Big Data in Finance

Academic Year 2025–2026

Programme:	MSc Finance (Elective)
Duration:	7 weeks
Format:	3-hour lectures + 1-hour tutorials
Assessment:	10% Participation, 25% Group Project, 65% Exam

Course Team

Module Lead: Dr. Daniele Bianchi
Email: d.bianchi@qmul.ac.uk
Website: `whitesphd.com`
Office hours: by appointment

1 Course Overview

Core objective: Apply state-of-the-art machine learning methods to financial decision-making.

This module provides a comprehensive and rigorous introduction to machine learning methods for finance, with a systematic focus on both theoretical foundations and practical implementation. The module is structured around two main applications:

- **Market timing:** Time-series prediction of aggregate market returns (with factor investing as a related cross-sectional application)
- **Credit risk assessment:** Default prediction and credit scoring

Key features: Theory + Practice, Python implementation, production deployment considerations, AI-assisted learning, real financial datasets.

1.1 Lecture Structure

Each **3-hour week** is divided into two parts:

- **First 1.5 hours:** Theoretical foundations, methodological insights, algorithmic procedures, mathematical concepts
- **Second 1.5 hours:** Financial applications, case studies, implementation details, best practices

This bipartite structure ensures students develop both conceptual understanding and practical competence with industry relevance.

1.2 Tutorial Structure

1-hour hands-on sessions focusing on implementation: guided coding exercises, production-ready code practices, clean/documented/version-controlled code, and model deployment considerations.

AI-Assisted Learning: Students are *encouraged* to leverage AI tools (ChatGPT, Claude, etc.) for debugging code, code optimization, interpreting model outputs, and understanding error messages. **Important:** You must understand and explain all code and implementations, regardless of source.

1.3 Learning Outcomes

By the end of this module, you will be able to:

1. Describe complex models relevant to financial decision making
2. Compute complex models to predict default in corporate and retail credit markets
3. Appraise a range of quantitative techniques to build trading strategies based on stock characteristics
4. Produce deployment-ready pipelines for investment and credit risk management

1.4 Prerequisites

- Basic Python programming
- Statistics and probability theory
- Linear algebra fundamentals
- Financial markets knowledge

2 Module Structure

Week 1: Foundations of Machine Learning

Topics: Supervised vs. unsupervised learning, loss functions and empirical risk minimization, bias-variance decomposition, model selection (training/validation/test sets), cross-validation for time-series, multiple testing and data snooping in finance.

Readings:

- James et al. (2023) *ISL* – Chapters 2, 5
- Harvey, Liu & Zhu (2016) "...and the Cross-Section of Expected Returns" *RFS*
  - Gu, Kelly & Xiu (2020) "Empirical Asset Pricing via ML" *RFS* – Intro & Sec. 2

Tutorial 1: Environment setup, data pipeline construction, time-series train-test splits

Weeks 2–3: Regression Methods

Topics: OLS limitations in high dimensions; Ridge, Lasso, and Elastic Net regularization; tree-based methods (CART, Random Forests, Gradient Boosting); XGBoost/LightGBM; hyperparameter tuning via cross-validation.

Application: Factor investing and market timing using macroeconomic predictors and firm characteristics.

Readings:

- James et al. (2023) *ISL* – Chapters 3, 6, 8
- Goyal, Welch & Zafirov (2024) "A Comprehensive 2022 Look at Equity Premium Prediction" *RFS*
  - Kozak, Nagel & Santosh (2020) "Shrinking the Cross-Section" *JFE*
  - Breiman (2001) "Random Forests" *Machine Learning*

Tutorials 2–3: Regularized regression pipeline, Random Forest and XGBoost implementation, feature importance analysis

Week 4: Classification Methods

Topics: Logistic regression; Linear and Quadratic Discriminant Analysis; classification trees and ensembles; evaluation metrics (accuracy, precision, recall, F1, ROC, AUC); class imbalance handling (SMOTE, class weights); probability calibration; threshold optimization.

Application: Credit risk assessment using LendingClub data—default prediction, credit scoring, regulatory considerations.

