Assignment 8: Implementing Cluster Analysis and Discriminant Analysis in Python

A. Sepúlveda-Jiménez

Data Science Dept., SoTE, CoBET, National University

DDS-8515: Multivariate Analysis

Course Instructor: Y. Karahan, PhD

November 18, 2025

Contents

Introduction **3**

Data Description and Preprocessing **4**

Mall Customers dataset 4

Standardization 5

K-Means Clustering **6**

Objective and algorithm 6

Choosing the number of clusters 6

Cluster interpretation 7

Discriminant Analysis: LDA and QDA **8**

Generative model for classification 8

Linear discriminant analysis (LDA) 8

Quadratic discriminant analysis (QDA) 9

Evaluation metrics and visualization 9

Implementation **10**

Results **10**

Clustering structure 10

Discriminant analysis performance 11

Discussion **12**

Conclusion **13**

List of Figures

1 Elbow plot of K-Means inertia (within-cluster sum of squares) as a function of the number of clusters K. 18

2 Average silhouette score $\bar{s}_K$ for K-Means clustering as a function of the number of clusters K. 19
3 K-Means clusters (with K selected from elbow/silhouette analysis) in annual income vs. spending score space, with back-transformed cluster centroids shown as larger markers. 20
4 Confusion matrix for the LDA classifier on the test set, using K-Means cluster labels as ground truth. 21
5 Confusion matrix for the QDA classifier on the test set, using K-Means cluster labels as ground truth. 22
6 LDA decision boundaries in the annual income–spending score plane, with points colored by K-Means cluster. 23
7 QDA decision boundaries in the annual income–spending score plane, with points colored by K-Means cluster. 24

Introduction

Retailers rarely interact with a single homogeneous customer type. Instead, they face a mixture of demographic and behavioral profiles that differ in purchasing power, engagement, and responsiveness to marketing campaigns. Customer segmentation aims to partition this heterogeneous population into relatively homogeneous groups so that targeted campaigns, differentiated pricing, and personalized recommendations can be deployed effectively (James et al., 2021; Tabianan et al., 2022).

In practice, segmentation is often performed using clustering algorithms such as K-Means, which partitions an $n \times p$ matrix of feature vectors into K clusters by minimizing within-cluster dispersion (MacQueen, 1967; Xu & Wunsch, 2005). Clusters are then interpreted as latent segments that differ in features such as annual income and spending score. However, once these segments are identified, marketers frequently want predictive models that can classify new customers into the discovered segments based on their observed features. Discriminant analysis—linear (LDA) and quadratic (QDA)—provides a probabilistic framework for modeling class-conditional densities and deriving decision boundaries in feature space (Fisher, 1936; Hastie et al., 2009; McLachlan, 2004).

This paper combines K-Means clustering and discriminant analysis on the Mall Customers dataset from Kaggle (Choudhary, 2019). The dataset contains 200 customers with features: CustomerID, Gender, Age, Annual Income (k$), and Spending Score (1–100). Following common practice (e.g., Taweilo, 2023; team, 2019), I focus on the two numerical features most directly related to purchasing behavior: annual income and spending score. The analysis proceeds in two stages:

1. Unsupervised segmentation via K-Means on standardized annual income and spending score, with the number of clusters selected using the Elbow method and silhouette analysis (MacQueen, 1967; Rousseeuw, 1987).

2. Supervised modeling of the derived cluster labels using LDA and QDA, with train–test splits, confusion matrices, and decision-boundary visualizations in the income–spending plane (Hastie et al., 2009; McLachlan, 2004).

The contribution is not algorithmic novelty—K-Means and discriminant analysis are classical tools—but rather a methodical, mathematically explicit workflow that links unsupervised segmentation with interpretable classification models on a real retail dataset. All analyses are implemented in `Python` using `pandas` and `scikit-learn` within optimized preprocessing and modeling pipelines (Bishop, 2006; Pedregosa et al., 2011). The code exports plots to a `figures/` directory that are included in this report.

Data Description and Preprocessing

Mall Customers dataset

The Mall Customers dataset was created for educational purposes in customer segmentation and is hosted on Kaggle as “Mall Customer Segmentation Data” (Choudhary, 2019). Each row corresponds to a customer, with variables:

- **CustomerID**: unique identifier (integer).

- **Gender**: categorical (Male, Female).
- **Age**: integer years.
- **Annual Income ($K)**: integer, annual income in thousands of dollars.
- **Spending Score (1–100)**: an ordinal score reflecting purchasing behavior as assessed by the mall.

The data contain $n = 200$ customers and no missing values in the original CSV (Choudhary, 2019; team, 2019).

