ĐỀ CƯƠNG ÔN TẬP

# HỌC MÁY

Dành cho sinh viên đại học khối kinh tế, tài chính, công nghệ tài chính

*Machine Learning – a bilingual study guide*

Hồi quy · Phân lớp · Kiểm định chéo · Lasso · Cây quyết định
Rừng ngẫu nhiên · SVM · PCA · Phân cụm · Ứng dụng tài chính

Bài tập có lời giải · Tự luận có dàn ý · Trắc nghiệm có đáp án · Mã Python chạy thật

Song ngữ Việt – Anh · Vietnamese – English

**Phiên bản 1.0 – Tháng 11 năm 2026**
https://tailieuso.vin

# MỤC LỤC

*CONTENTS*

**Chương 1. Học máy là gì: tổng quan và quy trình – *Chapter 1. What machine learning is: an overview and workflow*** ..... **6**

1.1. Học máy, học thống kê và trí tuệ nhân tạo – *1.1. Machine learning, statistical learning and artificial intelligence* ..... 6

1.2. Mô hình Y = f(X) + ε: ta ước lượng cái gì – *1.2. The model Y = f(X) + ε: what we estimate* ..... 7

1.3. Vì sao ước lượng f: dự báo và suy diễn – *1.3. Why estimate f: prediction and inference* ..... 9

1.4. Ước lượng f: phương pháp có tham số và phi tham số – *1.4. Estimating f: parametric and non-parametric methods* ..... 11

1.5. Đánh đổi giữa độ chính xác và khả năng diễn giải – *1.5. The trade-off between accuracy and interpretability* ..... 15

1.6. Học có giám sát, không giám sát và bán giám sát – *1.6. Supervised, unsupervised and semi-supervised learning* ..... 16

1.7. Bài toán hồi quy và bài toán phân lớp – *1.7. Regression and classification problems* 18

1.8. Dữ liệu trong tài chính – *1.8. Data in finance* ..... 19

1.9. Quy trình một dự án học máy – *1.9. The workflow of a machine learning project* ..... 20

1.10. Ứng dụng tài chính, thiên lệch dữ liệu và nghề nghiệp – *1.10. Finance applications, data bias and careers* ..... 22

1.11. Tính bằng Python – *1.11. Working in Python* ..... 24

1.12. Bài tập vận dụng có lời giải – *Applied exercises with solutions* ..... 32

1.13. Câu hỏi tự luận và dàn ý trả lời – *Essay questions and answer outlines* ..... 34

1.14. Câu hỏi trắc nghiệm – *Multiple-choice questions* ..... 35

**Chương 2. Đánh giá độ chính xác của mô hình – *Chapter 2. Assessing model accuracy*** ..... **38**

2.1. Vì sao phải đo độ chính xác của mô hình – *2.1. Why we measure a model's accuracy* ..... 38

2.2. Sai số bình phương trung bình: huấn luyện và kiểm tra – *2.2. Mean squared error: training and test* ..... 40

2.3. Quá khớp và chưa khớp: hình chữ U – *2.3. Overfitting and underfitting: the U shape* ..... 42

2.4. Đánh đổi độ chệch – phương sai – *2.4. The bias–variance trade-off* ..... 44

2.5. Phân lớp: tỷ lệ lỗi và bộ phân lớp Bayes – *2.5. Classification: the error rate and the Bayes classifier* ..... 47

2.6. KNN: phân lớp bằng k láng giềng gần nhất – *2.6. KNN: classifying with the k nearest neighbours* ..... 49

2.7. Ma trận nhầm lẫn và các chỉ số phân lớp – *2.7. The confusion matrix and classification measures* ..... 52

2.8. Đường ROC và AUC – *2.8. The ROC curve and the AUC* ..... 55

2.9. Dữ liệu mất cân bằng và mô hình cơ sở – *2.9. Imbalanced data and the baseline model* ..... 58
2.10. Ứng dụng: chọn ngưỡng duyệt vay theo chi phí – *2.10. Application: choosing a lending threshold by cost* ..... 59
2.11. Tính bằng Python – *2.11. Computing with Python* ..... 61
2.12. Bài tập vận dụng có lời giải – *Applied exercises with solutions* ..... 69
2.13. Câu hỏi tự luận và dàn ý trả lời – *Essay questions and answer outlines* ..... 71
2.14. Câu hỏi trắc nghiệm – *Multiple-choice questions* ..... 73

**Chương 3. Hồi quy tuyến tính – *Chapter 3. Linear regression* ..... 75**
3.1. Hồi quy đơn: đường thẳng khớp nhất với dữ liệu – *3.1. Simple regression: the straight line that fits best* ..... 75
3.2. Hệ số và mô hình chính xác đến đâu – *3.2. How accurate are the coefficients and the model* ..... 78
3.3. Hồi quy bội: nhiều biến giải thích cùng lúc – *3.3. Multiple regression: several predictors at once* ..... 81
3.4. Bốn câu hỏi về mô hình hồi quy bội – *3.4. Four questions about a multiple regression* ..... 84
3.5. Biến định tính: biến giả – *3.5. Qualitative predictors: dummy variables* ..... 86
3.6. Mở rộng mô hình: tương tác và đa thức – *3.6. Extending the model: interactions and polynomials* ..... 88
3.7. Sáu vấn đề thường gặp – *3.7. Six common problems* ..... 90
3.8. Ứng dụng: kế hoạch tiếp thị và hạn mức tín dụng – *3.8. Application: the marketing plan and credit limits* ..... 93
3.9. So sánh hồi quy tuyến tính với KNN hồi quy – *3.9. Linear regression compared with KNN regression* ..... 95
3.10. Tính bằng Python – *3.10. Doing it in Python* ..... 97
3.11. Bài tập vận dụng có lời giải – *Applied exercises with solutions* ..... 106
3.12. Câu hỏi tự luận và dàn ý trả lời – *Essay questions and answer outlines* ..... 109
3.13. Câu hỏi trắc nghiệm – *Multiple-choice questions* ..... 110

**Chương 4. Phân lớp – *Chapter 4. Classification* ..... 112**
4.1. Phân lớp là gì và dùng vào việc gì – *4.1. What classification is and what it is used for* ..... 112
4.2. Vì sao không dùng hồi quy tuyến tính – *4.2. Why not use linear regression* ..... 114
4.3. Hồi quy logistic – *4.3. Logistic regression* ..... 116
4.4. Định lý Bayes cho phân lớp – *4.4. Bayes' theorem for classification* ..... 122
4.5. Phân tích biệt thức tuyến tính (LDA) – *4.5. Linear discriminant analysis (LDA)* ..... 123
4.6. Phân tích biệt thức bậc hai (QDA) – *4.6. Quadratic discriminant analysis (QDA)* ..... 126
4.7. So sánh logistic, LDA, QDA và KNN – *4.7. Comparing logistic, LDA, QDA and KNN* ..... 128
4.8. Chấm điểm tín dụng ở Lúa Vàng: từ xác suất đến quyết định – *4.8. Credit scoring at Lua Vang: from probability to decision* ..... 130
4.9. Tính bằng Python – *4.9. Computing with Python* ..... 133
4.10. Bài tập vận dụng có lời giải – *Applied exercises with solutions* ..... 140

4.11. Câu hỏi tự luận và dàn ý trả lời – *Essay questions and answer outlines* ..... 142
4.12. Câu hỏi trắc nghiệm – *Multiple-choice questions* ..... 143

**Chương 5. Phương pháp lấy mẫu lại – *Chapter 5. Resampling methods*..... 146**
5.1. Vì sao cần lấy mẫu lại – *5.1. Why resample* ..... 146
5.2. Tập kiểm định – *5.2. The validation set* ..... 148
5.3. Kiểm định chéo bỏ một (LOOCV) – *5.3. Leave-one-out cross-validation (LOOCV)* ..... 152
5.4. Kiểm định chéo k phần (k-fold) – *5.4. k-fold cross-validation* ..... 155
5.5. Đánh đổi độ chệch và phương sai của k-fold – *5.5. The bias–variance trade-off of k-fold* ..... 158
5.6. Kiểm định chéo cho bài toán phân lớp – *5.6. Cross-validation for classification* ..... 160
5.7. Bootstrap – *5.7. The bootstrap* ..... 164
5.8. Chuỗi thời gian tài chính: không xáo trộn – *5.8. Financial time series: no shuffling* ..... 169
5.9. Ứng dụng: kiểm tra mô hình chấm điểm tín dụng của Lúa Vàng – *5.9. Application: checking a credit-scoring model at Lua Vang* ..... 171
5.10. Tính bằng Python – *5.10. Computing with Python* ..... 173
5.11. Bài tập vận dụng có lời giải – *Applied exercises with solutions* ..... 176
5.12. Câu hỏi tự luận và dàn ý trả lời – *Essay questions and answer outlines* ..... 178
5.13. Câu hỏi trắc nghiệm – *Multiple-choice questions* ..... 180

**Chương 6. Chọn mô hình và chính quy hóa – *Chapter 6. Model selection and regularisation*..... 182**
6.1. Vì sao cần bớt biến – *6.1. Why drop variables* ..... 182
6.2. Chọn tập con biến: thử hết và thêm – bớt từng biến – *6.2. Subset selection: try everything, or add and remove one at a time* ..... 184
6.3. Chọn mô hình tối ưu: các tiêu chí đo sai số trên dữ liệu mới – *6.3. Choosing the best model: criteria that estimate error on new data* ..... 189
6.4. Hồi quy ridge: kéo hệ số về gần 0 – *6.4. Ridge regression: pulling coefficients towards 0* ..... 192
6.5. Lasso: kéo hệ số về đúng 0 – *6.5. The lasso: pulling coefficients to exactly 0* ..... 195
6.6. Chọn λ bằng kiểm định chéo – *6.6. Choosing λ by cross-validation* ..... 198
6.7. Giảm chiều: gộp biến rồi hồi quy – *6.7. Dimension reduction: merge variables, then regress* ..... 200
6.8. Dữ liệu nhiều chiều: khi số biến lớn hơn số quan sát – *6.8. High-dimensional data: when variables outnumber observations* ..... 203
6.9. Ứng dụng: chọn biến cho mô hình hạn mức tại Lúa Vàng – *6.9. Application: choosing variables for the credit limit model at Lua Vang* ..... 205
6.10. Tính bằng Python – *6.10. Doing it in Python* ..... 207
6.11. Bài tập vận dụng có lời giải – *Applied exercises with solutions* ..... 215
6.12. Câu hỏi tự luận và dàn ý trả lời – *Essay questions and answer outlines* ..... 217
6.13. Câu hỏi trắc nghiệm – *Multiple-choice questions* ..... 219

**Chương 7. Vượt ra ngoài tuyến tính – *Chapter 7. Moving beyond linearity* ..... 221**

7.1. Vì sao cần vượt ra ngoài đường thẳng – *7.1. Why go beyond a straight line* ..... 221
7.2. Hồi quy đa thức – *7.2. Polynomial regression* ..... 223
7.3. Hàm bậc thang – *7.3. Step functions* ..... 226
7.4. Hàm cơ sở: khung chung – *7.4. Basis functions: one common frame* ..... 228
7.5. Spline hồi quy – *7.5. Regression splines* ..... 229
7.6. Spline làm trơn – *7.6. Smoothing splines* ..... 233
7.7. Hồi quy cục bộ – *7.7. Local regression* ..... 237
7.8. Mô hình cộng tổng quát (GAM) – *7.8. Generalised additive models (GAM)* ..... 239
7.9. Ứng dụng tài chính – ngân hàng: thu nhập và xác suất vỡ nợ – *7.9. Banking and finance application: income and default probability* ..... 243
7.10. Tính bằng Python – *7.10. Computing with Python* ..... 244
7.11. Bài tập vận dụng có lời giải – *Applied exercises with solutions* ..... 253
7.12. Câu hỏi tự luận và dàn ý trả lời – *Essay questions and answer outlines* ..... 255
7.13. Câu hỏi trắc nghiệm – *Multiple-choice questions* ..... 256

**Chương 8. Cây quyết định và phương pháp tổ hợp – *Chapter 8. Decision trees and ensemble methods* ..... 259**
8.1. Cây quyết định: trò hai mươi câu hỏi với dữ liệu – *8.1. Decision trees: twenty questions played with data* ..... 259
8.2. Cây hồi quy: chia dữ liệu thành các vùng – *8.2. Regression trees: splitting the data into regions* ..... 260
8.3. Tỉa cây: cây to quá thì học thuộc đề cũ – *8.3. Pruning: a tree that is too big learns the old exam by heart* ..... 264
8.4. Cây phân lớp: lá ghi nhãn, nút đo độ thuần – *8.4. Classification trees: leaves hold labels, nodes measure purity* ..... 267
8.5. Cây và mô hình tuyến tính: ai thắng khi nào – *8.5. Trees versus linear models: who wins when* ..... 270
8.6. Bagging: hỏi nhiều cây rồi lấy trung bình – *8.6. Bagging: ask many trees and average* ..... 272
8.7. Rừng ngẫu nhiên: cả lớp bỏ phiếu, mỗi bạn chỉ nhìn một phần đề – *8.7. Random forests: the whole class votes, each pupil seeing only part of the question* ..... 274
8.8. Boosting: mỗi cây nhỏ sửa lỗi cho các cây trước – *8.8. Boosting: each small tree corrects the earlier trees* ..... 276
8.9. Ứng dụng: phát hiện gian lận thẻ ở Lúa Vàng – *8.9. Application: card fraud detection at Lúa Vàng* ..... 279
8.10. Tính bằng Python – *8.10. Computing with Python* ..... 281
8.11. Bài tập vận dụng có lời giải – *Applied exercises with solutions* ..... 289
8.12. Câu hỏi tự luận và dàn ý trả lời – *Essay questions and answer outlines* ..... 291
8.13. Câu hỏi trắc nghiệm – *Multiple-choice questions* ..... 293

**Chương 9. Máy vectơ hỗ trợ – *Chapter 9. Support vector machines* ..... 295**
9.1. Siêu phẳng: kẻ một đường giữa hai nhóm hồ sơ – *9.1. Hyperplanes: drawing a line between two groups of records* ..... 295
9.2. Bộ phân lớp lề cực đại – *9.2. The maximal margin classifier* ..... 298
9.3. Khi hai nhóm không tách được – *9.3. When the two groups cannot be separated* ..... 302

9.4. Bộ phân lớp vectơ hỗ trợ: biến lỏng và tham số C – *9.4. The support vector classifier: slack variables and the parameter C* ..... 302
9.5. Máy vectơ hỗ trợ: uốn ranh giới bằng hàm nhân – *9.5. The support vector machine: bending the boundary with kernels* ..... 305
9.6. SVM cho nhiều hơn hai lớp – *9.6. SVM with more than two classes* ..... 310
9.7. Quan hệ với hồi quy logistic và ROC của SVM – *9.7. Relation to logistic regression, and the ROC of an SVM* ..... 311
9.8. Ứng dụng tài chính – ngân hàng tại Lúa Vàng – *9.8. Application in banking and finance at Lua Vang* ..... 314
9.9. Tính bằng Python – *9.9. Computing with Python* ..... 316
9.10. Bài tập vận dụng có lời giải – *Applied exercises with solutions* ..... 322
9.11. Câu hỏi tự luận và dàn ý trả lời – *Essay questions and answer outlines* ..... 324
9.12. Câu hỏi trắc nghiệm – *Multiple-choice questions* ..... 325

