưu điểm chính của PCA là gì?**

    - Giảm chiều dữ liệu, giúp giảm độ phức tạp và thời gian tính toán.
    - Giúp loại bỏ nhiễu bằng cách tập trung vào các components chứa nhiều thông tin nhất.
    - Giúp phát hiện các mối quan hệ và cấu trúc trong dữ liệu.
    - Hỗ trợ trực quan hóa dữ liệu trong không gian 2D hoặc 3D.

23. **PCA có những nhược điểm gì?**

    - PCA là kỹ thuật tuyến tính và không thể nắm bắt được các quan hệ phi tuyến tính trong dữ liệu.
    - Có thể làm mất thông tin quan trọng nếu số lượng principal components được chọn quá ít.
    - Yêu cầu dữ liệu đầu vào phải được chuẩn hóa.
    - PCA không dễ diễn giải vì các principal components là các tổ hợp tuyến tính của các biến gốc.

24. **PCA có thể gặp vấn đề gì khi dữ liệu đầu vào không được chuẩn hóa?** Nếu dữ liệu không được chuẩn hóa, các biến có đơn vị đo lường lớn hơn sẽ chiếm ưu thế trong quá trình tính toán principal components, dẫn đến việc các components không phản ánh đúng mối quan hệ giữa các biến.

25. **Tại sao PCA không

thể làm việc tốt với dữ liệu phi tuyến tính?** PCA là một kỹ thuật tuyến tính và chỉ tìm ra các tổ hợp tuyến tính của các biến gốc. Do đó, nó không thể nắm bắt được các mối quan hệ phi tuyến tính trong dữ liệu.

26. **PCA có thể làm giảm thông tin quan trọng nào trong dữ liệu?** Nếu số lượng principal components được chọn quá ít, PCA có thể loại bỏ các components chứa thông tin quan trọng nhưng có phương sai nhỏ. Điều này dẫn đến mất mát thông tin quan trọng.

27. **Ưu điểm của việc sử dụng PCA trong việc giảm chiều dữ liệu trước khi áp dụng các thuật toán học máy là gì?**

    - Giảm độ phức tạp của dữ liệu, giúp cải thiện hiệu suất và tốc độ của các thuật toán học máy.
    - Loại bỏ nhiễu và các biến ít quan trọng, giúp tăng độ chính xác của các mô hình.
    - Giúp tránh overfitting bằng cách giảm số lượng features.

28. **Tại sao PCA có thể không hiệu quả khi số lượng các principal components giữ lại quá ít?** Khi số lượng principal components giữ lại quá ít, PCA có thể loại bỏ các components chứa thông tin quan trọng, dẫn đến mất mát thông tin và giảm độ chính xác của các mô hình học máy.

29. **PCA có thể được cải thiện bằng cách sử dụng các kỹ thuật nào khác?**

    - Kernel PCA: mở rộng PCA cho dữ liệu phi tuyến tính bằng cách sử dụng kernel trick.
    - t-SNE: kỹ thuật giảm chiều phi tuyến tính giúp trực quan hóa dữ liệu trong không gian 2D hoặc 3D.
    - Autoencoders: mô hình học sâu để giảm chiều dữ liệu phi tuyến tính.

0.0.5 Ứng dụng và ví dụ của PCA

30. **Một số ứng dụng thực tế của PCA trong phân tích dữ liệu là gì?**

    - Phân tích hình ảnh: giảm chiều dữ liệu ảnh để nén ảnh hoặc trích xuất đặc trưng.
    - Phân tích gene: giảm chiều dữ liệu gene để tìm ra các gene quan trọng.
    - Nhận dạng mẫu: giảm chiều dữ liệu để nhận dạng khuôn mặt, chữ viết tay, v.v.
    - Thị trường tài chính: phân tích dữ liệu tài chính để phát hiện các yếu tố chính ảnh hưởng đến giá cổ phiếu.