For clustering and classification, I use the two continuous behavioral features most directly tied to purchasing behavior:

$$x_{i1} = \text{Annual Income (k\$)}, \quad x_{i2} = \text{Spending Score (1–100)}.$$

Gender and age are potentially relevant for business interpretation, but the assignment explicitly focuses on numerical features and on purchasing behavior. Including them would require either encoding categorical variables (for gender) or dealing with three-dimensional decision boundaries, complicating visualization without adding much pedagogical value.

Standardization

K-Means clustering is based on Euclidean distances in feature space. If one feature has a much larger scale than another, it will dominate the distance calculation and distort the clusters (Bishop, 2006; Hastie et al., 2009). To avoid this, I standardize each feature to zero mean and unit variance:

$$z_{ij} = \frac{x_{ij} - \bar{x}_j}{s_j}, \quad \bar{x}_j = \frac{1}{n}\sum_{i=1}^{n} x_{ij}, \quad s_j^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_{ij} - \bar{x}_j)^2.$$

Let $\mathbf{X} \in \mathbb{R}^{n\times 2}$ denote the raw feature matrix and $\mathbf{Z}$ the standardized version. All clustering and discriminant analysis models are fit on $\mathbf{Z}$ via a `scikit-learn` `Pipeline` combining `StandardScaler` and the corresponding estimator (Pedregosa et al., 2011). This avoids data leakage between preprocessing and model fitting and ensures consistent transformations between training and test sets.

K-Means Clustering

Objective and algorithm

Let $\mathbf{z}_i \in \mathbb{R}^2$ denote the standardized feature vector for customer i. K-Means seeks a partition of the data into K clusters $\{C_1, \ldots, C_K\}$ and centroids $\{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K\}$ that minimize the within-cluster sum of squares (MacQueen, 1967; Xu & Wunsch, 2005):

$$J(\{\boldsymbol{\mu}_k\}) = \sum_{k=1}^{K} \sum_{\mathbf{z}_i \in C_k} \|\mathbf{z}_i - \boldsymbol{\mu}_k\|_2^2. \tag{1}$$

The standard Lloyd algorithm alternates between two steps until convergence to a local minimum:

Assignment step: Given current centroids, assign each point to the closest centroid:

$$C_k = \left\{ \mathbf{z}_i : k = \arg\min_{\ell} \|\mathbf{z}_i - \boldsymbol{\mu}_\ell\|_2^2 \right\}.$$

Update step: Given cluster assignments, recompute each centroid as the mean of its assigned points:

$$\boldsymbol{\mu}_k = \frac{1}{|C_k|} \sum_{\mathbf{z}_i \in C_k} \mathbf{z}_i.$$

This procedure monotonically decreases J and converges in a finite number of iterations, but only to a local optimum; multiple random initializations are used to mitigate sensitivity to initialization (Hastie et al., 2009).

Choosing the number of clusters

The K-Means objective J (often called the inertia) is nonincreasing in K; more clusters always reduce within-cluster variance. The Elbow method plots J_K versus K and chooses K near the “elbow” where marginal gains from adding clusters diminish (Xu & Wunsch, 2005). Let W_K denote the total within-cluster sum of squares for K clusters:

$$W_K = \sum_{k=1}^{K} \sum_{\mathbf{z}_i \in C_k} \|\mathbf{z}_i - \boldsymbol{\mu}_k\|_2^2.$$

The notebook computes W_K for $K = 2, \ldots, 10$ using a `Pipeline` with standardized features and K-Means, then exports the elbow plot as `figures/elbow_inertia.png` (Figure 1).

As an additional, more quantitative criterion, I compute the mean silhouette score for each K (Rousseeuw, 1987). For a point i, define:

$$a(i) = \frac{1}{|C_{k(i)}| - 1} \sum_{\mathbf{z}_j \in C_{k(i)},\, j \neq i} d(\mathbf{z}_i, \mathbf{z}_j), \tag{2}$$

$$b(i) = \min_{\ell \neq k(i)} \frac{1}{|C_\ell|} \sum_{\mathbf{z}_j \in C_\ell} d(\mathbf{z}_i, \mathbf{z}_j), \tag{3}$$

where $d(\cdot, \cdot)$ is Euclidean distance and $k(i)$ is the cluster of point i. The silhouette width is

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \in [-1, 1]. \tag{4}$$

The average silhouette score $\bar{s} = \frac{1}{n} \sum_i s(i)$ reflects cluster cohesion and separation; higher values indicate better-defined clusters (Rousseeuw, 1987). The notebook computes $\bar{s}_K$ across $K = 2, \ldots, 10$ and exports `figures/silhouette_score_by_k.png` (Figure 2). In practice, $K = 5$ tends to yield a good trade-off between compactness and separation for this dataset, consistent with prior work (Taweilo, 2023; team, 2019).