**Chương 10. Học không giám sát – *Chapter 10. Unsupervised learning* ..... 327**
10.1. Thách thức của học không giám sát – *10.1. The challenge of unsupervised learning* ..... 327
10.2. Phân tích thành phần chính: trục biến thiên lớn nhất – *10.2. Principal components analysis: the axis of greatest variation* ..... 329
10.3. Cách hiểu thứ hai, chuẩn hóa và tính duy nhất – *10.3. A second view, standardising and uniqueness* ..... 332
10.4. Tỷ lệ phương sai giải thích, đồ thị scree và công dụng khác – *10.4. Proportion of variance explained, the scree plot and other uses* ..... 333
10.5. K-means: chia khách thành k cụm – *10.5. K-means: splitting customers into k clusters* ..... 335
10.6. Phân cụm thứ bậc: cây các lần gộp – *10.6. Hierarchical clustering: a tree of merges* ..... 338
10.7. Vấn đề thực hành khi phân cụm – *10.7. Practical issues in clustering* ..... 343
10.8. Ứng dụng: phân khúc khách hàng và phát hiện bất thường tại Lúa Vàng – *10.8. Application: customer segmentation and anomaly detection at Lua Vang* ..... 344
10.9. Tính bằng Python – *10.9. Doing it in Python* ..... 347
10.10. Bài tập vận dụng có lời giải – *Applied exercises with solutions* ..... 355
10.11. Câu hỏi tự luận và dàn ý trả lời – *Essay questions and answer outlines* ..... 357
10.12. Câu hỏi trắc nghiệm – *Multiple-choice questions* ..... 359

**Phụ lục A. Đáp án câu hỏi trắc nghiệm – *Appendix A. Answer key* ..... 362**

**Phụ lục B. Công thức – *Appendix B. Formulas* ..... 367**

**Phụ lục C. Thuật ngữ Việt – Anh – *Appendix C. Vietnamese–English glossary* ..... 374**

**Phụ lục D. Tài liệu tham khảo – *Appendix D. References* ..... 381**

# Chương 1. Học máy là gì: tổng quan và quy trình – *Chapter 1. What machine learning is: an overview and workflow*

**Mục tiêu chương** – *Chapter objectives*

- Phân biệt học máy, học thống kê, trí tuệ nhân tạo và nói được mỗi thứ nằm ở đâu.
  *Tell machine learning, statistical learning and artificial intelligence apart and say where each sits.*
- Viết được Y = f(X) + ε và chỉ ra phần sai số nào giảm được, phần nào không.
  *Write Y = f(X) + ε and say which part of the error can be reduced and which cannot.*
- Chọn đúng giữa dự báo và suy diễn, giữa phương pháp có tham số và phi tham số, giữa hồi quy và phân lớp.
  *Choose correctly between prediction and inference, parametric and non-parametric methods, regression and classification.*
- Kể đủ các bước của một dự án học máy, từ đặt bài toán đến theo dõi sau triển khai.
  *List every step of a machine learning project, from framing the problem to monitoring after deployment.*
- Đọc tệp CSV, tính thống kê mô tả, lập bảng tần số, tóm tắt theo nhóm và chia tập huấn luyện – kiểm tra bằng Python.
  *Read a CSV file, compute descriptive statistics, build frequency tables, summarise by group and split training and test sets in Python.*

## 1.1. Học máy, học thống kê và trí tuệ nhân tạo – *1.1. Machine learning, statistical learning and artificial intelligence*

Ngày đầu thực tập ở Công ty tài chính tiêu dùng Lúa Vàng, Hà My nghe anh Quân hỏi: "Muốn dạy một bạn mới phân biệt hồ sơ vay tốt và xấu, em chọn cách nào? Đưa bạn ấy cuốn quy tắc để học thuộc, hay đưa 10.000 hồ sơ cũ có ghi kết quả để bạn tự rút ra quy tắc?" Cách thứ hai là **học máy** (machine learning, viết tắt ML). Ta đưa cho máy nhiều ví dụ có kết quả, và máy tự tìm quy tắc nối thông tin trong hồ sơ với kết quả.

*On her first day as an intern at Lua Vang Consumer Finance, Ha My heard Quan ask: "To teach a newcomer to tell good loan files from bad ones, which way would you pick? Hand over a rulebook to memorise, or hand over 10,000 past files with their outcomes and let the newcomer work out the rules?" The second way is **machine learning** (ML). You give the computer many examples with results, and it finds the rule linking the information in a file to the outcome.*

Hai cái tên khác hay đi cùng học máy. **Học thống kê** (statistical learning) cũng cho máy học từ dữ liệu, nhưng nhìn từ phía thống kê: chú trọng đo độ tin cậy của kết quả và giải thích được mô hình. **Trí tuệ nhân tạo** (artificial intelligence, viết tắt AI) là chiếc ô lớn nhất. Nó gồm mọi cách làm cho máy hành xử thông minh, kể cả bộ quy tắc do người viết sẵn. Học máy là một nhánh nằm dưới chiếc ô đó.

*Two other names often travel with machine learning. **Statistical learning** also lets a computer learn from data, but it looks from the statistics side: it stresses measuring how far to trust a result and being able to explain the model. **Artificial intelligence** (AI) is the biggest umbrella. It*

*covers every way of making a machine act intelligently, including rule sets written by people. Machine learning is one branch under that umbrella.*

| **Tiêu chí**<br>*Criterion* | **Học thống kê**<br>*Statistical learning* | **Học máy**<br>*Machine learning* | **Trí tuệ nhân tạo**<br>*Artificial intelligence* |
|---|---|---|---|
| Câu hỏi chính<br>*Main question* | Mối liên hệ này có đáng tin không?<br>*Can this relationship be trusted?* | Máy dự báo đúng đến đâu?<br>*How well does the machine predict?* | Máy làm được việc cần trí khôn không?<br>*Can a machine do a task that needs intelligence?* |
| Điều quan tâm nhất<br>*What matters most* | Sai số, khoảng tin cậy<br>*Error and confidence intervals* | Độ chính xác trên dữ liệu mới<br>*Accuracy on new data* | Kết quả của cả hệ thống<br>*The result of the whole system* |
| Ví dụ ở Lúa Vàng<br>*Example at Lua Vang* | Mỗi lần trễ hạn thêm làm xác suất vỡ nợ đổi bao nhiêu<br>*How much each extra late payment changes the default probability* | Chấm điểm hồ sơ mới từ 10.000 hồ sơ cũ<br>*Scoring a new file from 10,000 old ones* | Trợ lý ảo trả lời khách, hoặc quy tắc "trễ từ 3 lần thì từ chối"<br>*A virtual assistant answering customers, or the rule "reject after 3 late payments"* |

Đọc bảng: ba cột chồng lên nhau. Hồi quy logistic (thuật toán dự báo xác suất, học ở Chương 4) xuất hiện ở cả ba. Nhà thống kê dùng nó để giải thích, kỹ sư học máy dùng nó để chấm điểm, và hệ chấm điểm tự động là một hệ AI. Quy tắc "trễ từ 3 lần thì từ chối" là AI nhưng không phải học máy, vì không ai để máy tự rút quy tắc.

*Reading the table: the three columns overlap. Logistic regression (an algorithm that predicts probabilities, met in Chapter 4) appears in all three. A statistician uses it to explain, a machine learning engineer uses it to score, and an automatic scoring system is an AI system. The rule "reject after 3 late payments" is AI but not machine learning, because nobody lets the machine find the rule.*

## 1.2. Mô hình Y = f(X) + ε: ta ước lượng cái gì – *1.2. The model Y = f(X) + ε: what we estimate*

Chị Linh hỏi Hà My: "Tháng sau tăng chi quảng cáo thêm 2 triệu đồng thì có thêm bao nhiêu khách mới?" Để trả lời, Hà My phải nói được quảng cáo liên quan thế nào đến số khách. Hai đại lượng này có tên riêng. Thứ ta biết và dùng để dự báo (chi quảng cáo) gọi là **biến đầu vào** (predictor, còn gọi là input hay feature), ký hiệu X. Thứ ta muốn dự báo (số khách mới) gọi là **biến đầu ra** (response), ký hiệu Y.

*Ms Linh asks Ha My: "If next month we raise advertising spend by 2 million dong, how many extra new customers will we get?" To answer, Ha My must say how advertising relates to customers. The two quantities have names. What we know and use to predict (advertising spend) is the **predictor** (also called input or feature), written X. What we want to predict (new customers) is the **response**, written Y.*

Giả sử có một quy luật thật nối hai bên, ta gọi nó là hàm f. Ta không bao giờ thấy f trực tiếp. Ngoài quảng cáo còn vô số thứ khác làm số khách đổi: thời tiết, khuyến mãi của đối

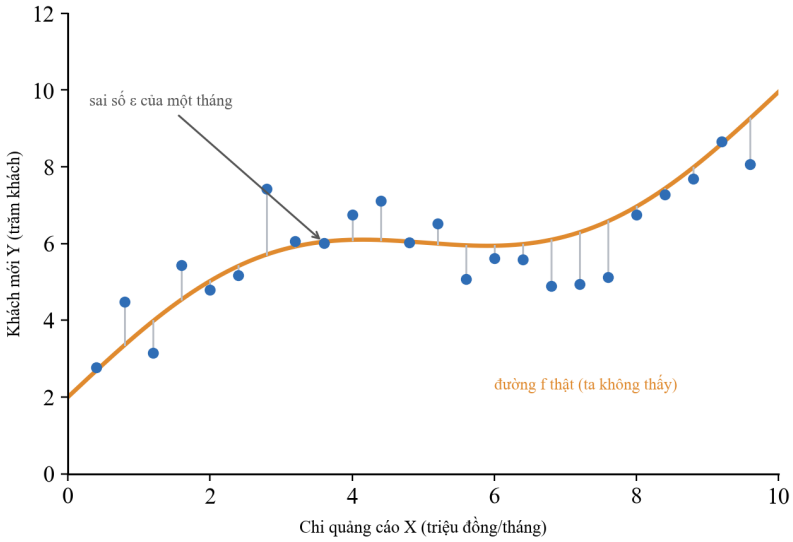
thủ, lỗi ghi sổ. Gom tất cả vào một số ngẫu nhiên ε (đọc là "epsilon"), gọi là **sai số** (error term). Ta viết:

*Suppose a true rule links the two; call it the function f. We never see f directly. Besides advertising, countless other things move the customer count: the weather, a rival's promotion, bookkeeping slips. Gather them into one random number ε (read "epsilon"), called the **error term**. We write:*

$$Y = f(X) + \varepsilon \tag{1.1}$$

Đọc công thức: số khách mới bằng phần quảng cáo giải thích được, cộng một phần ngẫu nhiên còn lại. Thường ta giả định ε có trung bình bằng 0: có tháng khách nhiều hơn dự kiến, có tháng ít hơn, không lệch về bên nào.

*Reading the formula: new customers equal the part advertising explains plus a random remainder. We usually assume ε averages 0: some months bring more customers than expected, some fewer, with no lean to either side.*




**Hình 1.1.** Đường cam là hàm f thật, ta không nhìn thấy nó. Mỗi chấm xanh là một tháng, nằm trên hoặc dưới đường cam một đoạn ε (nét xám). Dù biết đúng đường cam, ta vẫn lệch vì các đoạn xám đó.

***Figure 1.1.** The orange line is the true function f, which we cannot see. Each blue dot is a month, sitting above or below the orange line by an amount ε (grey stroke). Even knowing the orange line exactly, we still miss because of those grey strokes.*

Khi có dữ liệu, ta dựng một ước lượng của f, ký hiệu $\hat{f}$ (đọc "f mũ"), rồi dự báo $\hat{Y} = \hat{f}(X)$. Dự báo lệch khỏi Y vì hai lý do khác nhau:

*With data in hand, we build an estimate of f, written $\hat{f}$ (read "f hat"), and predict $\hat{Y} = \hat{f}(X)$. The prediction misses Y for two different reasons:*

- **Sai số giảm được** (reducible error): $\hat{f}$ chưa giống f. Ta giảm nó bằng nhiều dữ liệu hơn hoặc phương pháp tốt hơn.
  ***Reducible error**: $\hat{f}$ does not yet resemble f. More data or a better method shrinks it.*
- **Sai số không giảm được** (irreducible error): chính là ε. Có $\hat{f}$ giống hệt f thì ε vẫn còn, vì ε chứa cả những thứ ta không đo được.

***Irreducible error**: this is ε itself. Even with $\hat{f}$ identical to f, ε remains, because ε holds things we cannot measure.*

$$E(Y - \hat{Y})^2 = [f(X) - \hat{f}(X)]^2 + Var(\varepsilon) \quad (1.2)$$

Ở đây E là giá trị trung bình khi lặp lại nhiều lần, còn Var(ε) là phương sai của ε, tức ε dao động rộng đến đâu quanh 0. Số hạng thứ nhất giảm được, số hạng thứ hai thì không.

*Here E is the average over many repetitions, and Var(ε) is the variance of ε, meaning how widely ε swings around 0. The first term can be reduced; the second cannot.*

**VÍ DỤ / EXAMPLE –** Tại x = 4 triệu đồng, f(4) = 6 trăm khách nhưng $\hat{f}(4)$ = 5,5; Var(ε) = 0,8. Phần giảm được: $(6 - 5{,}5)^2 = 0{,}25$. Phần không giảm được: 0,8. Tổng: 0,25 + 0,8 = 1,05. Nghĩa là dù sửa $\hat{f}$ cho đúng hẳn, sai số bình phương trung bình vẫn còn 0,8 và không thể xuống thấp hơn.

*At x = 4 million dong, f(4) = 6 hundred customers but $\hat{f}(4)$ = 5.5; Var(ε) = 0.8. Reducible part: $(6 - 5.5)^2 = 0.25$. Irreducible part: 0.8. Total: 0.25 + 0.8 = 1.05. So even if you fix $\hat{f}$ exactly, the mean squared error stays at 0.8 and cannot fall lower.*

Trên dữ liệu thật ta không biết f nên không tách được hai phần này. Vì vậy ta đo sai số trên **tập kiểm tra** (test set): phần dữ liệu để dành, không dùng khi dựng $\hat{f}$ (Chương 2).

*On real data we do not know f, so we cannot split the two parts. That is why we measure error on the **test set**: a part of the data held back and not used to build $\hat{f}$ (Chapter 2).*

**LỖI HAY GẶP / COMMON MISTAKE –** Coi ε là lỗi nhập liệu và hy vọng làm sạch dữ liệu sẽ đưa sai số về 0. Làm sạch loại được lỗi ghi sổ, nhưng phần lớn ε là những yếu tố không nằm trong bảng dữ liệu.

*Treating ε as data-entry error and hoping that cleaning will bring the error to 0. Cleaning removes bookkeeping slips, but most of ε is factors that are not in the data table.*

## 1.3. Vì sao ước lượng f: dự báo và suy diễn – *1.3. Why estimate f: prediction and inference*

Có hai lý do để ước lượng f, và chúng đòi hai kiểu mô hình khác nhau. Lý do thứ nhất là **dự báo** (prediction): ta cần con số Ŷ cho một trường hợp mới và không quan tâm f trông ra sao. Khi đó f như một hộp đen, miễn đầu ra chính xác. Ví dụ ở Lúa Vàng: dự báo xác suất vỡ nợ của hồ sơ vừa nộp, dự báo số tiền khách rút ở cây ATM ngày mai, đánh dấu giao dịch thẻ nghi gian lận.

*There are two reasons to estimate f, and they call for different kinds of model. The first is **prediction**: you need the number Ŷ for a new case and do not care what f looks like. Then f is a black box, as long as the output is accurate. Lua Vang examples: the default probability of a file just submitted, the cash customers withdraw at an ATM tomorrow, flagging card transactions that look fraudulent.*

Lý do thứ hai là **suy diễn** (inference): ta muốn hiểu f. Biến nào có ảnh hưởng? Ảnh hưởng cùng chiều hay ngược chiều? Mối quan hệ gần đường thẳng hay cong? Ví dụ: mỗi lần trễ hạn thêm làm xác suất vỡ nợ đổi bao nhiêu, hay thu nhập và số lần trễ hạn biến

nào ảnh hưởng mạnh hơn. Muốn suy diễn thì f phải mở ra được, ta không thể để nó là hộp đen.