31. **Làm thế nào PCA có thể được sử dụng trong lĩnh vực nhận dạng mẫu?** PCA có thể giảm chiều dữ liệu đầu vào, giúp tập trung vào các đặc trưng quan trọng nhất, từ đó cải thiện hiệu suất của các thuật toán nhận dạng mẫu như SVM, KNN, và neural networks.

32. **PCA có thể giúp gì trong việc phát hiện và xử lý dữ liệu ngoại lai?** PCA giúp phát hiện dữ liệu ngoại lai bằng cách giảm chiều dữ liệu và xác định các điểm nằm xa so với các principal components. Những điểm này có thể là các ngoại lai trong dữ liệu.

33. **Làm thế nào PCA có thể giúp trong việc trực quan hóa dữ liệu?** PCA giúp giảm chiều dữ liệu xuống 2 hoặc 3 dimensions, cho phép chúng ta trực quan hóa dữ liệu trong không gian 2D hoặc 3D, giúp dễ dàng phát hiện các cụm, xu hướng và mối quan hệ trong dữ liệu.

34. **Một ví dụ cụ thể về việc sử dụng PCA trong xử lý ảnh là gì?** PCA có thể được sử dụng để nén ảnh bằng cách giảm chiều dữ liệu của ảnh. Ví dụ, trong nhận dạng khuôn mặt,

PCA có thể được sử dụng để trích xuất các đặc trưng chính của khuôn mặt và lưu trữ chúng dưới dạng các principal components, giảm thiểu kích thước dữ liệu ảnh mà vẫn giữ lại phần lớn thông tin quan trọng.

0.0.6 Đánh giá output thu được từ PCA

35. **Làm thế nào để đánh giá chất lượng của output thu được từ PCA?**

    Để đánh giá chất lượng của output thu được từ PCA, chúng ta có thể sử dụng các phương pháp sau:

    - Explained Variance Ratio: Kiểm tra tỷ lệ phương sai được giải thích bởi mỗi principal component. Tổng explained variance ratio của các components giữ lại nên càng cao càng tốt, thường là trên 90% để đảm bảo rằng phần lớn thông tin quan trọng được giữ lại.
    - Scree Plot: Sử dụng scree plot để xác định số lượng principal components cần giữ lại. Điểm gấp khúc (elbow point) trong scree plot cho thấy số lượng components tối ưu cần chọn.
    - Cumulative Explained Variance: Kiểm tra tổng explained variance theo cumulative. Chúng ta chọn số lượng components sao cho tổng cumulative explained variance đạt đến một ngưỡng nhất định (thường là 90-95%).
    - Reconstruction Error: Đo lường sai số giữa dữ liệu gốc và dữ liệu được tái cấu trúc từ các principal components. Reconstruction error càng nhỏ thì chất lượng PCA càng tốt.
    - Visual Inspection: Trực quan hóa dữ liệu sau khi giảm chiều bằng PCA. Điều này giúp chúng ta thấy rõ hơn các cấu trúc, xu hướng và cụm trong dữ liệu.

36. **Làm thế nào để tính toán và đánh giá độ mất mát thông tin trong PCA?**

    Để tính toán và đánh giá độ mất mát thông tin trong PCA, chúng ta có thể làm theo các bước sau:

    - Tính toán Reconstruction Error: Dùng các principal components để tái cấu trúc lại dữ liệu và so sánh với dữ liệu gốc. Reconstruction Error là trung bình bình phương sai số (Mean Squared Error - MSE) giữa dữ liệu gốc và dữ liệu tái cấu trúc.
    - So sánh Explained Variance: Tổng explained variance của các principal components giữ lại nên càng cao càng tốt. Điều này đảm bảo phần lớn thông tin được giữ lại.
    - Visualization: Trực quan hóa dữ liệu trước và sau khi thực hiện PCA để xem cấu trúc dữ liệu có bị thay đổi đáng kể không.