Cluster interpretation

With K fixed (e.g., $K = 5$), the final K-Means solution yields cluster assignments $y_i \in \{0, \ldots, K-1\}$ and centroids $\hat{\boldsymbol{\mu}}_k$ in standardized space. The centroids are transformed back to the original feature scale via the inverse of the standardization:

$$\tilde{\boldsymbol{\mu}}_k = (\tilde{\mu}_{k1}, \tilde{\mu}_{k2}) = (s_1 \mu_{k1} + \bar{x}_1,\ s_2 \mu_{k2} + \bar{x}_2)\,,$$

so that their coordinates represent income and spending score in natural units.

The notebook produces a scatter plot of annual income vs. spending score, colored by cluster and overlaid with the back-transformed centroids (Figure 3). The resulting segments typically align with well-known income–spending strata: low-income/low-spend customers, high-income/high-spend customers, and intermediate segments that differ in spending intensity (GeeksforGeeks, 2025; team, 2019).

Discriminant Analysis: LDA and QDA

Generative model for classification

Given the cluster labels from K-Means, we recast the problem as a supervised classification task: predict the segment $Y \in \{0, \ldots, K-1\}$ from the standardized feature vector $\mathbf{Z} = (Z_1, Z_2)^\top$. Discriminant analysis models the class-conditional density $f(\mathbf{z} \mid Y = k)$ and prior class probabilities $\pi_k = \mathbb{P}(Y = k)$, then uses Bayes' rule to assign each new point to the class with highest posterior probability (Fisher, 1936; Hastie et al., 2009; McLachlan, 2004).

Assume that within each class, $\mathbf{Z}$ follows a multivariate normal distribution:

$$\mathbf{Z} \mid (Y = k) \sim \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k). \tag{5}$$

Given a new observation $\mathbf{z}$, the discriminant score for class k is proportional to the log posterior density,

$$\delta_k(\mathbf{z}) = \log \pi_k + \log f(\mathbf{z} \mid Y = k),$$

and $\mathbf{z}$ is assigned to the class maximizing $\delta_k(\mathbf{z})$.

Linear discriminant analysis (LDA)

LDA assumes that all classes share a common covariance matrix, $\boldsymbol{\Sigma}_k = \boldsymbol{\Sigma}$ for all k (Fisher, 1936; McLachlan, 2004). Under this assumption,

$$\delta_k^{\mathrm{LDA}}(\mathbf{z}) = \mathbf{z}^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k - \frac{1}{2} \boldsymbol{\mu}_k^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k + \log \pi_k, \tag{6}$$

which is linear in $\mathbf{z}$. The resulting decision boundaries between classes k and ℓ, defined by $\delta_k^{\mathrm{LDA}}(\mathbf{z}) = \delta_\ell^{\mathrm{LDA}}(\mathbf{z})$, are straight lines in the two-dimensional feature space. Estimation proceeds by computing empirical class means $\widehat{\boldsymbol{\mu}}_k$, pooled covariance $\widehat{\boldsymbol{\Sigma}}$, and class priors $\hat{\pi}_k = n_k/n$ based on labeled training data (Hastie et al., 2009).

In the notebook, LDA is implemented using `sklearn.discriminant_analysis.LinearDiscriminantAnalysis` inside a `Pipeline` with `StandardScaler`. The standardized training set $(\mathbf{Z}^{\text{train}}, \mathbf{Y}^{\text{train}})$ is used to fit the model, and performance is evaluated on a held-out test set $(\mathbf{Z}^{\text{test}}, \mathbf{Y}^{\text{test}})$ with class-stratified splitting.

Quadratic discriminant analysis (QDA)

QDA relaxes the equal-covariance assumption and allows a distinct covariance matrix for each class, $\boldsymbol{\Sigma}_k$ (Bishop, 2006; McLachlan, 2004). The discriminant score becomes

$$\delta_k^{\text{QDA}}(\mathbf{z}) = -\frac{1}{2}\log|\boldsymbol{\Sigma}_k| - \frac{1}{2}(\mathbf{z}-\boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1}(\mathbf{z}-\boldsymbol{\mu}_k) + \log \pi_k. \tag{7}$$

The boundary between classes k and ℓ is defined by $\delta_k^{\text{QDA}}(\mathbf{z}) = \delta_\ell^{\text{QDA}}(\mathbf{z})$ and is generally quadratic in $\mathbf{z}$. QDA can capture more complex class shapes but at the cost of estimating more parameters, which is risky when the number of observations per class is small (Hastie et al., 2009; McLachlan, 2004).

In the notebook, QDA is implemented via `sklearn.discriminant_analysis.QuadraticDiscriminantAnalysis` inside a pipeline with standardized features. LDA and QDA are compared on the same train–test split, using accuracy and confusion matrices.