*The second reason is **inference**: you want to understand f. Which variables matter? Do they push the same way or the opposite way? Is the relationship close to a line or curved? Examples: how much each extra late payment changes the default probability, or which of income and late payments matters more. To infer, f must be open to inspection; it cannot stay a black box.*

| **Câu hỏi của chị Linh**<br>*Ms Linh's question* | **Mục tiêu**<br>*Aim* | **Vì sao**<br>*Why* |
|---|---|---|
| "Hồ sơ mới này vỡ nợ với xác suất bao nhiêu?"<br>*"What is the default probability of this new file?"* | Dự báo<br>*Prediction* | Cần một con số cho từng hồ sơ.<br>*One number per file is needed.* |
| "Thu nhập hay số lần trễ hạn ảnh hưởng đến vỡ nợ nhiều hơn?"<br>*"Does income or number of late payments affect default more?"* | Suy diễn<br>*Inference* | So mức ảnh hưởng của hai biến.<br>*Compares the effect of two variables.* |
| "Ngày mai cây ATM ở trụ sở cần nạp bao nhiêu tiền?"<br>*"How much cash does the head-office ATM need tomorrow?"* | Dự báo<br>*Prediction* | Chỉ cần con số.<br>*Only the number is needed.* |
| "Vì sao hồ sơ của khách A bị từ chối?"<br>*"Why was customer A's file rejected?"* | Suy diễn kèm dự báo<br>*Inference plus prediction* | Phải chỉ ra biến nào kéo điểm xuống.<br>*You must show which variable pulled the score down.* |
| "Chi thêm quảng cáo có làm tăng khách mới không?"<br>*"Does more advertising raise new customers?"* | Suy diễn<br>*Inference* | Cần mức tăng; hai đại lượng cùng chiều chưa chứng minh nguyên nhân.<br>*The size of the rise is needed; moving together does not prove a cause.* |

Đọc bảng: câu 4 cho thấy hai mục tiêu thường đi chung. Chấm điểm tín dụng cần dự báo đúng, nhưng khách bị từ chối có quyền hỏi lý do, nên mô hình còn phải giải thích được. Câu 5 nhắc rằng suy diễn cẩn thận hơn dự báo: một mô hình dự báo tốt chưa đủ để kết luận nguyên nhân.

*Reading the table: row 4 shows that the two aims often travel together. Credit scoring must predict well, but a rejected customer has the right to ask why, so the model must also be explainable. Row 5 reminds us that inference needs more care than prediction: a good forecasting model is not enough to conclude a cause.*

Một lời dặn cho dự báo tài chính. Lợi suất ngày của một rổ cổ phiếu có ε lớn so với phần f giải thích được. Với bài toán loại này, ta so mô hình với **mô hình cơ sở** (baseline), tức cách dự báo đơn giản nhất như luôn đoán giá trị trung bình. Kết luận luôn có dạng "trên tập kiểm tra, mô hình đạt X; mô hình cơ sở đạt Y", kèm một câu về giới hạn. Không có lời hứa kiếm lời. Với mô hình tín dụng, không dùng giới tính, dân tộc, tôn giáo làm biến dự báo.

*A word of warning on financial forecasting. The daily return of a basket of shares has a large ε compared with what f explains. For problems like this, we compare the model with a **baseline model**, the simplest way to predict, such as always guessing the average. The conclusion always reads "on the test set, the model scores X; the baseline scores Y", with one sentence on limits. There is no promise of profit. In credit models, gender, ethnicity and religion are not used as predictors.*

## 1.4. Ước lượng f: phương pháp có tham số và phi tham số – *1.4. Estimating f: parametric and non-parametric methods*

Để ước lượng f, ta cần **dữ liệu huấn luyện** (training data): các cặp $(x_i, y_i)$ đã biết, như chi quảng cáo và số khách của từng tháng trong quá khứ. Có hai cách làm lớn.

*To estimate f we need **training data**: known pairs $(x_i, y_i)$, such as the advertising spend and customer count of each past month. There are two broad approaches.*

**Phương pháp có tham số** (parametric) làm hai việc. Việc một: chọn trước dạng của f, ví dụ đường thẳng. Việc hai: tìm các con số, gọi là tham số, để đường đó gần các điểm dữ liệu nhất. Với đường thẳng chỉ có hai tham số:

*A **parametric method** does two things. First, choose the form of f in advance, for example a straight line. Second, find the numbers, called parameters, that put that line closest to the data points. A straight line has just two parameters:*

$$\hat{f}(x) = \hat{\beta}_0 + \hat{\beta}_1 x \tag{1.3}$$

$\beta_0$ là **hệ số chặn** (intercept): giá trị dự báo khi x = 0. $\beta_1$ là **độ dốc** (slope): x tăng 1 đơn vị thì dự báo tăng $\beta_1$. Việc tìm hai số này gọi là **huấn luyện** hay khớp mô hình (fitting). Ưu điểm: ít tham số nên cần ít dữ liệu, và đọc được ngay. Nhược điểm: nếu quy luật thật là đường cong, đường thẳng sẽ lệch mãi, dù có thêm bao nhiêu dữ liệu.

*$\beta_0$ is the **intercept**: the prediction when $x = 0$. $\beta_1$ is the **slope**: when x rises by 1 unit, the prediction rises by $\beta_1$. Finding these two numbers is called **training** or fitting the model. Strength: few parameters, so little data is needed and the result can be read at once. Weakness: if the true rule is a curve, the line stays off however much data you add.*

**Phương pháp phi tham số** (non-parametric) không chọn trước dạng hàm. Nó để dữ liệu tự quyết hình dạng. Ví dụ dễ nhất là **k láng giềng gần nhất** (k-nearest neighbours, viết tắt KNN): muốn dự báo tại $x_0$, ta tìm k tháng có chi quảng cáo gần $x_0$ nhất rồi lấy trung bình số khách của chúng.

*A **non-parametric method** does not fix the form of the function in advance. It lets the data decide the shape. The easiest example is **k-nearest neighbours** (KNN): to predict at $x_0$, find the k months whose advertising spend is closest to $x_0$ and average their customer counts.*

$$\hat{f}(x_0) = (1/k) \sum y_i, \text{ lấy k điểm gần } x_0 \text{ nhất} \tag{1.4}$$

Ưu điểm: uốn theo gần như mọi hình dạng. Nhược điểm: cần nhiều dữ liệu, và dễ bám theo cả nhiễu, nghĩa là **quá khớp** (overfitting). Quá khớp giống học sinh học thuộc đề cũ: điểm bài cũ cao, gặp đề mới thì kém (Chương 2).

*Strength: it bends to almost any shape. Weakness: it needs plenty of data and easily follows the noise too, which is **overfitting**. Overfitting is like a pupil who memorises old exam papers: top marks on the old paper, poor marks on a new one (Chapter 2).*

### Ví dụ làm mẫu: hai cách dự báo trên 8 tháng – *Worked example: two ways to predict from 8 months*

Hà My có 8 tháng dữ liệu: x là chi quảng cáo (triệu đồng/tháng), y là khách mới (trăm khách). Cô cần dự báo y khi $x_0 = 2{,}7$. Đây là dữ kiện đề bài.

*Ha My has 8 months of data: x is advertising spend (million dong per month) and y is new customers (hundreds). She needs to predict y at $x_0 = 2.7$. This is the given data.*

1. Tính trung bình: $\bar{x} = 36 / 8 = 4{,}5$ và $\bar{y} = 50 / 8 = 6{,}25$. Rồi lập bảng độ lệch so với trung bình.
   *Compute the means: $\bar{x} = 36 / 8 = 4.5$ and $\bar{y} = 50 / 8 = 6.25$. Then build a table of deviations from the means.*

| x | y | $x - \bar{x}$ | $y - \bar{y}$ | Tích *Product* | $(x - \bar{x})^2$ |
|---|---|---|---|---|---|
| 1 | 2 | −3,5 | −4,25 | 14,875 | 12,25 |
| 2 | 4 | −2,5 | −2,25 | 5,625 | 6,25 |
| 3 | 5 | −1,5 | −1,25 | 1,875 | 2,25 |
| 4 | 6 | −0,5 | −0,25 | 0,125 | 0,25 |
| 5 | 7 | 0,5 | 0,75 | 0,375 | 0,25 |
| 6 | 8 | 1,5 | 1,75 | 2,625 | 2,25 |
| 7 | 9 | 2,5 | 2,75 | 6,875 | 6,25 |
| 8 | 9 | 3,5 | 2,75 | 9,625 | 12,25 |
| Tổng *Total* | 36 | 50 | 0 | 42 | 42 |

Đọc bảng: cột "Tích" nhân hai độ lệch của cùng một tháng. Hai cột cuối cộng lại đều bằng 42.

*Reading the table: the "Product" column multiplies the two deviations of the same month. The last two columns both add up to 42.*

2. Độ dốc bằng tổng cột "Tích" chia tổng cột $(x - \bar{x})^2$: $\beta_1 = 42 / 42 = 1$. (Chương 3 giải thích vì sao công thức này cho đường gần dữ liệu nhất.)
   *The slope is the total of the Product column divided by the total of the $(x - \bar{x})^2$ column: $\beta_1 = 42 / 42 = 1$. (Chapter 3 explains why this formula gives the line closest to the data.)*
3. Hệ số chặn: $\beta_0 = \bar{y} - \beta_1 * \bar{x} = 6{,}25 - 1 * 4{,}5 = 1{,}75$. Đường thẳng là $\hat{f}(x) = 1{,}75 + x$.
   *Intercept: $\beta_0 = \bar{y} - \beta_1 * \bar{x} = 6.25 - 1 * 4.5 = 1.75$. The line is $\hat{f}(x) = 1.75 + x$.*
4. Dự báo có tham số tại $x_0 = 2{,}7$: $1{,}75 + 2{,}7 = 4{,}45$ trăm khách.
   *Parametric prediction at $x_0 = 2.7$: $1.75 + 2.7 = 4.45$ hundred customers.*

5. Dự báo phi tham số với k = 3: ba tháng gần 2,7 nhất là x = 3 (cách 0,3), x = 2 (cách 0,7), x = 4 (cách 1,3). Số khách của chúng là 5, 4, 6. Trung bình: 15 / 3 = 5 trăm khách.
   *Non-parametric prediction with k = 3: the three months closest to 2.7 are x = 3 (distance 0.3), x = 2 (0.7) and x = 4 (1.3). Their customer counts are 5, 4, 6. Average: 15 / 3 = 5 hundred customers.*

6. Đo độ khớp của đường thẳng: phần dư $y - \hat{y}$ của 8 tháng là −0,75; 0,25; 0,25; 0,25; 0,25; 0,25; 0,25; −0,75. Bình phương rồi lấy trung bình: $(2 * 0{,}5625 + 6 * 0{,}0625) / 8 = 1{,}5 / 8 = 0{,}1875$. Đây là **sai số trung bình bình phương** (mean squared error, viết tắt MSE) trên chính dữ liệu huấn luyện.
   *Measure how well the line fits: the residuals $y - \hat{y}$ of the 8 months are −0.75, 0.25, 0.25, 0.25, 0.25, 0.25, 0.25, −0.75. Square and average: $(2 * 0.5625 + 6 * 0.0625) / 8 = 1.5 / 8 = 0.1875$. This is the **mean squared error** (MSE) on the training data itself.*

Đọc kết quả: tại $x_0 = 2{,}7$ đường thẳng cho 4,45 và trung bình láng giềng cho 5,00; hai cách lệch nhau 0,55 trăm khách. Bây giờ cho máy làm lại, và so với tay.

*Reading the result: at $x_0 = 2.7$ the line gives 4.45 and the neighbour average gives 5.00; the two methods differ by 0.55 hundred customers. Now let the computer repeat the work, and compare with the hand result.*

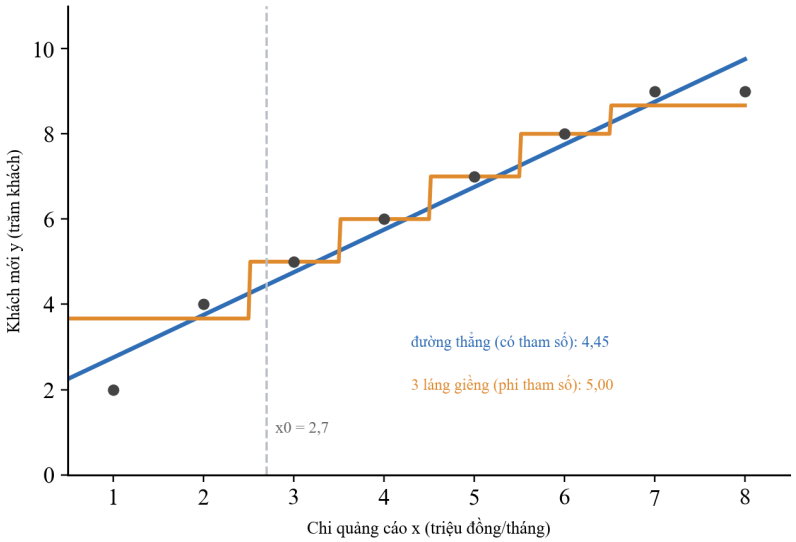
```
x = [1, 2, 3, 4, 5, 6, 7, 8]   # chi quảng cáo (triệu đồng/tháng)
y = [2, 4, 5, 6, 7, 8, 9, 9]   # khách mới (trăm khách)
n = len(x)
xb = sum(x) / n
yb = sum(y) / n
sxy = sum((a - xb) * (b - yb) for a, b in zip(x, y))
sxx = sum((a - xb) ** 2 for a in x)
b1 = sxy / sxx                  # độ dốc
b0 = yb - b1 * xb               # hệ số chặn
print(f"hệ số chặn b0 = {b0:.2f}, độ dốc b1 = {b1:.2f}")
x0 = 2.7
duong_thang = b0 + b1 * x0
cap = sorted(zip(x, y), key=lambda t: abs(t[0] - x0))
gan = cap[:3]
print("3 điểm gần x0 =", x0, ":", gan)
lang_gieng = sum(b for a, b in gan) / 3
print(f"dự báo đường thẳng tại x0 : {duong_thang:.2f}")
print(f"dự báo 3 láng giềng       : {lang_gieng:.2f}")
du = [b - (b0 + b1 * a) for a, b in zip(x, y)]
print("phần dư:", [round(r, 2) for r in du])
print(f"MSE huấn luyện = {sum(r * r for r in du) / n:.4f}")
```

**Kết quả chạy / *Output***

```
hệ số chặn b0 = 1.75, độ dốc b1 = 1.00
3 điểm gần x0 = 2.7 : [(3, 5), (2, 4), (4, 6)]
dự báo đường thẳng tại x0 : 4.45
dự báo 3 láng giềng       : 5.00
phần dư: [-0.75, 0.25, 0.25, 0.25, 0.25, 0.25, 0.25, -0.75]
MSE huấn luyện = 0.1875
```

Kết quả khớp với tay: $\beta_0 = 1{,}75$, $\beta_1 = 1{,}00$, dự báo 4,45 và 5,00, MSE huấn luyện 0,1875.

*The output matches the hand work: $\beta_0 = 1.75$, $\beta_1 = 1.00$, predictions 4.45 and 5.00, training MSE 0.1875.*




**Hình 1.2.** Đường xanh thẳng là phương pháp có tham số; đường cam gập bậc là trung bình 3 láng giềng. Đường cam bám sát từng chấm, đường xanh chỉ đi theo xu hướng chung. Tại đường nét đứt $x_0 = 2,7$, hai cách cho 4,45 và 5,00.

