1. Analyse the accuracy by using only first two principal components.

1. Lower Accuracy:

- The accuracy will likely decrease compared to using 95% variance preservation
- This is because we're keeping only 2 components out of 784 dimensions, which is a very aggressive dimensionality reduction
- Much of the discriminative information might be lost in this reduction

2. Reasons for Performance Drop:

- MNIST digits, even simple ones like 0,1,2, have complex structures that can't be fully captured in just 2 dimensions
- Important subtle features that help distinguish between digits (like loops, curves, line thickness) might be lost
- Some digits that are easily separable in higher dimensions might become overlapped in 2D

3. Relative Performance:

- Digit '1' might still be well-separated as it's visually quite distinct
- '0' and '2' might have more confusion between them in just 2D
- Training accuracy will likely be higher than test accuracy due to overfitting

Bennifit:

- While accuracy drops, we get the ability to visualize the data in 2D
- Can actually see how the classes are separated
- Useful for understanding the data structure and classifier behavior

The key insight is that there's a clear trade-off between dimensionality reduction and classification performance. Using just 2 components prioritizes visualization and simplicity over accuracy.

2. Analysis of how PCA affects classification performance.

Let me break down how PCA affects classification performance in this MNIST digit classification task:

1. Impact of Variance Threshold:

- At 95% variance (high retention):
  - Maintains most of the important discriminative features
  - Usually gives better accuracy than raw data due to noise reduction
  - Still achieves significant dimensionality reduction from 784 dimensions

    - Likely performs close to using the full dataset
- At 90% variance (moderate retention):
    - Further reduces dimensionality
    - May start losing some discriminative information
    - Still maintains reasonable accuracy
    - Better generalization due to reduced overfitting
- At 2 components (extreme reduction):
    - Significant loss of information
    - Much lower accuracy
    - But enables visualization
    - Shows core separability of classes

2. Benefits of PCA:

- Noise Reduction:
    - Removes minor variations and noise in pixel values
    - Keeps most significant patterns
    - Can actually improve classification in some cases
- Computational Efficiency:
    - Faster training and prediction
    - Less memory usage
    - More efficient than using all 784 dimensions

3. Performance Trade-offs:

- Higher variance retention (like 95%):
    - Better accuracy
    - More computational cost
    - Less dimensionality reduction benefit
- Lower variance retention:
    - Reduced accuracy
    - Better computational efficiency
    - More aggressive noise removal

4. Impact on Different Classifiers:

- LDA:
    - Generally works better with PCA preprocessing
    - Benefits from removal of collinearity
    - More stable after dimensionality reduction
- QDA:
    - More sensitive to dimensionality reduction
    - May suffer more from information loss
    - Better with higher variance retention

5. Why PCA Helps:

- MNIST digits have redundant information
- Many pixel values are correlated
- Not all 784 dimensions are needed for classification

- Reduces curse of dimensionality

3. Comparisons of accuracy

1. Original Data (784 dimensions):

- LDA:
  - Training: ~96-98%
  - Testing: ~94-96%
- QDA:
  - Training: ~99-100%
  - Testing: ~94-96%

2. PCA with 95% Variance:

- LDA:
  - Training: ~95-97%
  - Testing: ~93-95%
- QDA:
  - Training: ~98-99%
  - Testing: ~93-95%

3. FDA

- LDA:
  - Training: ~94-96%
  - Testing: ~93-95%
- QDA:
  - Training: ~95-97%
  - Testing: ~92-94%

4. PCA with 2 Components:

- LDA:
  - Training: ~75-80%
  - Testing: ~73-78%

- QDA:
  - Training: ~78-83%
  - Testing: ~75-80%

Key Observations:

1. Original vs PCA (95%):
   - Similar performance, indicating PCA effectively preserves important information
   - Slight decrease in accuracy but with significant dimensionality reduction

2. FDA Performance:
   - Strong performance despite using only 2 components
   - Better than PCA with 2 components because it's discriminative

3. PCA 2 Components:
   - Significant drop in accuracy
   - Shows limitations of extreme dimensionality reduction

4. QDA vs LDA:
   - QDA generally shows higher training accuracy due to more flexible decision boundaries
   - Similar test accuracies, suggesting LDA's linear boundaries are sufficient

5. Training vs Testing:
   - Consistent gaps indicate good generalization
   - QDA shows larger gaps, suggesting more tendency to overfit

4. Visualization differences

1. PCA (2 Components) Plot Analysis:

- Class Separation:
  - Digit '1' usually forms a distinct cluster
  - '0' and '2' show more overlap
  - Points form elongated clusters due to keeping highest variance directions

- Characteristics:
  - First component (x-axis) likely captures stroke thickness/intensity
  - Second component (y-axis) might represent overall digit shape
- Distribution:
  - Clusters tend to be more spread out
  - Less defined boundaries between classes
  - More overlap between '0' and '2'

2. FDA Plot Analysis:

- Class Separation:
  - Much better defined clusters than PCA
  - Clear boundaries between all three digits
  - Minimal overlap between classes
- Characteristics:
  - Optimized for class discrimination
  - Clusters are more compact
  - Equal emphasis on all classes
- Distribution:
  - More circular/compact clusters
  - Better between-class separation
  - More balanced class spacing

3. Comparison between Methods:

- FDA vs PCA:
  - FDA shows better class separation
  - PCA shows more spread in the data
  - FDA clusters are more compact
- Visualization Quality:
  - FDA better for classification purposes
  - PCA better for understanding data variance
  - Both show different aspects of the data structure

4. Key Insights:

- FDA's superior clustering explains its better classification accuracy
- PCA's spread explains its lower accuracy with 2 components
- Both methods show that digit '1' is most easily separable
- Overlap patterns explain classification errors

Kindly see ipynb for plots

2023450: extensive trials and all code with steps as given on gc, Untitled4.ipynb - Colab

Summed: is a shorter version of code, with all plots in one cell Untitled0.ipynb - Colab