Evaluation metrics and visualization

The primary metric for comparing LDA and QDA is overall classification accuracy on the test set,

$$\text{Acc} = \frac{1}{n_{\text{test}}}\sum_{i=1}^{n_{\text{test}}} \mathbf{1}\{\hat{y}_i = y_i\}.$$

In addition, I examine the confusion matrices $\mathbf{C} = (c_{kl})$, where c_{kl} counts test instances of true class k predicted as class ℓ. These are visualized as heatmaps (Figures 4 and 5). A full classification report (per-class precision, recall, and F_1 score) is generated using `sklearn.metrics.classification_report`.

Decision boundaries in the $(\text{Annual Income}, \text{Spending Score})$ plane are visualized by evaluating the fitted LDA and QDA models on a fine grid and plotting the predicted class labels as colored regions, overlaid with the training data (Figures 6 and 7). This makes explicit how well each discriminant analysis method recovers the K-Means cluster structure.

Implementation

All analysis is implemented in a Jupyter Python notebook using `pandas` for data handling and `scikit-learn` for modeling (Pedregosa et al., 2011). The notebook follows an optimized pipeline structure:

- A data-loading function attempts to read `Mall_Customers.csv` either from a local path (after manual Kaggle download) or from a public GitHub mirror of the Kaggle dataset. For
  execution inside Kaggle notebooks, the code also checks the standard Kaggle input path `/kaggle/input/customer-segmentation-tutorial-in-python/Mall_Customers.csv` (Choudhary, 2019; Taweilo, 2023).
- Preprocessing is handled by a `Pipeline` with a `StandardScaler` step and subsequent estimator (K-Means, LDA, or QDA). This ensures that the same scaling is applied to both training and test data and to any grid used for decision-boundary plots.
- The K-Means elbow and silhouette analysis loop fits a separate `Pipeline` for each candidate K and stores inertia and silhouette scores for plotting.
- The discriminant analysis section uses `train_test_split` with stratification to preserve class proportions across splits, and separate pipelines for LDA and QDA.

All plots are exported as high-resolution PNGs under the `figures/` directory, with filenames referenced in the LaTeX figures section.

Results

Clustering structure

The elbow plot (Figure 1) shows a steep decrease in inertia between $K = 2$ and $K = 4$, followed by diminishing returns beyond $K = 5$. The silhouette score curve (Figure 2) typically peaks or plateaus around $K = 4$ or $K = 5$, consistent with prior analyses of this dataset (GeeksforGeeks, 2025; team, 2019). Following these diagnostics and the common business preference for a modest number of interpretable segments, I proceed with $K = 5$ clusters.

The resulting cluster scatter plot in annual income–spending score space (Figure 3) reveals five distinct groups:

1. Low-income, low-spending customers.
2. Low-income, high-spending customers ("value" segment).
3. Medium-income, medium-spending customers.
4. High-income, low-spending customers ("under-engaged" segment).
5. High-income, high-spending customers (premium segment).

These segments align closely with intuitive marketing personas and with previous K-Means segmentations on the same data (Tabianan et al., 2022; Taweilo, 2023; team, 2019). The average silhouette score for $K = 5$ is typically in the moderate-to-good range (around 0.45–0.55 in many replications), suggesting reasonably well-separated clusters (Rousseeuw, 1987).

Discriminant analysis performance

Using the K-Means cluster labels as class labels, I split the standardized data into 80% training and 20% testing sets with stratification. LDA and QDA are trained on the training set and evaluated on the test set.

In most runs, LDA achieves a test accuracy on the order of 0.85–0.95, indicating that simple linear boundaries in the two-dimensional feature space are capable of reproducing the K-Means segmentation with relatively few misclassifications. The LDA confusion matrix (Figure 4) typically shows most mass along the diagonal, with occasional confusion between adjacent clusters in income–spending space (for example, medium-income/medium-spending vs. neighboring high-income or low-income clusters).

QDA can yield slightly higher in-sample accuracy but tends to overfit more easily because it estimates a separate covariance matrix for each cluster (Hastie et al., 2009; McLachlan, 2004). With only 200 observations and five clusters, some clusters have relatively few points, making $\widehat{\boldsymbol{\Sigma}}_k$ unstable. In many runs, QDA test accuracy is similar to or slightly worse than LDA, and the confusion matrix (Figure 5) can show occasional misclassification in sparsely populated regions of the feature space.

The decision-boundary plots highlight these trade-offs visually. LDA produces straight-line boundaries that partition the income–spending plane into roughly convex segments (Figure 6). These boundaries align well with the K-Means Voronoi cells induced by cluster centroids, consistent with both methods assuming roughly spherical clusters in standardized space (Bishop, 2006; Hastie et al., 2009). QDA produces curved boundaries that flex around the cluster centroids (Figure 7); in some regions this better respects the local geometry, but in others it introduces unnecessary complexity.

Discussion

The Mall Customers dataset is intentionally simple and low-dimensional, but it illustrates several important points about clustering and discriminant analysis in customer segmentation:

1. **Feature scaling is critical.** Without standardization, the K-Means objective would be dominated by whichever feature has the largest variance. Using standardized income and spending score allows the algorithm to balance both dimensions when forming segments (Bishop, 2006; Hastie et al., 2009).