Readings:

- James et al. (2023) *ISL* – Chapters 4, 8
- Shumway (2001) "Forecasting Bankruptcy More Accurately" *Journal of Business*
- Khandani, Kim & Lo (2010) "Consumer Credit-Risk Models via ML" *JBF*
  - Campbell, Hilscher & Szilagyi (2008) "In Search of Distress Risk" *JF*

Tutorial 4: Logistic regression and XGBoost for credit scoring, class imbalance handling, ROC analysis

Weeks 5–7: Advanced Topics

Week 5: Unsupervised Learning

- Principal Component Analysis, factor extraction, Instrumented PCA
- Clustering methods (K-means, hierarchical)
- Applications: dimensionality reduction, portfolio construction, regime detection

Week 6: Model Interpretability

- SHAP values, LIME, partial dependence plots
- Walk-forward validation and backtesting
- Model monitoring and regulatory compliance (SR 11-7)

Week 7: Neural Networks

- Multi-layer perceptron architecture, activation functions
- Backpropagation, regularization (dropout, early stopping)
- Applications: return prediction and credit risk comparison

Readings:

- James et al. (2023) *ISL* – Chapters 10, 12
- Kelly, Pruitt & Su (2019) "Characteristics are Covariances" *JFE*
- Lundberg & Lee (2017) "A Unified Approach to Interpreting Model Predictions" *NeurIPS*
  - Goodfellow, Bengio & Courville (2016) *Deep Learning* – Chapters 6–7

3 Assessment

3.1 Participation (10%)

- Active engagement in lectures and tutorials
- Quality of questions and contributions

3.2 Group Project (25%)

Teams of 4–5 students. Choose one option:

- **Option A: ML Factor Investing Strategy**
  - Implement 3+ ML methods for return prediction
  - Compare to benchmark models
  - Portfolio construction and backtesting
- **Option B: Credit Risk Model**
  - Build classification pipeline with 3+ methods
  - Address class imbalance rigorously
  - Model interpretation and validation

Deliverables: Jupyter notebook, written report (5 pages max), presentation (10 min including Q&A).

Grading: Implementation quality (40%), Interpretation and analysis (40%), Deployment plan (20%).

3.3 Final Exam (65%)

3-hour written exam with three sections:

Section A (30 points): Conceptual Understanding—when to use which method and why, interpreting model outputs, identifying methodological errors, deployment considerations.

Section B (20 points): Applied Problems—interpreting scikit-learn outputs, short coding exercises (fill-in-the-blank), debugging exercises, hyperparameter selection with justification.

Section C (15 points): Case Study Analysis—critical evaluation of ML study in finance, identify methodological issues and suggest improvements.

4 Core Textbooks

Recommended (all available free online):

1. **James, Witten, Hastie, Tibshirani & Taylor (2023)** *An Introduction to Statistical Learning with Applications in Python*
   Primary textbook, accessible treatment. https://www.statlearning.com/
2. **Hastie, Tibshirani & Friedman (2009)** *Elements of Statistical Learning*
   More advanced theoretical reference. https://hastie.su.domains/ElemStatLearn/
3. **Goodfellow, Bengio & Courville (2016)** *Deep Learning*
   Neural networks reference. https://www.deeplearningbook.org/

Optional (purchase):

- **López de Prado (2018)** *Advances in Financial Machine Learning*—Practical financial ML focus.

4.1 Reading Strategy

Each lecture has a structured reading list:

- **Recommended readings (∼2 per lecture):** Complement lecture slides and notes; discussed in class; important background.
- **Advanced readings (1–2 per lecture):** For research-interested students; technical proofs and extensions; reference for assignments.

Readings are organized thematically, not strictly by week.

5 Software and Tools

5.1 Required

- Python 3.8+
- Core libraries: numpy, pandas, scikit-learn, matplotlib, seaborn
- ML libraries: XGBoost, SHAP
- IDE: VS Code, Spyder, or PyCharm

5.2 Recommended

- Environment: Anaconda/Miniconda
- Version control (optional): GitHub/GitLab

5.3 Datasets

- Return prediction: Goyal and Welch (provided/provided)
- Credit risk: LendingClub loan data (public/provided)

6 Course Policies

6.1 Academic Integrity

- All individual work must be your own
- **AI tools:** May be used for learning, but you must understand all the material. Domain expertise is key to boosting productivity via AI tools. Exam tests independent understanding.
- **Group project:** Collaboration within groups only
- **Proper citations** required for all sources
- Plagiarism will result in course failure

6.2 Collaboration Policy

- Tutorials: Collaboration encouraged
- Group project: Within groups only
- Exam: Individual work only

6.3 Late Submissions

- Group projects: 10% penalty per day late
- Extensions: Only for documented emergencies—contact the Programme Office as soon as possible
- All team members share responsibility for timely submission

7 What Makes This Module Different

1. **Production focus:** Not just building models, but deploying them—model monitoring and drift detection, documentation and versioning, deployment considerations throughout.
2. **Financial context:** Finance-specific challenges—time-series validation (avoid look-ahead bias), overfitting dangers in finance, regulatory requirements.
3. **Interpretability emphasis:** Black-box models are not enough—SHAP values, LIME, partial dependence plots; explaining predictions to stakeholders.
4. **AI-assisted learning:** Reflecting modern development practices—learn to use AI tools effectively, but maintain deep understanding.

Last Updated: June 28, 2026
Version 2.0