***Figure 1.2.*** *The straight blue line is the parametric method; the stepped orange line is the 3-neighbour average. The orange line hugs each dot, while the blue line follows only the general trend. At the dashed line $x_0 = 2.7$ the two methods give 4.45 and 5.00.*

| **Tiêu chí**<br>*Criterion* | **Có tham số**<br>*Parametric* | **Phi tham số**<br>*Non-parametric* |
|---|---|---|
| Dạng của f<br>*Form of f* | Chọn trước (đường thẳng, …)<br>*Chosen in advance (a line, ...)* | Để dữ liệu quyết<br>*Left to the data* |
| Việc phải làm<br>*Task* | Tìm vài tham số<br>*Find a few parameters* | Giữ lại dữ liệu và tìm điểm gần nhất<br>*Keep the data and find the nearest points* |
| Lượng dữ liệu cần<br>*Data needed* | Ít<br>*Little* | Nhiều<br>*A lot* |
| Rủi ro chính<br>*Main risk* | Chọn dạng sai nên lệch mãi<br>*Wrong form, so it stays off* | Quá khớp, bám nhiễu<br>*Overfitting, following noise* |
| Dễ đọc<br>*Readability* | Đọc được hệ số<br>*Coefficients can be read* | Khó nói biến nào ảnh hưởng thế nào<br>*Hard to say how each variable acts* |

**PHẢI THUỘC / MEMORISE –** Có tham số: chọn dạng trước, tìm vài số. Phi tham số: không chọn dạng, để dữ liệu quyết. Hai cách đều chạy trên dữ liệu huấn luyện và đều phải được chấm trên dữ liệu mới.
*Parametric: pick the form first, find a few numbers. Non-parametric: pick no form, let the data decide. Both run on training data and both must be scored on new data.*

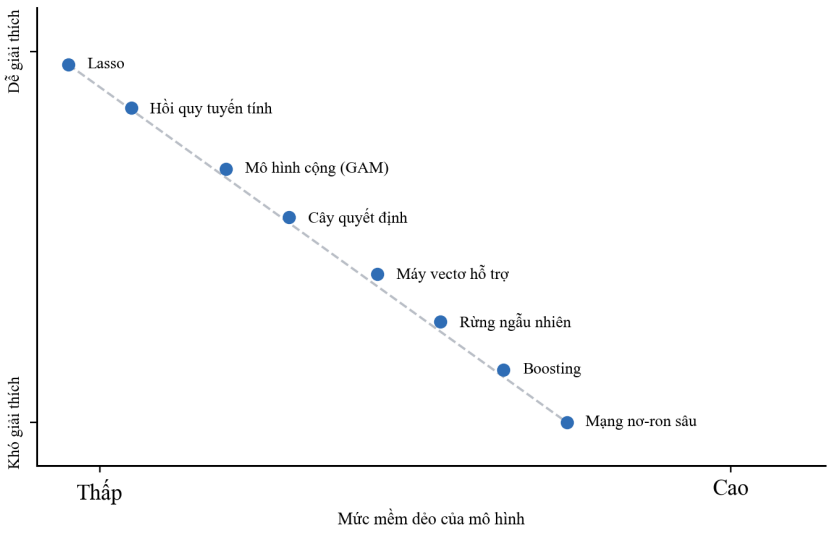
**LỖI HAY GẶP / COMMON MISTAKE –** Tưởng "phi tham số" là mô hình không có con số nào để chọn. KNN vẫn phải chọn k, và k do bạn quyết định trước khi huấn luyện, không phải dữ liệu quyết.

*Thinking "non-parametric" means there is nothing to choose. KNN still needs k, and you decide k before training; the data do not decide it.*

**LỖI HAY GẶP / COMMON MISTAKE –** Quên chuẩn hóa biến trước khi dùng KNN. Thu nhập tính bằng triệu đồng còn số lần trễ tính bằng đơn vị, nên khoảng cách bị thu nhập lấn át. Hãy đưa các biến về cùng thang đo trước.
*Forgetting to standardise variables before KNN. Income is in millions and late payments in single units, so distance is swamped by income. Bring the variables to a common scale first.*

## 1.5. Đánh đổi giữa độ chính xác và khả năng diễn giải – *1.5. The trade-off between accuracy and interpretability*

Ta gọi **mức mềm dẻo** (flexibility) là số hình dạng mà mô hình uốn được. Đường thẳng có mức mềm dẻo thấp: nó chỉ vẽ được đường thẳng. Trung bình láng giềng có mức mềm dẻo cao: nó vẽ được gần như mọi đường. Mô hình càng mềm dẻo càng khó đọc. Khi chị Linh hỏi "vì sao hồ sơ này bị từ chối?", mô hình tuyến tính trả lời được: "Dư nợ cao, mỗi 10 triệu đồng thêm làm điểm giảm 3". Một mạng nơ-ron sâu (nhiều lớp phép tính chồng nhau) có hàng triệu con số thì không có câu trả lời gọn như vậy.
*We call **flexibility** the number of shapes a model can bend into. A straight line has low flexibility: it can only draw straight lines. The neighbour average has high flexibility: it can draw almost any curve. The more flexible a model is, the harder it is to read. When Ms Linh asks "why was this file rejected?", a linear model can answer: "Debt is high; each extra 10 million dong lowers the score by 3." A deep neural network (many stacked layers of calculation) with millions of numbers has no answer that short.*




**Hình 1.3.** Trục ngang là mức mềm dẻo, trục dọc là mức dễ giải thích. Đi từ trái sang phải, các phương pháp uốn được nhiều hơn nhưng khó đọc hơn: Lasso và hồi quy tuyến tính ở góc trên bên trái, mạng nơ-ron sâu ở góc dưới bên phải.
***Figure 1.3.** The horizontal axis is flexibility and the vertical axis is ease of explanation. From left to right the methods bend more but are harder to read: Lasso and linear regression sit top left, the deep neural network bottom right.*

| **Phương pháp**<br>*Method* | **Mức mềm dẻo**<br>*Flexibility* | **Dễ giải thích**<br>*Ease of explanation* | **Học ở**<br>*Studied in* |
|---|---|---|---|
| Hồi quy tuyến tính<br>*Linear regression* | Thấp<br>*Low* | Cao<br>*High* | Chương 3<br>*Chapter 3* |
| Lasso (hồi quy tuyến tính có phạt)<br>*Lasso (penalised linear regression)* | Thấp<br>*Low* | Cao<br>*High* | Chương 6<br>*Chapter 6* |
| Mô hình cộng tổng quát (GAM)<br>*Generalised additive model (GAM)* | Vừa<br>*Medium* | Khá cao<br>*Fairly high* | Chương 7<br>*Chapter 7* |
| Cây quyết định<br>*Decision tree* | Vừa<br>*Medium* | Khá cao<br>*Fairly high* | Chương 8<br>*Chapter 8* |
| Máy vectơ hỗ trợ (SVM)<br>*Support vector machine (SVM)* | Cao<br>*High* | Thấp<br>*Low* | Chương 9<br>*Chapter 9* |
| Rừng ngẫu nhiên, boosting<br>*Random forest, boosting* | Cao<br>*High* | Thấp<br>*Low* | Chương 8<br>*Chapter 8* |

Đọc bảng: cột "Học ở" cho biết bạn sẽ gặp từng phương pháp sau. Đừng nghĩ mô hình mềm dẻo hơn luôn chính xác hơn. Nếu mềm dẻo quá mức, mô hình học thuộc cả nhiễu và sai khi gặp dữ liệu mới. Đôi khi đường thẳng dự báo tốt hơn một mô hình phức tạp trên dữ liệu mới.

*Reading the table: the "Studied in" column tells you where each method comes later. Do not assume a more flexible model is always more accurate. Too much flexibility makes a model memorise the noise and fail on new data. Sometimes a straight line predicts new data better than a complex model.*

Cách chọn theo mục đích: khi cần suy diễn hoặc cần giải thích cho khách và cơ quan quản lý, bắt đầu từ mô hình ít mềm dẻo. Khi chỉ cần dự báo, thử nhiều mô hình và chọn theo sai số trên tập kiểm tra.

*Choosing by purpose: when you need inference, or must explain to customers and regulators, start from a model with low flexibility. When you only need prediction, try several models and pick by error on the test set.*

> **LỖI HAY GẶP / COMMON MISTAKE –** Cho rằng mô hình mềm dẻo hơn thì chắc chắn chính xác hơn. Hãy tin số đo trên tập kiểm tra, không tin cảm giác về độ phức tạp.
> *Assuming a more flexible model is certainly more accurate. Trust the number on the test set, not the feeling of complexity.*

## 1.6. Học có giám sát, không giám sát và bán giám sát – *1.6. Supervised, unsupervised and semi-supervised learning*

**Học có giám sát** (supervised learning) giống học sinh làm bài có đáp án: mỗi hồ sơ có sẵn kết quả y (đã vỡ nợ hay chưa), máy thử đoán rồi đối chiếu với đáp án để sửa. Mọi ví dụ ở mục 1.2 đến 1.5 thuộc loại này.

***Supervised learning*** *is like a pupil doing exercises with an answer key: every file comes with its outcome y (defaulted or not), and the machine guesses, then checks against the answer to correct itself. Every example in sections 1.2 to 1.5 is of this kind.*

**Học không giám sát** (unsupervised learning) là khi chỉ có X, không có đáp án. Máy không được hỏi "đúng hay sai" mà được nhờ "tìm cấu trúc trong dữ liệu". Ví dụ: chia 50.000 khách hàng thành vài nhóm có thói quen chi tiêu giống nhau (phân cụm, clustering), để phòng kinh doanh thiết kế sản phẩm riêng cho từng nhóm. Chương 10 nói về loại này.

***Unsupervised learning*** *is when there is only X and no answer. The machine is not asked "right or wrong" but "find structure in the data". Example: split 50,000 customers into a few groups with similar spending habits (clustering), so the sales team can design a product for each group. Chapter 10 covers this kind.*

**Học bán giám sát** (semi-supervised learning) nằm giữa hai loại trên: chỉ một phần nhỏ quan sát (mỗi hồ sơ, giao dịch hay ngày là một quan sát) có đáp án, phần lớn không. Nhãn khan hiếm vì tốn công người. Ví dụ: điều tra viên mới kiểm tra và gán nhãn gian lận cho 500 giao dịch thẻ, còn 500.000 giao dịch khác chưa ai xem. Ví dụ khác: Lúa Vàng có 20.000 hồ sơ, nhưng chỉ 3.000 hồ sơ đủ 12 tháng để biết có vỡ nợ hay không; 17.000 hồ sơ còn lại chưa có kết quả. Dùng riêng 3.000 hồ sơ là học có giám sát. Dùng cả 20.000 hồ sơ để rút thêm thông tin là học bán giám sát.

***Semi-supervised learning*** *sits between the two: only a small share of observations (each file, transaction or day is one observation) has an answer and most do not. Labels are scarce because people's time is costly. Example: investigators have checked and labelled fraud for only 500 card transactions, while 500,000 others have not been looked at. Another example: Lua Vang has 20,000 files, but only 3,000 are old enough (12 months) to know whether they defaulted; the other 17,000 have no outcome yet. Using the 3,000 alone is supervised learning. Using all 20,000 to draw extra information is semi-supervised learning.*

| **Loại**<br>*Type* | **Có đáp án y?**<br>*Has answer y?* | **Mục tiêu**<br>*Aim* | **Ví dụ ở Lúa Vàng**<br>*Lua Vang example* |
|---|---|---|---|
| Có giám sát<br>*Supervised* | Có, cho mọi quan sát<br>*Yes, for every observation* | Dự báo y hoặc hiểu quan hệ X và y<br>*Predict y or understand the link between X and y* | Chấm điểm tín dụng<br>*Credit scoring* |
| Không giám sát<br>*Unsupervised* | Không<br>*No* | Tìm nhóm hoặc cấu trúc trong X<br>*Find groups or structure in X* | Chia nhóm khách hàng<br>*Customer segmentation* |
| Bán giám sát<br>*Semi-supervised* | Chỉ một phần nhỏ<br>*Only a small part* | Dùng cả dữ liệu có và không có nhãn<br>*Use labelled and unlabelled data together* | Phát hiện gian lận khi ít nhãn<br>*Fraud detection with few labels* |

Đọc bảng: cách nhanh nhất để phân loại là hỏi "bảng dữ liệu có cột đáp án không, và cột đó điền đủ cho bao nhiêu dòng?". Đủ mọi dòng thì có giám sát, không có dòng nào thì không giám sát, chỉ vài dòng thì bán giám sát.

*Reading the table: the quickest way to classify is to ask "does the data table have an answer column, and for how many rows is it filled in?". All rows means supervised, no rows means unsupervised, a few rows means semi-supervised.*

## 1.7. Bài toán hồi quy và bài toán phân lớp – *1.7. Regression and classification problems*

Biến có hai loại. **Biến định lượng** (quantitative) là số đo mà phép tính có nghĩa: thu nhập, số tiền giao dịch, số ngày trễ hạn. **Biến định tính** (qualitative, còn gọi categorical) là nhóm hoặc nhãn: vỡ nợ hay không, kênh vay (chi nhánh, trực tuyến, đại lý), xếp hạng nội bộ A, B hoặc C.

*Variables come in two kinds. A **quantitative** variable is a measurement for which arithmetic makes sense: income, transaction amount, days overdue. A **qualitative** variable (also called categorical) is a group or label: defaulted or not, loan channel (branch, online, agent), internal grade A, B or C.*

Loại của biến đầu ra Y quyết định loại bài toán. Y định lượng thì là bài toán **hồi quy** (regression). Y định tính thì là bài toán **phân lớp** (classification). Loại của biến đầu vào X không quyết định điều này: X có thể là số, là nhóm, hoặc lẫn cả hai.

*The type of the response Y decides the type of problem. A quantitative Y gives a **regression** problem. A qualitative Y gives a **classification** problem. The type of the input X does not decide this: X can be numbers, groups, or a mix.*

| **Bài toán**<br>*Problem* | **Biến đầu ra Y**<br>*Response Y* | **Loại**<br>*Type* |
|---|---|---|
| Dự báo số tiền khách rút ở cây ATM ngày mai<br>*Forecast the cash withdrawn at an ATM tomorrow* | Số tiền (đồng)<br>*Amount (dong)* | Hồi quy<br>*Regression* |
| Dự báo hồ sơ vay có vỡ nợ không<br>*Predict whether a loan file defaults* | Có hoặc không<br>*Yes or no* | Phân lớp<br>*Classification* |
| Xếp hồ sơ vào hạng nội bộ A, B hoặc C<br>*Place a file into internal grade A, B or C* | Hạng A, B, C<br>*Grade A, B, C* | Phân lớp (3 lớp)<br>*Classification (3 classes)* |
| Dự báo số ngày trễ hạn của khoản vay<br>*Predict days overdue for a loan* | Số ngày<br>*Number of days* | Hồi quy<br>*Regression* |
| Đánh dấu giao dịch thẻ gian lận<br>*Flag fraudulent card transactions* | Gian lận hoặc không<br>*Fraud or not* | Phân lớp<br>*Classification* |
| Dự báo lợi suất ngày của rổ cổ phiếu<br>*Predict the daily return of a share basket* | Lợi suất (%)<br>*Return (%)* | Hồi quy<br>*Regression* |

Đọc bảng: dòng 3 vẫn là phân lớp dù có ba nhóm. Ranh giới đôi khi mờ: hồi quy logistic (Chương 4) mang tên "hồi quy" nhưng dự báo xác suất thuộc một lớp, nên nó dùng để phân lớp. Cứ nhìn Y là số đo hay là nhóm thì không sai.

*Reading the table: row 3 is still classification even though there are three groups. The line is sometimes blurred: logistic regression (Chapter 4) carries the word "regression" but predicts the probability of a class, so it is used for classification. Looking at whether Y is a measurement or a group never misleads.*

## 1.8. Dữ liệu trong tài chính – *1.8. Data in finance*

Dữ liệu tài chính thường nằm trong bảng: mỗi dòng là một **quan sát** (observation, ví dụ một hồ sơ vay, một giao dịch, một ngày giao dịch), mỗi cột là một **biến** (variable). Bảng sau liệt kê các nguồn hay gặp và điều cần để ý ở mỗi nguồn.

*Financial data usually sit in tables: each row is an **observation** (for example a loan file, a transaction or a trading day) and each column is a **variable**. The next table lists common sources and what to watch for in each.*

| **Nguồn** *Source* | **Biến ví dụ** *Example variables* | **Dạng** *Form* | **Điều cần để ý** *What to watch* |
|---|---|---|---|
| Hồ sơ vay *Loan files* | Thu nhập, dư nợ, số lần trễ hạn, vỡ nợ có hoặc không *Income, debt, late payments, default yes or no* | Bảng, mỗi dòng một hồ sơ *Table, one file per row* | Nhiều ô trống; chỉ có hồ sơ đã duyệt *Many blanks; only approved files* |
| Giao dịch thẻ *Card transactions* | Số tiền, giờ, khoảng cách tới nơi ở, gian lận có hoặc không *Amount, hour, distance from home, fraud yes or no* | Bảng dài, gian lận hiếm *A long table, fraud is rare* | Mất cân bằng nhãn *Unbalanced labels* |
| Giá chứng khoán *Share prices* | Giá đóng cửa, khối lượng, lợi suất ngày *Closing price, volume, daily return* | Chuỗi theo thời gian *Time series* | Thứ tự thời gian quan trọng *Time order matters* |
| Khảo sát, đơn từ *Surveys, applications* | Câu trả lời, ghi chú của nhân viên *Answers, staff notes* | Chữ *Text* | Phải đổi thành số trước khi dùng *Must be turned into numbers first* |
| Hành vi trên ứng dụng *App behaviour* | Số lần đăng nhập, tính năng đã dùng *Logins, features used* | Nhật ký sự kiện *Event log* | Dữ liệu cá nhân, cần căn cứ pháp lý *Personal data, needs a legal basis* |

Đọc bảng: cột cuối là điều dễ bị bỏ quên. Hồ sơ vay chỉ có người đã được duyệt, nên mô hình không biết người bị từ chối ra sao. Giao dịch thẻ có gian lận hiếm, nên mô hình "luôn báo không gian lận" trông đúng gần hết (Chương 2).