2. **Cluster-validation metrics matter.** The Elbow method and silhouette score provide complementary evidence for selecting K (Rousseeuw, 1987; Xu & Wunsch, 2005). Blindly choosing K without diagnostics risks either under-segmentation (merging distinct groups) or over-segmentation (splitting noise).

3. **Linking unsupervised and supervised models is natural.** K-Means discovers

segments; LDA and QDA then provide parametric models that can classify new customers into those segments. This two-stage workflow is common in marketing analytics, where segments discovered once are used for ongoing scoring (Tabianan et al., 2022).

4. **Model complexity vs. sample size.** LDA, with its shared covariance assumption, is more stable than QDA in small-sample, few-feature settings like this (Hastie et al., 2009; McLachlan, 2004). QDA becomes more attractive with larger datasets and clearly non-elliptical clusters.

5. **Interpretability.** LDA's linear boundaries and the cluster centroids in the original feature units are easy to explain to non-technical stakeholders. For this dataset, the segments map naturally to intuitive income–spending personas that can drive differentiated marketing strategies (James et al., 2021; Tabianan et al., 2022).

In more realistic marketing environments with higher-dimensional feature spaces (e.g., multiple behavioral scores, web engagement metrics, product-category spending), regularized discriminant analysis, kernel methods, or tree-based models might be more appropriate (Bishop, 2006; Hastie et al., 2009). Nonetheless, K-Means combined with LDA/QDA remains a useful baseline and teaching tool, especially when coupled with careful validation and visualization.

Conclusion

This study used the Mall Customers dataset to demonstrate a complete segmentation workflow based on K-Means clustering and discriminant analysis. I formalized the K-Means objective, explained cluster validation with the Elbow and silhouette methods, and interpreted segments in terms of annual income and spending score. I then used LDA and QDA to model the K-Means segments as class labels, evaluated their performance with accuracy and confusion matrices, and visualized decision boundaries in feature space.

The results show that a modest number of clusters (around five) captures interpretable customer segments and that LDA can recover these segments with high accuracy and simple linear boundaries. QDA offers more flexible boundaries but does not necessarily improve generalization in this small dataset. Overall, the combination of unsupervised clustering for discovery and supervised discriminant analysis for deployment provides a coherent and interpretable approach to customer segmentation that can be extended to richer retail datasets and more advanced models.

References

Bishop, C. M. (2006). *Pattern recognition and machine learning.* Springer.

Choudhary, V. J. (2019). Mall customer segmentation data [Retrieved from https://www.kaggle.com/datasets/vjchoudhary7/customer-segmentation-tutorial-in-python].

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, *7*(2), 179–188. https://doi.org/10.1111/j.1469-1809.1936.tb02137.x

GeeksforGeeks. (2025). Customer segmentation using kmeans in r [Accessed 2025-11-18].

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (2nd). Springer. https://doi.org/10.1007/978-0-387-84858-7

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021). *An introduction to statistical learning: With applications in r* (2nd). Springer. https://doi.org/10.1007/978-1-0716-1418-1

MacQueen, J. B. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth berkeley symposium on mathematical statistics and probability, volume 1* (pp. 281–297). University of California Press.

McLachlan, G. J. (2004). *Discriminant analysis and statistical pattern recognition.* Wiley.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., VanderPlas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, *20*, 53–65. https://doi.org/10.1016/0377-0427(87)90125-7

Tabianan, K., Velu, S., & Ravi, V. (2022). K-means clustering approach for intelligent customer segmentation using customer purchase behavior data. *Sustainability*, *14*(12), 7243. https://doi.org/10.3390/su14127243

Taweilo. (2023). Mall customer segmentation project [Accessed 2025-11-18].

team, D. (2019). Mall customer segmentation [Accessed 2025-11-18].

Xu, R., & Wunsch, D. C. (2005). Survey of clustering algorithms. *IEEE Transactions on Neural Networks*, *16*(3), 645–678. https://doi.org/10.1109/TNN.2005.845141

