Module 6 Summary: Unsupervised Learning

Data Science with Python – BSc Course

What We Covered

Module 6: Unsupervised Learning – Complete Journey

- **L29: K-Means Clustering** – Partitioning data into K groups by minimizing inertia
- **L30: Hierarchical Clustering** – Building dendrograms to discover nested cluster structures
- **L31: PCA** – Reducing dimensions while preserving variance
- **L32: ML Pipeline** – End-to-end workflows from preprocessing to deployment

You've mastered the core techniques for discovering structure in unlabeled data and building production-ready ML systems.

From exploration to production – complete unsupervised learning toolkit

L29 Recap: K-Means Clustering

Key Takeaways from K-Means Clustering:

- K-Means iteratively assigns points to K cluster centers by minimizing within-cluster variance (inertia)
- Choose K using elbow method (look for bend) + silhouette score (quality metric)
- Always standardize features before clustering – large-scale features dominate distance calculations
- Interpret centroids to understand cluster meaning (e.g., "Blue Chips" vs "Growth Stocks")

Finance Application: Stock segmentation by risk/return characteristics for portfolio diversification

K-Means = partition data into K spherical clusters. Elbow + silhouette guide K selection.

L30 Recap: Hierarchical Clustering

Key Takeaways from Hierarchical Clustering:

- Hierarchical clustering builds a tree (dendrogram) showing relationships at all scales
- Linkage method matters: Ward produces balanced clusters (recommended default)
- Cut dendrogram at desired height to obtain flat clusters – no need to specify K upfront
- Finance: correlation-based clustering for Hierarchical Risk Parity (HRP) portfolio construction

Advantage over K-Means: Reveals full hierarchy + works for non-spherical clusters

Dendrogram = tree diagram. Ward linkage = balanced. Cut horizontally to get clusters.

L31 Recap: PCA

Key Takeaways from PCA:

- PCA finds orthogonal (uncorrelated) directions of maximum variance in data
- Scree plot and cumulative variance guide component selection (95% rule common)
- Always standardize before PCA – large-scale features dominate without scaling
- Finance: PCA extracts statistical factors from returns (PC1 often = market factor)

Use Cases: Dimensionality reduction for visualization, feature engineering, noise filtering

PCA rotates to max variance axes. PC1 = most important direction. Standardize first!

L32 Recap: ML Pipeline

Key Takeaways from ML Pipeline:

- Pipelines chain preprocessing and modeling – ensures reproducible, leak-free workflows
- Cross-validation estimates generalization performance robustly
- GridSearchCV/RandomizedSearchCV for hyperparameter tuning with cross-validation
- TimeSeriesSplit for financial data – respects temporal order, prevents future leakage

Production: Save pipelines with joblib – single object captures entire workflow

Pipeline = chain steps. GridSearchCV = optimize. TimeSeriesSplit = respect time.

Key Concepts Mastered

Five Core Skills from This Module:

1. **Clustering** – K-Means and hierarchical methods for grouping unlabeled data
2. **Dimensionality Reduction** – PCA for visualization and feature engineering
3. **Cluster Validation** – Elbow method, silhouette score, dendrogram interpretation
4. **Pipeline Construction** – Chaining preprocessing, transformations, and models
5. **Time Series Handling** – TimeSeriesSplit for leak-free financial ML workflows

These techniques power segmentation, regime detection, and production ML systems

Finance Applications

How Unsupervised Learning Powers Financial Systems:

- **Stock Segmentation** – K-Means clusters stocks by risk/return for diversified portfolios
- **Hierarchical Risk Parity** – Correlation-based dendrograms allocate inversely to cluster variance
- **Factor Discovery** – PCA extracts statistical risk factors (market, sector, style)
- **Regime Detection** – Clustering identifies bull/bear/sideways markets without labels
- **Client Segmentation** – Robo-advisors group investors by behavior for personalized advice

Renaissance Technologies, Betterment, and risk managers use these techniques daily

Tools & Code Patterns

sklearn Toolkit for Unsupervised Learning:

- **K-Means:** `KMeans(n_clusters=K, random_state=42).fit_predict(X_scaled)`
- **Hierarchical:** `scipy.cluster.hierarchy.linkage(X, method='ward')` + `dendrogram()`
- **PCA:** `PCA(n_components=0.95).fit_transform(X_scaled)` – 95% variance
- **Pipeline:** `Pipeline([('scaler', StandardScaler()), ('pca', PCA()), ('clf', ...)])`
- **GridSearch:** `GridSearchCV(pipe, param_grid, cv=TimeSeriesSplit(5))`

Key Pattern: Always standardize before clustering/PCA – use StandardScaler first

sklearn makes complex ML workflows clean and reproducible

Critical Lessons

Three Non-Negotiable Rules:

1. **Standardize Features First**
   - K-Means and PCA are sensitive to scale
   - Large-scale features (market cap) dominate small-scale (P/E ratio)
   - Always use StandardScaler before clustering or PCA
2. **Prevent Data Leakage**
   - Fit transformers ONLY on training data
   - Use Pipelines to enforce this automatically
   - TimeSeriesSplit for temporal data – never leak future
3. **Validate Cluster Quality**
   - Elbow method shows diminishing returns
   - Silhouette score quantifies cluster separation
   - Domain knowledge confirms results make sense

These rules separate amateur analysis from production-ready systems

What's Next: Module 7

Module 7: Deep Learning (L33-L36)
Traditional ML has limits:

- K-Means requires you to specify K
- Linear models assume straight-line relationships
- PCA only captures linear combinations
- Decision trees struggle with high-dimensional interactions

Deep Learning breaks through these constraints:

- Neural networks learn complex, non-linear patterns
- Backpropagation optimizes millions of parameters
- From perceptrons to multi-layer networks
- Regularization techniques prevent overfitting

When traditional ML plateaus, deep learning breaks through the ceiling