*Reading the table: the last column is the one people forget. Loan files contain only people who were approved, so the model does not know how rejected people behave. Fraud is rare in card transactions, so a model that "always says no fraud" looks right almost every time (Chapter 2).*

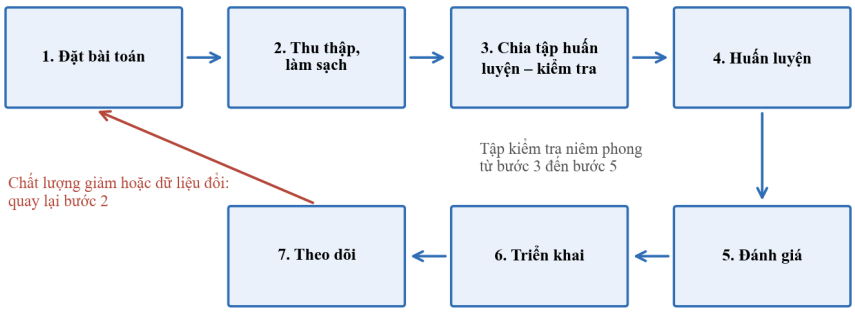
Hai tên gọi cần nhớ khi đọc một bảng: biến đầu vào còn gọi là **đặc trưng** (feature), và biến đầu ra còn gọi là **nhãn** (label) trong bài toán phân lớp. Trước khi dùng, bạn kiểm ba điều: kiểu của từng cột (số hay chữ), ô trống, và giá trị lạ (thu nhập âm, tuổi 300).

*Two names to remember when reading a table: an input variable is also called a **feature**, and the output variable is also called a **label** in a classification problem. Before use, check three things: the type of each column (number or text), blank cells, and odd values (negative income, age 300).*

## 1.9. Quy trình một dự án học máy – *1.9. The workflow of a machine learning project*

Anh Quân nói với Hà My: "Thuật toán chỉ là một bước trong bảy bước. Phần lớn thời gian nằm ở trước và sau nó." Hình sau là bản đồ của bảy bước.

*Quan tells Ha My: "The algorithm is only one of seven steps. Most of the time goes before it and after it." The next figure is a map of the seven steps.*




**Hình 1.4.** Đi theo mũi tên xanh từ bước 1 đến bước 7. Mũi tên đỏ là đường quay lại: khi chất lượng ở bước 7 giảm, ta trở về bước 2. Từ bước 3 đến bước 5, tập kiểm tra được giữ nguyên, không ai nhìn vào.

***Figure 1.4.** Follow the blue arrows from step 1 to step 7. The red arrow is the way back: when quality drops at step 7, we return to step 2. From step 3 to step 5 the test set is kept sealed and nobody looks at it.*

1. **Đặt bài toán.** Ai dùng kết quả, để quyết định gì, đo thành công bằng gì. Ví dụ: giảm tỷ lệ nợ xấu mà không từ chối quá nhiều khách tốt. Viết ra Y là gì, X lấy từ đâu, và đây là hồi quy hay phân lớp.
   ***Frame the problem.** Who uses the result, to decide what, and how success is measured. Example: cut the bad-debt rate without rejecting too many good customers. Write down what Y is, where X comes from, and whether this is regression or classification.*
2. **Thu thập và làm sạch.** Gom dữ liệu từ các nguồn, bỏ dòng trùng, xử lý ô trống, soát giá trị lạ, đổi chữ thành số khi cần.
   ***Collect and clean.** Gather the data from the sources, drop duplicate rows, handle blanks, check odd values, turn text into numbers when needed.*

3. **Chia tập huấn luyện – kiểm tra.** Cất một phần (thường 20% đến 30%) làm **tập kiểm tra** (test set), như đề thi niêm phong. Phần còn lại là **tập huấn luyện** (training set). Với dữ liệu theo thời gian, tập kiểm tra là giai đoạn sau.
   ***Split into training and test sets.*** *Put a part (usually 20% to 30%) aside as the **test set**, like a sealed exam paper. The rest is the **training set**. With time-ordered data, the test set is the later period.*
4. **Huấn luyện.** Chọn thuật toán, cho nó học trên tập huấn luyện, và chỉnh các lựa chọn (như k của KNN) bằng cách chia nhỏ tập huấn luyện (Chương 5).
   ***Train.*** *Choose an algorithm, let it learn on the training set, and tune its choices (such as the k of KNN) by splitting the training set further (Chapter 5).*
5. **Đánh giá.** Chấm điểm mô hình đã chốt trên tập kiểm tra đúng một lần, và so với **mô hình cơ sở** (baseline): cách dự báo đơn giản nhất, như luôn đoán giá trị trung bình, hoặc luôn đoán lớp phổ biến nhất.
   ***Evaluate.*** *Score the finished model on the test set exactly once, and compare it with a **baseline model**: the simplest way to predict, such as always guessing the average, or always guessing the most common class.*
6. **Triển khai.** Đưa mô hình vào hệ thống duyệt hồ sơ. Quyết định cuối cùng thường vẫn do người duyệt, mô hình chỉ cho điểm và lý do.
   ***Deploy.*** *Put the model into the file-approval system. The final decision usually still rests with a person; the model gives a score and reasons.*
7. **Theo dõi.** Mỗi tháng đo lại chất lượng trên hồ sơ mới. Khách hàng đổi thói quen, kinh tế đổi, nên mô hình cũ yếu dần. Khi chất lượng giảm, quay lại bước 2 với dữ liệu mới.
   ***Monitor.*** *Every month, measure quality again on new files. Customers change habits and the economy shifts, so an old model weakens over time. When quality drops, go back to step 2 with fresh data.*

**VÍ DỤ / EXAMPLE –** Dữ kiện đề bài: Lúa Vàng có 5.000 hồ sơ, 400 hồ sơ vỡ nợ (8%). Chia 70% huấn luyện, 30% kiểm tra: 3.500 hồ sơ huấn luyện và 1.500 hồ sơ kiểm tra. Nếu chia ngẫu nhiên giữ nguyên tỷ lệ, tập kiểm tra có khoảng $8\% * 1.500 = 120$ hồ sơ vỡ nợ. Mô hình cơ sở "luôn báo không vỡ nợ" đúng ở $1.500 - 120 = 1.380$ hồ sơ, tức $1.380 / 1.500 = 92\%$. Một mô hình đạt 93% chỉ hơn cơ sở 1 điểm phần trăm, chưa đủ để thuyết phục chị Linh.
*Given data: Lua Vang has 5,000 files, 400 of which defaulted (8%). Split 70% training and 30% test: 3,500 training files and 1,500 test files. If the split is random and keeps the proportion, the test set holds about $8\% * 1,500 = 120$ defaulted files. The baseline "always say no default" is right on $1,500 - 120 = 1,380$ files, which is $1,380 / 1,500 = 92\%$. A model scoring 93% beats the baseline by only 1 percentage point, not enough to convince Ms Linh.*

**LỖI HAY GẶP / COMMON MISTAKE –** Chấm mô hình trên chính tập huấn luyện. Mô hình đã thấy những dòng đó nên điểm luôn đẹp; chỉ tập kiểm tra mới cho biết mô hình gặp khách mới ra sao.
*Scoring the model on the training set itself. The model has seen those rows, so the score always looks good; only the test set shows how the model meets new customers.*

**LỖI HAY GẶP / COMMON MISTAKE –** Rò rỉ dữ liệu tương lai: tính trung bình để chuẩn hóa bằng cả bộ dữ liệu trước khi chia, hoặc dùng một biến chỉ biết sau khi khoản vay kết thúc. Mô hình vô tình "nhìn trộm" đáp án. Hãy chia trước, và chỉ tính thống kê từ tập huấn luyện.
*Leakage of future data: computing the average for scaling on the whole data set before splitting, or using a variable known only after the loan ends. The model quietly peeks at the answers. Split first, and take statistics from the training set only.*

## 1.10. Ứng dụng tài chính, thiên lệch dữ liệu và nghề nghiệp – *1.10. Finance applications, data bias and careers*

### Ứng dụng tài chính – ngân hàng – *Applications in finance and banking*

Anh Quân vẽ cho Hà My bảng các việc ở Lúa Vàng đang dùng học máy, kèm loại học và loại bài toán của từng việc.
*Quan sketches for Ha My the jobs at Lua Vang that use machine learning, with the type of learning and the type of problem for each.*

| **Ứng dụng** *Application* | **Học** *Learning* | **Bài toán** *Problem* | **Câu hỏi của ứng dụng** *The question it answers* |
|---|---|---|---|
| Chấm điểm tín dụng *Credit scoring* | Có giám sát *Supervised* | Phân lớp *Classification* | Hồ sơ này có vỡ nợ không? *Will this file default?* |
| Phát hiện gian lận *Fraud detection* | Có giám sát, bán giám sát *Supervised, semi-supervised* | Phân lớp *Classification* | Giao dịch này có đáng ngờ không? *Is this transaction suspicious?* |
| Chia nhóm khách hàng *Customer segmentation* | Không giám sát *Unsupervised* | Phân cụm *Clustering* | Khách nào giống nhau? *Which customers resemble each other?* |
| Dự báo tiền ATM *ATM cash forecast* | Có giám sát *Supervised* | Hồi quy *Regression* | Ngày mai cần nạp bao nhiêu? *How much to load tomorrow?* |
| Dự báo lợi suất ngày *Daily return forecast* | Có giám sát *Supervised* | Hồi quy *Regression* | Chỉ so với mô hình cơ sở *Only compared with a baseline* |

Đọc bảng: dòng cuối là bài khó nhất vì ε lớn. Với bài này, bạn báo "trên tập kiểm tra, mô hình đạt X; mô hình cơ sở đạt Y", và nói rõ giới hạn.
*Reading the table: the last row is the hardest because ε is large. For this one, you report "on the test set, the model scores X; the baseline scores Y" and state the limits plainly.*

**LẬP LUẬN ỦNG HỘ – PHẢN ĐỐI / FOR AND AGAINST – Nên dùng học máy phức tạp để chấm điểm tín dụng không?** Ủng hộ: mô hình mềm dẻo bắt được quan hệ cong, ghép nhiều biến, thường dự báo đúng hơn trên tập kiểm tra. Phản đối: khó giải thích cho khách bị từ chối, khó kiểm tra thiên lệch, và dễ quá khớp khi dữ liệu ít. Cách cân nhắc: so mô hình phức tạp với một mô hình tuyến tính đơn giản trên cùng tập kiểm tra. Nếu mức hơn nhau nhỏ, chọn mô hình đơn giản.

> ***Should a complex machine learning model be used for credit scoring?*** *For: a flexible model catches curved relationships and combines many variables, and often predicts better on the test set. Against: it is hard to explain to a rejected customer, hard to check for bias, and prone to overfitting when data are few. How to weigh it: compare the complex model with a simple linear one on the same test set. If the gap is small, choose the simple model.*

### Thiên lệch dữ liệu, công bằng và dữ liệu cá nhân – *Data bias, fairness and personal data*

**Thiên lệch dữ liệu** (data bias) xảy ra khi dữ liệu không đại diện đúng cho những người mô hình sẽ phục vụ. Ví dụ quen thuộc: chỉ có hồ sơ đã được duyệt, nên mô hình học từ người đã qua cửa và không biết gì về người bị từ chối. Mô hình học đúng từ dữ liệu lệch vẫn cho kết quả lệch.

***Data bias*** *happens when the data do not fairly represent the people the model will serve. A familiar example: only approved files exist, so the model learns from people who got through and knows nothing about those rejected. A model that learns faithfully from skewed data still gives skewed results.*

**Công bằng** (fairness) nghĩa là mô hình không đối xử khác nhau với các nhóm người một cách không có căn cứ. Mô hình tín dụng không dùng giới tính, dân tộc, tôn giáo làm biến dự báo. Còn phải để ý **biến đại diện** (proxy): một biến tưởng vô hại nhưng lại phản ánh nhóm người, như khu vực sống. Cách kiểm đơn giản: so tỷ lệ duyệt và tỷ lệ lỗi giữa các nhóm trên tập kiểm tra.

***Fairness*** *means the model does not treat groups of people differently without grounds. A credit model does not use gender, ethnicity or religion as predictors. You must also watch for* ***proxy variables****: a variable that looks harmless but reflects a group of people, such as area of residence. A simple check: compare approval rates and error rates across groups on the test set.*

**Dữ liệu cá nhân** (personal data) như tên, số căn cước, lịch sử giao dịch của khách chỉ được thu thập và dùng khi có căn cứ theo quy định bảo vệ dữ liệu cá nhân, chẳng hạn khách đã đồng ý cho mục đích rõ ràng. Chỉ lấy cột cần dùng, bỏ tên và số định danh khi huấn luyện, và lưu lại ai đã mở dữ liệu nào.

***Personal data*** *such as names, ID numbers and customers' transaction history may be collected and used only on a basis allowed by personal data protection rules, for instance the customer agreed to a clear purpose. Take only the columns you need, drop names and identifiers when training, and log who opened which data.*

### Vị trí việc làm – *Job roles*

| **Vị trí**<br>*Role* | **Việc hằng ngày**<br>*Daily work* | **Dùng nhiều nhất**<br>*Used most* |
|---|---|---|
| Chuyên viên phân tích dữ liệu<br>*Data analyst* | Làm sạch dữ liệu, thống kê mô tả, báo cáo<br>*Clean data, descriptive statistics, reports* | Mục 1.8 và 1.11<br>*Sections 1.8 and 1.11* |
| Nhà khoa học dữ liệu<br>*Data scientist* | Dựng và so sánh mô hình<br>*Build and compare models* | Mục 1.4, 1.5 và 1.9<br>*Sections 1.4, 1.5 and 1.9* |

| **Vị trí**<br>*Role* | **Việc hằng ngày**<br>*Daily work* | **Dùng nhiều nhất**<br>*Used most* |
|---|---|---|
| Kỹ sư học máy<br>*Machine learning engineer* | Đưa mô hình vào hệ thống, theo dõi<br>*Put models into systems, monitor them* | Bước 6 và 7 của quy trình<br>*Steps 6 and 7 of the workflow* |
| Chuyên viên rủi ro tín dụng<br>*Credit risk officer* | Dùng điểm để duyệt hồ sơ, hỏi lý do<br>*Use scores to approve files, ask for reasons* | Mục 1.3 và 1.10<br>*Sections 1.3 and 1.10* |
| Chuyên viên kiểm định mô hình<br>*Model validator* | Kiểm độc lập, soát thiên lệch<br>*Independent checks, bias review* | Bước 3 và 5, mục 1.10<br>*Steps 3 and 5, section 1.10* |

## 1.11. Tính bằng Python – *1.11. Working in Python*

Bạn làm lại bằng Python các việc cơ bản với dữ liệu: lệnh cơ bản, chỉ mục, đọc tệp CSV, thống kê mô tả, bảng tần số, tóm tắt theo nhóm, và chia tập huấn luyện – kiểm tra. Khi làm thật, người ta dùng các thư viện như NumPy, pandas và scikit-learn. Ở đây mã chỉ dùng thư viện chuẩn của Python và tự viết từng bước, để bạn thấy máy làm gì bên trong.