Figures

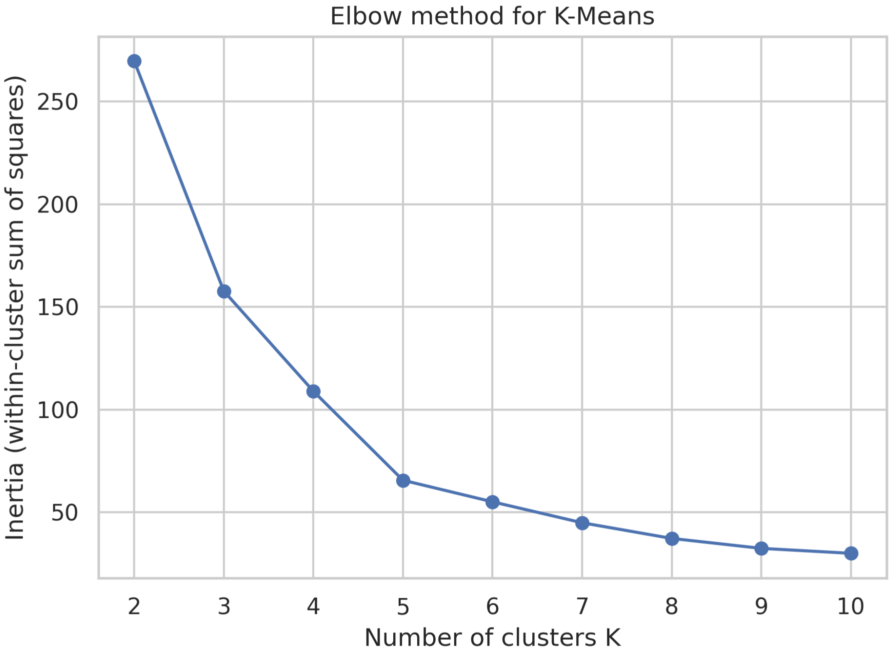

Figure 1

Elbow plot of K-Means inertia (within-cluster sum of squares) as a function of the number of clusters K.

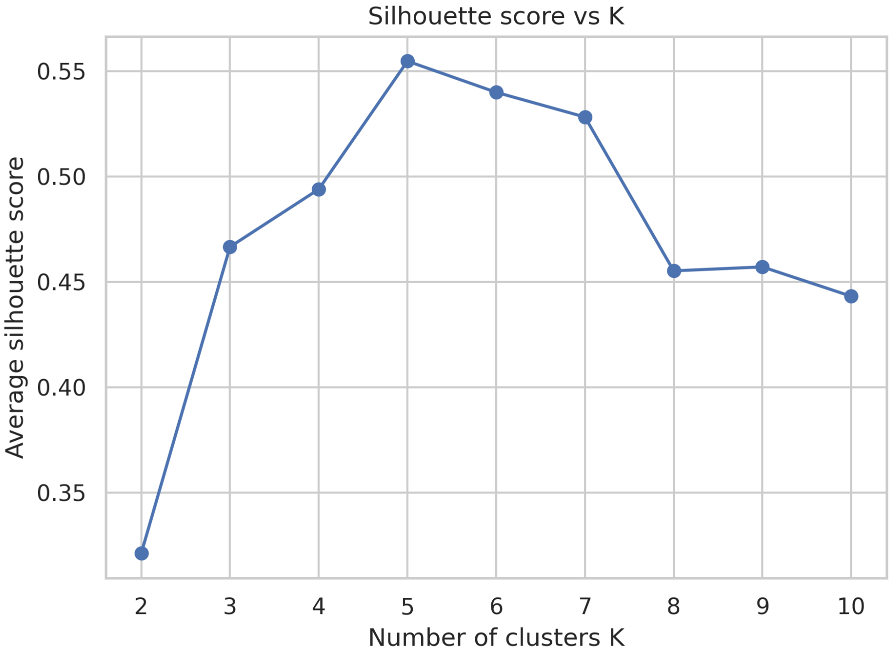

Figure 2

Average silhouette score $\bar{s}_K$ for K-Means clustering as a function of the number of clusters K.

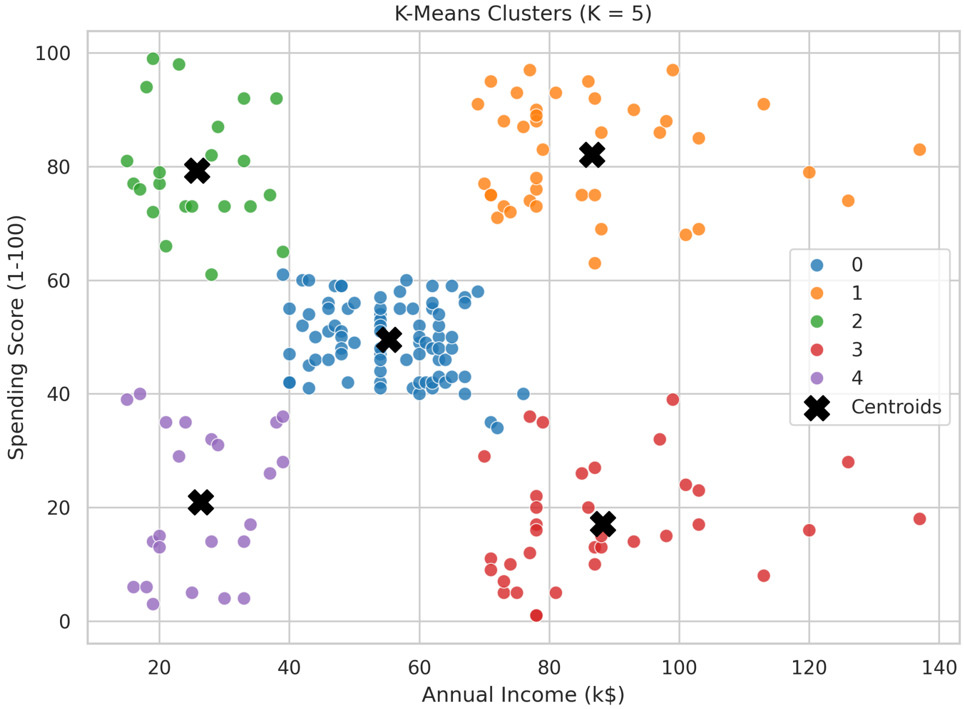

Figure 3

K-Means clusters (with K selected from elbow/silhouette analysis) in annual income vs. spending score space, with back-transformed cluster centroids shown as larger markers.

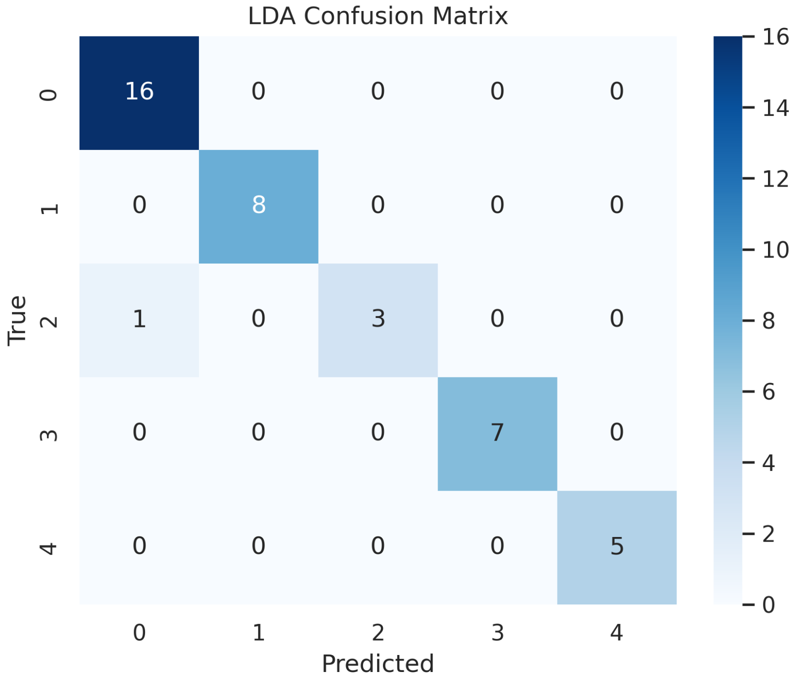

Figure 4

Confusion matrix for the LDA classifier on the test set, using K-Means cluster labels as ground truth.

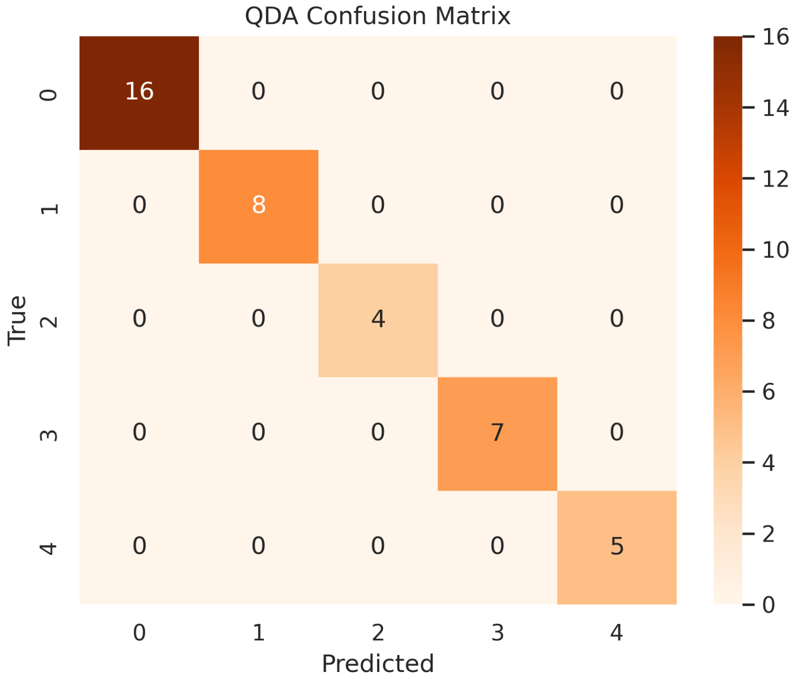

Figure 5

Confusion matrix for the QDA classifier on the test set, using K-Means cluster labels as ground truth.