*You now redo in Python the basic jobs with data: basic commands, indexing, reading a CSV file, descriptive statistics, frequency tables, summaries by group, and splitting training and test sets. In real work, people use libraries such as NumPy, pandas and scikit-learn. Here the code uses only Python's standard library and writes each step by hand, so you can see what the machine does inside.*

### Lệnh cơ bản: biến, danh sách, hàm – *Basic commands: variables, lists, functions*

**Biến** là cái tên gắn với một giá trị. **Danh sách** (list) giữ nhiều giá trị theo thứ tự. **Hàm** là đoạn mã có tên, nhận đầu vào và trả kết quả. Python đếm vị trí từ 0, nên phần tử đầu có chỉ mục 0 và phần tử cuối có chỉ mục −1.

*A **variable** is a name attached to a value. A **list** holds several values in order. A **function** is a named piece of code that takes inputs and returns a result. Python counts positions from 0, so the first item has index 0 and the last has index −1.*

```
thu_nhap = 18.5                  # triệu đồng/tháng
du_no = 54.0                     # triệu đồng
ty_le = du_no / thu_nhap
print("dư nợ trên thu nhập:", round(ty_le, 3), "lần")
so_lan_tre = [0, 2, 1, 0, 3, 0]  # danh sách: 6 hồ sơ
print("số hồ sơ:", len(so_lan_tre))
print("tổng số lần trễ:", sum(so_lan_tre))
print("hồ sơ thứ 3 (chỉ mục 2):", so_lan_tre[2])
print("3 hồ sơ đầu:", so_lan_tre[:3])
print("hồ sơ cuối:", so_lan_tre[-1])



def tre_nhieu(ds, nguong):
    return [i for i, v in enumerate(ds) if v >= nguong]



print("chỉ mục hồ sơ trễ từ 2 lần:", tre_nhieu(so_lan_tre, 2))
```

### Kết quả chạy / *Output*

```
dư nợ trên thu nhập: 2.919 lần
số hồ sơ: 6
tổng số lần trễ: 6
hồ sơ thứ 3 (chỉ mục 2): 1
3 hồ sơ đầu: [0, 2, 1]
hồ sơ cuối: 0
chỉ mục hồ sơ trễ từ 2 lần: [1, 4]
```

Đọc kết quả: dư nợ gấp 2,919 lần thu nhập tháng. Danh sách có 6 hồ sơ với tổng 6 lần trễ. Chỉ mục 2 là hồ sơ thứ ba (giá trị 1). Cắt lát [:3] lấy ba hồ sơ đầu. Hồ sơ ở chỉ mục 1 và 4 trễ từ 2 lần.

*Reading the output: debt is 2.919 times monthly income. The list has 6 files with 6 late payments in total. Index 2 is the third file (value 1). The slice [:3] takes the first three files. The files at index 1 and 4 are late twice or more.*

### Đọc dữ liệu từ tệp CSV và chỉ mục – *Reading data from a CSV file and indexing*

Tệp **CSV** (comma-separated values, giá trị cách nhau bằng dấu phẩy) là bảng lưu dưới dạng chữ, mỗi dòng một quan sát. Mã sau sinh 40 hồ sơ vay bằng số ngẫu nhiên với **hạt giống** (seed, số khởi đầu giúp dãy số ngẫu nhiên lặp lại y hệt) cố định, ghi vào bộ nhớ như một tệp CSV rồi đọc lại bằng mô-đun csv. Công thức xác suất vỡ nợ trong mã tăng theo số lần trễ và tỷ lệ dư nợ trên thu nhập. Đây là dữ kiện của bài tập.

*A **CSV** file (comma-separated values) is a table stored as text, one observation per line. The next code generates 40 loan files with random numbers from a fixed **seed** (a starting number that makes the random sequence repeat exactly), writes them into memory as if to a CSV file, and reads them back with the csv module. The default probability in the code rises with late payments and with the debt-to-income ratio. These are the figures of the exercise.*

```
import csv, io, random, statistics as st
random.seed(2025)
ghi = io.StringIO()               # chỗ chứa chữ, thay cho một tệp CSV
w = csv.writer(ghi)
w.writerow(["ma", "thu_nhap", "du_no", "tre", "kenh", "vo_no"])
for i in range(1, 41):
    tn = round(random.uniform(8, 40), 1)
    no = round(tn * random.uniform(0.5, 4.0), 1)
    tre = random.choice([0, 0, 0, 1, 1, 2, 3])
    kenh = random.choice(["chi nhánh", "trực tuyến", "đại lý"])
    p = 0.03 + 0.07 * tre + 0.02 * no / tn
    vo = 1 if random.random() < p else 0
    w.writerow([f"HS{i:02d}", tn, no, tre, kenh, vo])
ho_so = list(csv.DictReader(io.StringIO(ghi.getvalue())))
print("số dòng:", len(ho_so))
print("cột:", list(ho_so[0].keys()))
print("giá trị hồ sơ đầu:", list(ho_so[0].values()))
print("thu nhập hồ sơ thứ 5:", ho_so[4]["thu_nhap"])
print("kiểu dữ liệu đọc ra:", type(ho_so[4]["thu_nhap"]).__name__)
thu_nhap = [float(r["thu_nhap"]) for r in ho_so]
print("thu nhập lớn nhất:", max(thu_nhap))
tre2 = [r["ma"] for r in ho_so if int(r["tre"]) >= 2]
print("số hồ sơ trễ từ 2 lần:", len(tre2))
print("năm mã đầu tiên:", tre2[:5])
```

### Kết quả chạy / *Output*

```
số dòng: 40
cột: ['ma', 'thu_nhap', 'du_no', 'tre', 'kenh', 'vo_no']
giá trị hồ sơ đầu: ['HS01', '25.8', '71.2', '1', 'chi nhánh', '0']
thu nhập hồ sơ thứ 5: 14.2
kiểu dữ liệu đọc ra: str
thu nhập lớn nhất: 39.1
số hồ sơ trễ từ 2 lần: 10
năm mã đầu tiên: ['HS05', 'HS09', 'HS10', 'HS15', 'HS18']
```

Đọc kết quả: bảng có 40 dòng và 6 cột. Mỗi dòng là một từ điển (dictionary): tên cột đi với giá trị. Thu nhập hồ sơ thứ 5 là 14,2 nhưng kiểu dữ liệu là chuỗi chữ (str), nên phải đổi bằng float trước khi tính. Thu nhập cao nhất là 39,1 triệu đồng/tháng, và 10 hồ sơ trễ từ 2 lần.

*Reading the output: the table has 40 rows and 6 columns. Each row is a dictionary: column names paired with values. The income of file 5 is 14.2 but its type is a string (str), so it must be converted with float before calculating. The highest income is 39.1 million dong per month, and 10 files are late twice or more.*

### Thống kê mô tả và biểu đồ bằng chữ – *Descriptive statistics and a text chart*

**Thống kê mô tả** (descriptive statistics) tóm một cột số bằng vài con số: trung bình, trung vị (giá trị đứng giữa), độ lệch chuẩn (độ dàn trải quanh trung bình), nhỏ nhất, lớn nhất, và ba tứ phân vị (cắt dữ liệu thành bốn phần bằng nhau). Biểu đồ cột bằng chữ ở mã tiếp theo thay cho đồ thị: mỗi dấu # là một hồ sơ. Khi làm thật, bạn vẽ bằng thư viện matplotlib.

***Descriptive statistics*** *summarise a numeric column with a few numbers: mean, median (the middle value), standard deviation (the spread around the mean), minimum, maximum, and the three quartiles (cutting the data into four equal parts). The text bar chart that follows stands in for a graph: each # is one file. In real work you draw it with the matplotlib library.*

```
import csv, io, random, statistics as st
random.seed(2025)
ghi = io.StringIO()               # chỗ chứa chữ, thay cho một tệp CSV
w = csv.writer(ghi)
w.writerow(["ma", "thu_nhap", "du_no", "tre", "kenh", "vo_no"])
for i in range(1, 41):
    tn = round(random.uniform(8, 40), 1)
    no = round(tn * random.uniform(0.5, 4.0), 1)
    tre = random.choice([0, 0, 0, 1, 1, 2, 3])
    kenh = random.choice(["chi nhánh", "trực tuyến", "đại lý"])
    p = 0.03 + 0.07 * tre + 0.02 * no / tn
    vo = 1 if random.random() < p else 0
    w.writerow([f"HS{i:02d}", tn, no, tre, kenh, vo])
ho_so = list(csv.DictReader(io.StringIO(ghi.getvalue())))
tn = [float(r["thu_nhap"]) for r in ho_so]
print("trung bình:", round(st.mean(tn), 2))
print("trung vị:", round(st.median(tn), 2))
print("độ lệch chuẩn:", round(st.stdev(tn), 2))
print("nhỏ nhất, lớn nhất:", min(tn), max(tn))
q = st.quantiles(tn, n=4)
print("tứ phân vị:", [round(v, 2) for v in q])
canh = [8, 16, 24, 32, 41]
for a, b in zip(canh, canh[1:]):
    dem = sum(1 for v in tn if a <= v < b)
    print(f"{a:>2} đến {b - 1:<2} | {'#' * dem} {dem}")
```

### Kết quả chạy / *Output*

```
trung bình: 23.07
trung vị: 23.75
độ lệch chuẩn: 9.43
nhỏ nhất, lớn nhất: 9.0 39.1
tứ phân vị: [14.3, 23.75, 28.52]
 8 đến 15 | ############# 13
16 đến 23 | ####### 7
24 đến 31 | ############ 12
32 đến 40 | ######## 8
```

Đọc kết quả: thu nhập trung bình 23,07 triệu đồng, trung vị 23,75, độ lệch chuẩn 9,43. Trung bình và trung vị gần nhau nên phân phối không lệch nhiều. Bốn cột có 13, 7, 12, 8 hồ sơ: số hồ sơ ở nhóm thu nhập thấp nhất là đông nhất.

*Reading the output: mean income is 23.07 million dong, median 23.75, standard deviation 9.43. Mean and median are close, so the distribution is not strongly skewed. The four bars hold 13, 7, 12, 8 files: the lowest income band has the most files.*

### Bảng tần số và tóm tắt theo nhóm – *Frequency tables and summaries by group*

**Bảng tần số** (frequency table) đếm mỗi giá trị của biến định tính xuất hiện bao nhiêu lần. **Tóm tắt theo nhóm** (group summary) tách bảng theo một biến định tính rồi tính trung bình, tỷ lệ cho từng nhóm. Đó là hai việc nhiều người làm nhất khi mở một bảng dữ liệu lần đầu.

*A **frequency table** counts how many times each value of a qualitative variable appears. A **group summary** splits the table by a qualitative variable and computes means and proportions for each group. These are the two jobs people do most when they first open a data table.*

```
import csv, io, random, statistics as st
random.seed(2025)
ghi = io.StringIO()                # chỗ chứa chữ, thay cho một tệp CSV
w = csv.writer(ghi)
w.writerow(["ma", "thu_nhap", "du_no", "tre", "kenh", "vo_no"])
for i in range(1, 41):
    tn = round(random.uniform(8, 40), 1)
    no = round(tn * random.uniform(0.5, 4.0), 1)
    tre = random.choice([0, 0, 0, 1, 1, 2, 3])
    kenh = random.choice(["chi nhánh", "trực tuyến", "đại lý"])
    p = 0.03 + 0.07 * tre + 0.02 * no / tn
    vo = 1 if random.random() < p else 0
    w.writerow([f"HS{i:02d}", tn, no, tre, kenh, vo])
ho_so = list(csv.DictReader(io.StringIO(ghi.getvalue())))
from collections import Counter
dem = Counter(r["kenh"] for r in ho_so)
for k, v in sorted(dem.items()):
    print(f"{k:<11}{v:>3} hồ sơ {v / len(ho_so):>7.1%}")
nhom = {}
for r in ho_so:
    nhom.setdefault(r["kenh"], []).append(r)
print("kênh, số hồ sơ, thu nhập TB, tỷ lệ vỡ nợ")
for k in sorted(nhom):
    ds = nhom[k]
    tb = st.mean(float(r["thu_nhap"]) for r in ds)
    vo = sum(int(r["vo_no"]) for r in ds) / len(ds)
    print(f"{k:<11}{len(ds):>3}{tb:>8.2f}{vo:>8.1%}")
tre = Counter(int(r["tre"]) for r in ho_so)
print("số lần trễ, số hồ sơ:", sorted(tre.items()))
print("tổng hồ sơ vỡ nợ:", sum(int(r["vo_no"]) for r in ho_so))
```

### Kết quả chạy / *Output*

```
chi nhánh   15 hồ sơ   37.5%
trực tuyến  16 hồ sơ   40.0%
đại lý       9 hồ sơ   22.5%
kênh, số hồ sơ, thu nhập TB, tỷ lệ vỡ nợ
chi nhánh   15   20.83    0.0%
trực tuyến  16   24.43    6.2%
đại lý       9   24.39   22.2%
số lần trễ, số hồ sơ: [(0, 17), (1, 13), (2, 5), (3, 5)]
tổng hồ sơ vỡ nợ: 3
```

Đọc kết quả: cả 40 hồ sơ có 3 hồ sơ vỡ nợ. Kênh đại lý có 2 hồ sơ vỡ nợ trong 9 hồ sơ (22,2%), còn kênh chi nhánh có 0 trong 15 (0,0%). Mẫu chỉ 40 hồ sơ nên khác biệt này có thể do ngẫu nhiên, chưa đủ để kết luận kênh nào rủi ro hơn.

*Reading the output: of all 40 files, 3 defaulted. The đại lý channel has 2 defaults among 9 files (22.2%), while the chi nhánh channel has 0 among 15 (0.0%). With only 40 files this gap may be chance, and it is not enough to say which channel is riskier.*

Để dịch tên kênh: chi nhánh là branch, trực tuyến là online, đại lý là agent.

*To translate the channel names: chi nhánh is branch, trực tuyến is online, đại lý is agent.*

### Chia tập huấn luyện – kiểm tra và mô hình cơ sở – *Splitting training and test sets, and a baseline*

Mã cuối xáo trộn 40 hồ sơ, giữ 30 hồ sơ để huấn luyện và 10 hồ sơ để kiểm tra. Nhiệm vụ là dự báo dư nợ từ thu nhập. Mô hình cơ sở luôn đoán dư nợ trung bình của tập huấn luyện. Mô hình so sánh là đường thẳng dựng trên tập huấn luyện bằng công thức ở mục 1.4. Cả hai được chấm bằng MSE trên tập kiểm tra.

*The last code shuffles the 40 files, keeps 30 for training and 10 for testing. The task is to predict debt from income. The baseline always guesses the training-set average debt. The model for comparison is a straight line built on the training set with the formula of section 1.4. Both are scored by MSE on the test set.*

```
import csv, io, random, statistics as st
random.seed(2025)
ghi = io.StringIO()               # chỗ chứa chữ, thay cho một tệp CSV
w = csv.writer(ghi)
w.writerow(["ma", "thu_nhap", "du_no", "tre", "kenh", "vo_no"])
for i in range(1, 41):
    tn = round(random.uniform(8, 40), 1)
    no = round(tn * random.uniform(0.5, 4.0), 1)
    tre = random.choice([0, 0, 0, 1, 1, 2, 3])
    kenh = random.choice(["chi nhánh", "trực tuyến", "đại lý"])
    p = 0.03 + 0.07 * tre + 0.02 * no / tn
    vo = 1 if random.random() < p else 0
    w.writerow([f"HS{i:02d}", tn, no, tre, kenh, vo])
ho_so = list(csv.DictReader(io.StringIO(ghi.getvalue())))
random.seed(7)
ds = ho_so[:]
random.shuffle(ds)
huan_luyen, kiem_tra = ds[:30], ds[30:]



def cot(rs, ten):
    return [float(r[ten]) for r in rs]
```

```
xh, yh = cot(huan_luyen, "thu_nhap"), cot(huan_luyen, "du_no")
xk, yk = cot(kiem_tra, "thu_nhap"), cot(kiem_tra, "du_no")
xb, yb = st.mean(xh), st.mean(yh)
sxy = sum((a - xb) * (b - yb) for a, b in zip(xh, yh))
b1 = sxy / sum((a - xb) ** 2 for a in xh)
b0 = yb - b1 * xb



def mse(y, du_bao):
    return sum((a - b) ** 2 for a, b in zip(y, du_bao)) / len(y)



print(f"đường thẳng: du_no = {b0:.2f} + {b1:.2f} * thu_nhap")
print("MSE huấn luyện, mô hình:",
      round(mse(yh, [b0 + b1 * a for a in xh]), 1))
print("MSE kiểm tra, cơ sở    :", round(mse(yk, [yb] * len(yk)), 1))
print("MSE kiểm tra, mô hình  :",
      round(mse(yk, [b0 + b1 * a for a in xk]), 1))
```

### Kết quả chạy / *Output*

```
đường thẳng: du_no = -21.23 + 3.20 * thu_nhap
MSE huấn luyện, mô hình: 686.9
MSE kiểm tra, cơ sở    : 883.7
MSE kiểm tra, mô hình  : 478.8
```

Đọc kết quả: trên tập kiểm tra, mô hình đường thẳng đạt MSE 478,8, mô hình cơ sở đạt 883,7; MSE của mô hình thấp hơn mô hình cơ sở. MSE huấn luyện của đường thẳng là 686,9. Giới hạn: tập kiểm tra chỉ có 10 hồ sơ, và chỉ một lần chia; chia lại bằng hạt giống khác, con số sẽ đổi. Chương 5 dạy cách làm cho phép đo ổn định hơn.