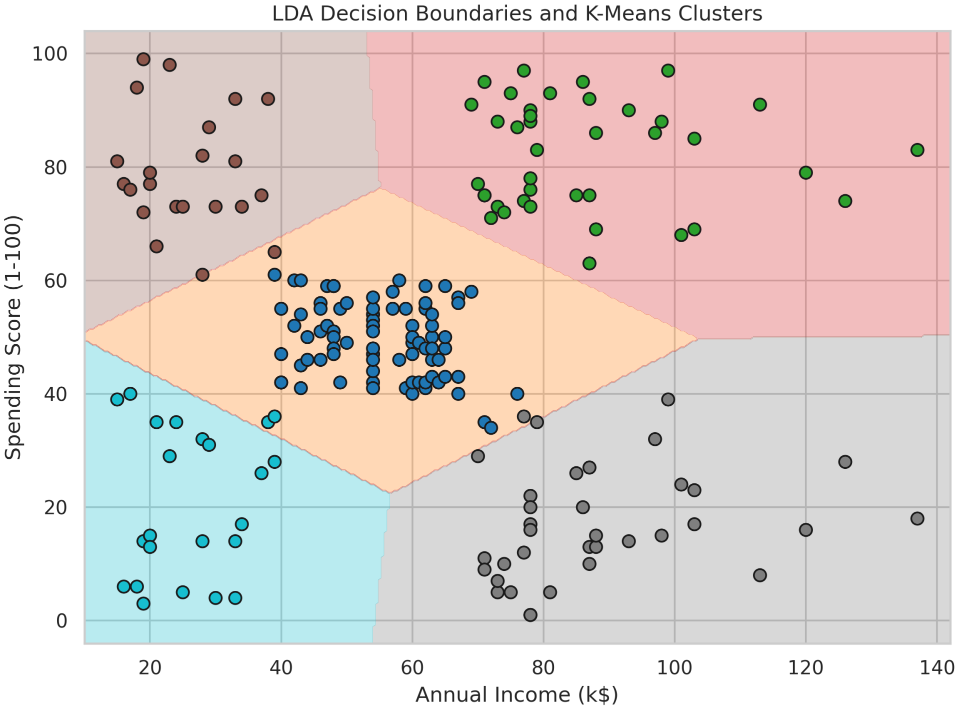

Figure 6

LDA decision boundaries in the annual income–spending score plane, with points colored by K-Means cluster.

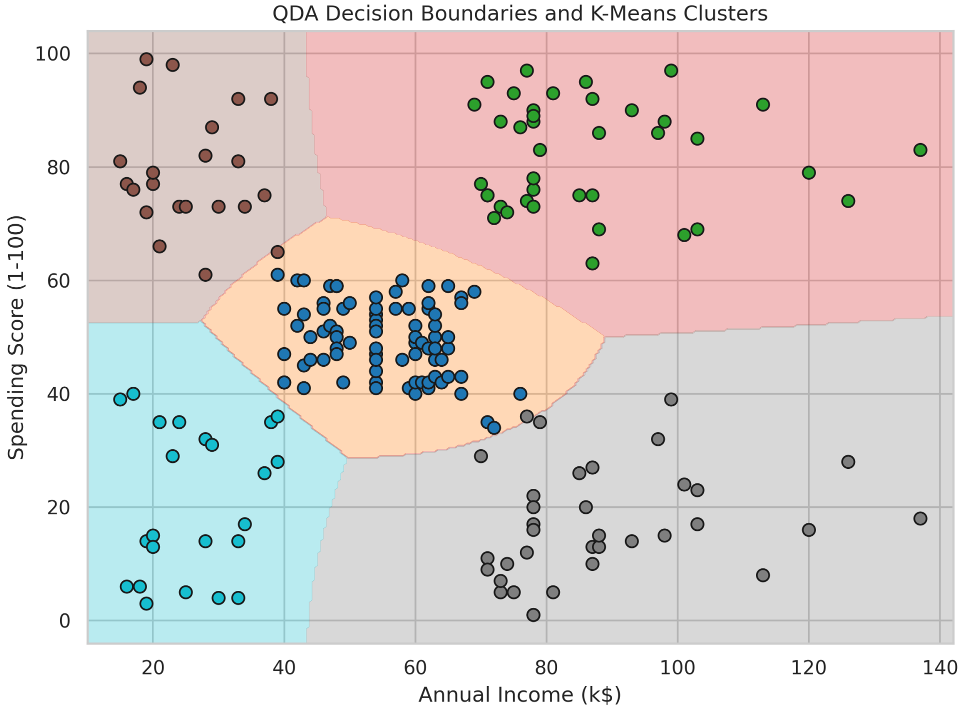

Figure 7

QDA decision boundaries in the annual income–spending score plane, with points colored by K-Means cluster.