*Reading the output: on the test set the straight-line model scores an MSE of 478.8 and the baseline scores 883.7; the model's MSE is lower than the baseline's. The line's training MSE is 686.9. Limit: the test set has only 10 files and there was only one split; with another seed the numbers would change. Chapter 5 teaches how to make the measurement steadier.*

**LỖI HAY GẶP / COMMON MISTAKE –** Chọn k hay các tham số khác bằng cách nhìn điểm trên tập kiểm tra rồi thử lại nhiều lần. Làm vậy tập kiểm tra không còn niêm phong, và điểm cuối cùng trở nên lạc quan. Chọn tham số trên tập huấn luyện (Chương 5), tập kiểm tra chỉ chấm một lần.

*Choosing k or other parameters by looking at the test score and trying again many times. That unseals the test set, and the final score turns optimistic. Choose parameters on the training set (Chapter 5) and score the test set once.*

### Bảng thuật ngữ – *Glossary*

| **Thuật ngữ**<br>*Term (Vietnamese)* | **Tiếng Anh**<br>*English* | **Giải thích ngắn**<br>*Short definition* |
|---|---|---|
| học máy | machine learning (ML) | Cho máy xem nhiều ví dụ để tự rút quy tắc, thay vì người viết từng quy tắc.<br>*Letting a computer find rules from many examples instead of having people write each rule.* |

| **Thuật ngữ**<br>*Term (Vietnamese)* | **Tiếng Anh**<br>*English* | **Giải thích ngắn**<br>*Short definition* |
|---|---|---|
| học thống kê | statistical learning | Học máy nhìn từ phía thống kê, chú trọng sai số và việc giải thích.<br>*Machine learning seen from statistics, with weight on error and explanation.* |
| trí tuệ nhân tạo | artificial intelligence (AI) | Mọi cách làm cho máy hành xử thông minh; học máy là một nhánh.<br>*Any way of making a machine act intelligently; machine learning is one branch.* |
| biến đầu vào | predictor (input, feature) | Thông tin dùng để dự báo, như thu nhập hay số lần trễ hạn.<br>*Information used to predict, such as income or number of late payments.* |
| biến đầu ra | response (output) | Điều ta muốn dự báo hoặc hiểu, như số khách mới hay vỡ nợ.<br>*What we want to predict or understand, such as new customers or default.* |
| sai số không giảm được | irreducible error | Phần lệch do ε, còn lại dù hàm f được ước lượng hoàn hảo.<br>*The part due to ε that remains even with a perfect estimate of f.* |
| sai số giảm được | reducible error | Phần lệch do ước lượng f chưa tốt, có thể giảm bằng mô hình hoặc dữ liệu tốt hơn.<br>*Error from estimating f poorly, which better models or data can reduce.* |
| dự báo | prediction | Dùng mô hình để đoán đầu ra của trường hợp mới.<br>*Using a model to guess the output for a new case.* |
| suy diễn | inference | Dùng mô hình để hiểu biến nào ảnh hưởng đầu ra và ảnh hưởng thế nào.<br>*Using a model to understand which variables affect the output and how.* |
| phương pháp có tham số | parametric method | Chọn trước dạng hàm rồi chỉ tìm vài tham số.<br>*Fix the form of the function first and then find a few parameters.* |
| phương pháp phi tham số | non-parametric method | Không chọn trước dạng hàm; để dữ liệu quyết định hình dạng.<br>*Assume no form for the function; let the data decide its shape.* |

| **Thuật ngữ**<br>*Term (Vietnamese)* | **Tiếng Anh**<br>*English* | **Giải thích ngắn**<br>*Short definition* |
|---|---|---|
| mức mềm dẻo | flexibility | Mô hình uốn được bao nhiêu hình dạng khác nhau.<br>*How many different shapes a model can bend into.* |
| học có giám sát | supervised learning | Học khi mỗi quan sát có sẵn đáp án đầu ra.<br>*Learning when every observation comes with a known output.* |
| học không giám sát | unsupervised learning | Học khi chỉ có đầu vào, nhằm tìm cấu trúc như các nhóm.<br>*Learning with inputs only, to find structure such as groups.* |
| học bán giám sát | semi-supervised learning | Học khi chỉ một phần nhỏ quan sát có đáp án.<br>*Learning when only a small share of observations has an answer.* |
| hồi quy | regression | Bài toán có đầu ra là số đo.<br>*A problem whose output is a measured number.* |
| phân lớp | classification | Bài toán có đầu ra là nhóm hoặc nhãn.<br>*A problem whose output is a group or label.* |
| tập huấn luyện | training set | Phần dữ liệu dùng để dạy mô hình.<br>*The part of the data used to teach the model.* |
| tập kiểm tra | test set | Phần dữ liệu để dành, chỉ dùng chấm điểm mô hình.<br>*The part of the data held back and used only to score the model.* |
| quá khớp | overfitting | Mô hình học thuộc cả nhiễu của tập huấn luyện nên dự báo kém trên dữ liệu mới.<br>*A model memorises even the noise in the training set and predicts poorly on new data.* |
| mô hình cơ sở | baseline model | Cách dự báo đơn giản nhất, dùng làm mốc so sánh.<br>*The simplest way to predict, used as a yardstick.* |
| rò rỉ dữ liệu | data leakage | Thông tin của tập kiểm tra hoặc của tương lai lọt vào lúc huấn luyện.<br>*Information from the test set or the future slipping into training.* |
| thiên lệch dữ liệu | data bias | Dữ liệu không đại diện đúng cho nhóm người cần dự báo.<br>*Data that do not fairly represent the people to be predicted.* |

| Thuật ngữ<br>*Term (Vietnamese)* | Tiếng Anh<br>*English* | Giải thích ngắn<br>*Short definition* |
|---|---|---|
| k láng giềng gần nhất | k-nearest neighbours (KNN) | Dự báo bằng trung bình đầu ra của k điểm gần nhất.<br>*Predicting with the average output of the k closest points.* |

## 1.12. Bài tập vận dụng có lời giải – *Applied exercises with solutions*

### Bài 1. Đường thẳng và trung bình láng giềng – *Exercise 1. A straight line and a neighbour average*

Dữ kiện đề bài: Hà My ghi chi quảng cáo x (triệu đồng/tháng) và số khách mới y (trăm khách) của 6 tháng: (1; 3), (2; 5), (3; 3), (4; 7), (5; 7), (6; 8). Bạn dự báo y tại x = 4,4 bằng hai cách: đường thẳng dựng từ dữ liệu và trung bình của 3 điểm gần nhất. Sau đó tính sai số trung bình bình phương của đường thẳng trên 6 tháng này.

*Given data: Ha My records advertising spend x (million dong per month) and new customers y (hundreds) for 6 months: (1, 3), (2, 5), (3, 3), (4, 7), (5, 7), (6, 8). You predict y at x = 4.4 in two ways: a straight line built from the data and the average of the 3 nearest points. Then compute the mean squared error of the line on these 6 months.*

1. Trung bình: $\bar{x} = 21 / 6 = 3{,}5$ và $\bar{y} = 33 / 6 = 5{,}5$.
   *Means: $\bar{x} = 21 / 6 = 3.5$ and $\bar{y} = 33 / 6 = 5.5$.*
2. Tổng tích độ lệch = 6,25 + 0,75 + 1,25 + 0,75 + 2,25 + 6,25 = 17,5. Tổng bình phương độ lệch của x = 17,5.
   *Sum of products of deviations = 6.25 + 0.75 + 1.25 + 0.75 + 2.25 + 6.25 = 17.5. Sum of squared deviations of x = 17.5.*
3. Độ dốc = 17,5 / 17,5 = 1. Hệ số chặn = $5{,}5 - 1 * 3{,}5 = 2$. Đường thẳng: $\hat{y} = 2 + x$.
   *Slope = 17.5 / 17.5 = 1. Intercept = $5.5 - 1 * 3.5 = 2$. The line is $\hat{y} = 2 + x$.*
4. Dự báo tại 4,4: $\hat{y} = 2 + 4{,}4 = 6{,}4$ trăm khách.
   *Prediction at 4.4: $\hat{y} = 2 + 4.4 = 6.4$ hundred customers.*
5. Ba điểm gần 4,4 nhất là x = 4 (cách 0,4), x = 5 (cách 0,6), x = 3 (cách 1,4); y tương ứng 7, 7, 3. Trung bình = 17 / 3 ≈ 5,67.
   *The three points nearest 4.4 are x = 4 (distance 0.4), x = 5 (0.6) and x = 3 (1.4), with y = 7, 7, 3. Average = 17 / 3 ≈ 5.67.*
6. Phần dư $y - \hat{y}$ lần lượt 0, 1, −2, 1, 0, 0. Bình phương: 0, 1, 4, 1, 0, 0, tổng 6. MSE = 6 / 6 = 1.
   *Residuals $y - \hat{y}$ are 0, 1, −2, 1, 0, 0. Squares: 0, 1, 4, 1, 0, 0, total 6. MSE = 6 / 6 = 1.*
7. Đọc kết quả: hai cách lệch nhau 0,73 trăm khách tại x = 4,4. Đường thẳng có 2 tham số; trung bình láng giềng không có tham số nào về dạng hàm.
   *Reading: the two methods differ by 0.73 hundred customers at x = 4.4. The line has 2 parameters; the neighbour average assumes no form for the function.*

### Bài 2. Tỷ lệ lỗi và mô hình cơ sở – *Exercise 2. Error rate and the baseline model*

Dữ kiện đề bài: mô hình của Lúa Vàng báo vỡ nợ (1) hay không (0) cho 8 hồ sơ kiểm tra. Thực tế: 1, 0, 0, 1, 0, 0, 0, 1. Mô hình báo: 1, 0, 1, 0, 0, 0, 0, 1. Mô hình cơ sở luôn báo lớp phổ biến nhất. Tính tỷ lệ lỗi của hai mô hình.

*Given data: Lua Vang's model flags default (1) or not (0) for 8 test applications. Actual: 1, 0, 0, 1, 0, 0, 0, 1. Model output: 1, 0, 1, 0, 0, 0, 0, 1. The baseline always predicts the most common class. Compute both error rates.*

1. So từng cặp: lệch ở hồ sơ thứ 3 (thực 0, báo 1) và hồ sơ thứ 4 (thực 1, báo 0). Có 2 lỗi.
   *Compare each pair: they differ at application 3 (actual 0, flagged 1) and application 4 (actual 1, flagged 0). That is 2 errors.*
2. Tỷ lệ lỗi của mô hình = 2 / 8 = 25%.
   *The model's error rate = 2 / 8 = 25%.*
3. Lớp phổ biến nhất là 0 (5 hồ sơ). Mô hình cơ sở luôn báo 0 nên sai ở 3 hồ sơ vỡ nợ: 3 / 8 = 37,5%.
   *The most common class is 0 (5 applications). The baseline always says 0, so it errs on the 3 defaults: 3 / 8 = 37.5%.*
4. Chênh lệch: 37,5% − 25% = 12,5 điểm phần trăm. Mô hình tốt hơn cơ sở, nhưng 8 hồ sơ là quá ít để kết luận chắc.
   *Difference: 37.5% − 25% = 12.5 percentage points. The model beats the baseline, but 8 applications are too few to be sure.*

### Bài 3. Tập kiểm tra trung thực – *Exercise 3. An honest test set*

Dữ kiện đề bài: chi tiêu thẻ hằng tháng của 8 khách (trăm nghìn đồng). Tập huấn luyện: 6, 8, 10, 12, 14. Tập kiểm tra: 9, 13, 17. Mô hình cơ sở dự báo bằng trung bình tập huấn luyện. Một mô hình khác dự báo 10, 12, 16 cho 3 khách kiểm tra. Tính MSE kiểm tra của cả hai.

*Given data: monthly card spending of 8 customers (hundreds of thousands of dong). Training set: 6, 8, 10, 12, 14. Test set: 9, 13, 17. The baseline predicts the training mean. Another model predicts 10, 12, 16 for the 3 test customers. Compute the test MSE of both.*

1. Trung bình tập huấn luyện = 50 / 5 = 10. Cơ sở dự báo 10 cho cả ba khách kiểm tra.
   *Training mean = 50 / 5 = 10. The baseline predicts 10 for all three test customers.*
2. Sai số cơ sở: 9 − 10 = −1; 13 − 10 = 3; 17 − 10 = 7. Bình phương: 1, 9, 49; tổng 59. MSE = 59 / 3 ≈ 19,67.
   *Baseline errors: 9 − 10 = −1; 13 − 10 = 3; 17 − 10 = 7. Squares: 1, 9, 49; total 59. MSE = 59 / 3 ≈ 19.67.*
3. Sai số mô hình: 9 − 10 = −1; 13 − 12 = 1; 17 − 16 = 1. Bình phương: 1, 1, 1; tổng 3. MSE = 3 / 3 = 1.
   *Model errors: 9 − 10 = −1; 13 − 12 = 1; 17 − 16 = 1. Squares: 1, 1, 1; total 3. MSE = 3 / 3 = 1.*
4. Đọc kết quả: trên tập kiểm tra, mô hình đạt 1 còn cơ sở đạt 19,67. Căn bậc hai: 1 và khoảng 4,43 trăm nghìn đồng, tức mô hình lệch trung bình cỡ 100 nghìn đồng.

*Reading: on the test set the model scores 1 and the baseline 19.67. Square roots: 1 and about 4.43 hundred thousand dong, so the model is off by about 100,000 dong on average.*

5. Giới hạn: chỉ 3 khách kiểm tra. Nếu tính MSE trên chính 5 khách huấn luyện, con số sẽ quá đẹp và không nói gì về khách mới.
   *Limit: only 3 test customers. An MSE computed on the 5 training customers would look too good and say nothing about new customers.*

## 1.13. Câu hỏi tự luận và dàn ý trả lời – *Essay questions and answer outlines*

### Câu 1. Phân biệt mục tiêu dự báo và mục tiêu suy diễn. Cho mỗi mục tiêu một ví dụ ở Lúa Vàng và nói loại mô hình bạn ưu tiên.

***Question 1.*** *Distinguish the aim of prediction from the aim of inference. Give one Lua Vang example for each and say which kind of model you would favour.*

- Dự báo: cần con số $\hat{Y}$ chính xác cho trường hợp mới, không cần mở hộp đen.
  *Prediction: you need an accurate $\hat{Y}$ for a new case and do not need to open the black box.*
- Suy diễn: cần hiểu biến nào ảnh hưởng, chiều và mức ảnh hưởng.
  *Inference: you need to know which variables matter, in which direction and by how much.*
- Ví dụ dự báo: số tiền khách rút tại cây ATM ngày mai.
  *Prediction example: the cash customers withdraw at an ATM tomorrow.*
- Ví dụ suy diễn: mỗi lần trễ hạn thêm làm xác suất vỡ nợ đổi bao nhiêu.
  *Inference example: how much each extra late payment changes the default probability.*
- Ưu tiên: suy diễn dùng mô hình đơn giản, đọc được; dự báo thử nhiều mô hình và chọn theo sai số kiểm tra.
  *Preference: inference favours simple, readable models; prediction tries many models and picks by test error.*

### Câu 2. Kể bảy bước của một dự án học máy ở Lúa Vàng, nói ở bước nào hay xảy ra rò rỉ dữ liệu tương lai và cách phòng.

***Question 2.*** *List the seven steps of a machine learning project at Lua Vang, say at which step leakage of future data tends to happen and how to prevent it.*

- Bảy bước: đặt bài toán; thu thập và làm sạch; chia tập; huấn luyện; đánh giá; triển khai; theo dõi.
  *Seven steps: frame the problem; collect and clean; split; train; evaluate; deploy; monitor.*
- Rò rỉ hay xảy ra khi làm sạch hoặc chuẩn hóa bằng thống kê của cả bộ dữ liệu trước khi chia.
  *Leakage tends to happen when cleaning or scaling with statistics of the whole data set before splitting.*
- Phòng: chia trước, mọi phép tính thống kê chỉ lấy từ tập huấn luyện.
  *Prevention: split first and take every statistic from the training set only.*
- Với dữ liệu theo thời gian: tập kiểm tra phải nằm sau tập huấn luyện.
  *With time-ordered data the test set must come after the training set.*

- Nêu một biến chỉ biết sau khi khoản vay kết thúc mà không được đưa vào mô hình duyệt vay.
  *Name one variable known only after a loan ends that must not enter a lending model.*

**Câu 3. Chị Linh lo mô hình chấm điểm tín dụng thiên lệch. Trình bày ba nguồn thiên lệch có thể có và cách kiểm.**
***Question 3.*** *Ms Linh worries that the credit scoring model is biased. Describe three possible sources of bias and how to check them.*

- Dữ liệu chỉ gồm hồ sơ đã được duyệt: mô hình không thấy người bị từ chối.
  *The data hold only approved applications: the model never sees rejected people.*
- Biến đại diện: một biến tưởng vô hại lại phản ánh nhóm người.
  *Proxy variables: a seemingly harmless variable can stand for a group of people.*
- Quy tắc: không dùng giới tính, dân tộc, tôn giáo làm biến dự báo.
  *Rule: do not use gender, ethnicity or religion as predictors.*
- Cách kiểm: so tỷ lệ duyệt và tỷ lệ lỗi giữa các nhóm trên tập kiểm tra.
  *Check: compare approval rates and error rates across groups on the test set.*
- Giữ nhật ký quyết định để giải thích với khách hàng.
  *Keep a decision log so you can explain outcomes to customers.*

## 1.14. Câu hỏi trắc nghiệm – *Multiple-choice questions*

**1.** Trong mô hình $Y = f(X) + \varepsilon$, phần nào KHÔNG thể giảm dù bạn tìm được đúng hàm f?
*In the model $Y = f(X) + \varepsilon$, which part cannot be reduced even if you find the true function f?*

**A.** Sai số do chọn dạng hàm chưa phù hợp / *Error from choosing an unsuitable form of function*

**B.** Sai số do ước lượng hệ số từ quá ít dữ liệu / *Error from estimating coefficients with too little data*

**C.** Sai số ε do các yếu tố nằm ngoài biến đầu vào / *The error ε from factors outside the input variables*

**D.** Sai số do bỏ sót một biến có sẵn trong dữ liệu / *Error from leaving out a variable already in the data*

**2.** Chị Linh hỏi: mỗi lần trễ hạn thêm làm xác suất vỡ nợ đổi bao nhiêu? Đây là mục tiêu nào?
*Ms Linh asks how much each extra late payment changes the default probability. Which aim is this?*

**A.** Suy diễn: cần hiểu mức ảnh hưởng của từng biến / *Inference: you need the effect size of each variable*
**B.** Dự báo: chỉ cần số dự báo cho hồ sơ mới thôi / *Prediction: you only need a forecast for new files*
**C.** Phân cụm: tìm các nhóm khách hàng giống nhau / *Clustering: finding groups of similar customers*
**D.** Bán giám sát: gán nhãn cho hồ sơ chưa có kết quả / *Semi-supervised: labelling files with no outcome yet*

**3.** Mô hình $\hat{y} = 1{,}5 + 0{,}8x$. Với $x = 10$, thực tế $y = 11$. Sai số $y - \hat{y}$ bằng:
*The model is $\hat{y} = 1.5 + 0.8x$. For $x = 10$ the actual $y$ is 11. The error $y - \hat{y}$ equals:*

**A.** 0,5 / *0.5*
**B.** 1,0 / *1.0*
**C.** 9,5 / *9.5*
**D.** 1,5 / *1.5*

**4.** Bài toán nào sau đây là phân lớp?
*Which of these problems is classification?*

**A.** Dự báo số tiền khách rút ở cây ATM ngày mai / *Forecasting the cash withdrawn at an ATM tomorrow*
**B.** Dự báo một giao dịch thẻ có gian lận hay không / *Predicting whether a card transaction is fraudulent*
**C.** Dự báo số ngày trễ hạn của một khoản vay / *Predicting the number of days a loan is overdue*
**D.** Dự báo lợi suất ngày của một rổ cổ phiếu / *Predicting the daily return of a basket of shares*

**5.** Dự báo bằng trung bình 3 láng giềng gần nhất; ba điểm đó có $y = 4$, 6 và 11. Dự báo bằng:
*You predict with the average of the 3 nearest neighbours; those points have $y = 4$, 6 and 11. The prediction is:*

**A.** 6
**B.** 7
**C.** 8
**D.** 4

**6.** Chị Linh cần đọc từng hệ số để giải thích vì sao hồ sơ bị từ chối. Nên ưu tiên mô hình nào?
*Ms Linh must read each coefficient to explain why an application was rejected. Which model should you favour?*

**A.** Mạng nơ-ron sâu có nhiều lớp ẩn / *A deep neural network with many hidden layers*
**B.** Rừng ngẫu nhiên gồm hàng trăm cây / *A random forest of hundreds of trees*
**C.** Boosting với hàng nghìn cây nhỏ / *Boosting with thousands of small trees*
**D.** Hồi quy tuyến tính với ít biến đầu vào / *Linear regression with few input variables*

**7.** Sai số $y - \hat{y}$ trên 4 hồ sơ kiểm tra là 1, −2, 0 và 3. Sai số trung bình bình phương (MSE) bằng:
*The errors $y - \hat{y}$ on 4 test files are 1, −2, 0 and 3. The mean squared error (MSE) equals:*

**A.** 3,5 / *3.5*
**B.** 0,5 / *0.5*
**C.** 14
**D.** 1,75 / *1.75*

**8.** Tập có 200 hồ sơ, trong đó 12 hồ sơ vỡ nợ. Mô hình cơ sở luôn báo "không vỡ nợ" đạt độ chính xác:
*A set has 200 applications and 12 defaulted. A baseline that always says "no default" has an accuracy of:*

**A.** 6%
**B.** 12%
**C.** 94%
**D.** 88%

# Chương 2. Đánh giá độ chính xác của mô hình – *Chapter 2. Assessing model accuracy*

**Mục tiêu chương** – *Chapter objectives*

- Chia dữ liệu thành tập huấn luyện và tập kiểm tra, tính sai số bình phương trung bình (MSE, mean squared error) và đọc ra quá khớp, chưa khớp.
  *Split data into a training set and a test set, compute the mean squared error (MSE), and read off overfitting and underfitting.*
- Giải thích đánh đổi độ chệch (lệch đều một phía) – phương sai (đổi dáng theo bộ dữ liệu) bằng ví dụ bắn cung.
  *Explain the trade-off between bias (off in one direction) and variance (changing shape with the data set) with an archery picture.*
- Nói được bộ phân lớp Bayes (cách đoán tốt nhất khi biết xác suất thật) và tự phân lớp bằng k láng giềng gần nhất (KNN, k-nearest neighbours).
  *State what the Bayes classifier is (the best possible guess when the true probabilities are known) and classify by hand with k-nearest neighbours (KNN).*
- Lập ma trận nhầm lẫn, tính các thước đo của nó và chọn ngưỡng theo chi phí sai lầm.
  *Build a confusion matrix, compute its measures, and choose a threshold by the cost of mistakes.*
- Đọc đường cong ROC (receiver operating characteristic, đường đặc trưng vận hành của bộ thu) và chỉ số AUC (area under the curve); thấy vì sao độ chính xác đánh lừa khi dữ liệu mất cân bằng.
  *Read the ROC curve (receiver operating characteristic) and the AUC (area under the curve); see why accuracy misleads when the classes are imbalanced.*

## 2.1. Vì sao phải đo độ chính xác của mô hình – *2.1. Why we measure a model's accuracy*

Chị Linh hỏi Hà My: "Mô hình em làm đoán có tốt không?" Câu trả lời "sai số nhỏ" chưa có nghĩa gì nếu chưa nói sai số đo trên dữ liệu nào. Hãy nghĩ đến một học sinh học thuộc lòng 50 đề thi cũ. Làm lại đề cũ em được 10 điểm, nhưng vào phòng thi gặp đề mới chỉ được 5 điểm. Điểm trên đề cũ không cho biết em giỏi đến đâu.

*Chị Linh asks Hà My: "Does the model you built predict well?" The answer "the error is small" means nothing until you say which data the error was measured on. Think of a pupil who has learned 50 old exam papers by heart. Redoing an old paper earns 10 out of 10, but a new paper in the exam room earns only 5. The old-paper score does not tell you how good the pupil is.*

Mô hình học máy cũng vậy. Dữ liệu để mô hình học gọi là **tập huấn luyện** (training set). Phần dữ liệu cất riêng, mô hình chưa từng thấy và chỉ dùng để chấm điểm, gọi là **tập kiểm tra** (test set). Điểm trên tập huấn luyện là điểm đề cũ. Điểm trên tập kiểm tra gần với điểm đề mới, và đó mới là điều chị Linh cần biết.

*A machine learning model is the same. The data the model learns from is the **training set**. The data kept aside, never seen by the model and used only for marking, is the **test set**. The score on*

*the training set is the old-paper score. The score on the test set is close to the new-paper score, and that is what Chị Linh needs to know.*

Nhắc lại từ Chương 1: ta tin rằng y = f(X) + ε, trong đó f là quy luật thật mà ta không biết, còn ε là nhiễu ngẫu nhiên. Mô hình sau khi học là một công thức $\hat{f}$ (đọc "f mũ") ước lượng f. Với một quan sát có đầu vào x, giá trị mô hình đoán là $\hat{y} = \hat{f}(x)$, còn giá trị thật là y. Sai số của quan sát đó là $y - \hat{y}$.

*Recall from Chapter 1 that we believe $y = f(X) + \varepsilon$, where f is the true rule we do not know and ε is random noise. The model after learning is a formula $\hat{f}$ (read "f hat") that estimates f. For an observation with input x, the model's guess is $\hat{y} = \hat{f}(x)$ and the true value is y. The error of that observation is $y - \hat{y}$.*

Có hai loại bài toán và cách chấm khác nhau. Khi đoán một con số (hồi quy), ví dụ số khách mới mỗi tuần, ta dùng sai số bình phương trung bình ở mục 2.2. Khi đoán một nhãn (phân lớp), ví dụ hồ sơ vỡ nợ hay không, ta đếm số lần đoán sai và xem kỹ từng kiểu sai ở mục 2.5 đến 2.8.

*There are two kinds of problem and they are marked differently. When we guess a number (regression), such as new customers per week, we use the mean squared error of section 2.2. When we guess a label (classification), such as whether a loan defaults, we count wrong guesses and look closely at each kind of mistake, in sections 2.5 to 2.8.*

| **Loại dữ liệu**<br>*Data set* | **Mô hình có thấy khi học không?**<br>*Does the model see it while learning?* | **Dùng để làm gì**<br>*Purpose* |
|---|---|---|
| Tập huấn luyện<br>*Training set* | Có<br>*Yes* | Tìm hệ số, tức là để mô hình học<br>*Finding the coefficients, i.e. learning* |
| Tập kiểm tra<br>*Test set* | Không<br>*No* | Chấm điểm một lần cuối<br>*Marking once, at the end* |
| Dữ liệu mới khi triển khai<br>*New data in use* | Không (chưa tồn tại lúc học)<br>*No (did not exist at training time)* | Làm việc thật và theo dõi mô hình có xuống cấp không<br>*Real work, and watching whether the model decays* |

Ở Chương 5 bạn sẽ gặp thêm tập kiểm định (validation set) dùng để chọn tham số. Tạm thời nhớ một nguyên tắc: tập kiểm tra không bao giờ được dùng để học hay chỉnh mô hình.

*In Chapter 5 you will meet a validation set, used for choosing tuning parameters. For now remember one rule: the test set is never used to learn or to adjust the model.*

Anh Quân dặn Hà My quy trình bốn bước.

*Anh Quân gave Hà My a four-step routine.*

1. Chia dữ liệu: chọn ngẫu nhiên khoảng 70–80% hàng làm tập huấn luyện, phần còn lại làm tập kiểm tra. Cất tập kiểm tra đi.

*Split the data: choose about 70–80% of the rows at random as the training set and keep the rest as the test set. Put the test set away.*

2. Huấn luyện: cho mô hình học chỉ trên tập huấn luyện, rồi cố định mọi hệ số.
   *Train: let the model learn from the training set only, then freeze every coefficient.*
3. Dự báo: đưa các cột đầu vào của tập kiểm tra vào mô hình và ghi lại ŷ của từng hàng.
   *Predict: feed the input columns of the test set to the model and record ŷ for each row.*
4. Chấm điểm: so ŷ với y thật bằng một thước đo (mục 2.2 cho số, mục 2.7 cho nhãn), rồi đặt cạnh một cách đoán ngây thơ làm mốc (mô hình cơ sở, mục 2.2).
   *Mark: compare ŷ with the true y using one measure (section 2.2 for numbers, section 2.7 for labels), then set it beside a naive guess as a yardstick (the baseline model, section 2.2).*

## 2.2. Sai số bình phương trung bình: huấn luyện và kiểm tra – *2.2. Mean squared error: training and test*

Với bài toán đoán số, thước đo quen thuộc nhất là **sai số bình phương trung bình** (MSE). Cách tính nghe đúng như tên: lấy sai số của từng quan sát, bình phương lên, rồi lấy trung bình. Bình phương để sai số âm và sai số dương không triệt tiêu nhau. Bình phương cũng làm sai số lớn bị phạt nặng hơn sai số nhỏ.

*For guessing numbers, the most familiar measure is the **mean squared error** (MSE). The calculation sounds like its name: take the error of each observation, square it, then average. Squaring stops negative and positive errors from cancelling. It also punishes large errors more than small ones.*

$$\mathrm{MSE} = (1/n)\,\Sigma_{i=1}^{n}\,(y_i - \hat{y}_i)^2 \qquad (2.1)$$

Khi n là số hàng của tập huấn luyện, ta có **MSE huấn luyện**. Khi n là số hàng của tập kiểm tra, ta có **MSE kiểm tra**. Hai con số dùng chung công thức nhưng ý nghĩa khác hẳn: số đầu là điểm đề cũ, số sau là điểm đề mới.

*When n is the number of rows of the training set we get the **training MSE**. When n is the number of rows of the test set we get the **test MSE**. The two numbers share a formula but mean different things: the first is the old-paper score, the second the new-paper score.*

Cần dữ liệu gì: một mô hình đã huấn luyện, cột y thật và các cột đầu vào của tập cần chấm. Các bước tính như sau.

*What you need: a trained model, the true y column, and the input columns of the set being marked. The steps are as follows.*

1. Dùng mô hình đã huấn luyện để tính ŷ cho từng hàng.
   *Use the trained model to compute ŷ for each row.*
2. Tính sai số y − ŷ của từng hàng.
   *Compute the error y − ŷ for each row.*
3. Bình phương từng sai số rồi cộng tất cả lại.
   *Square each error and add them all up.*
4. Chia tổng cho số hàng n. Kết quả có đơn vị là đơn vị của y bình phương.
   *Divide the total by the number of rows n. The result is in units of y squared.*

**VÍ DỤ / EXAMPLE –** Hà My có 5 tuần huấn luyện: chi quảng cáo x (triệu đồng) và số khách mới y. Mô hình đã học là ŷ = 10,5 + 4,25 ∗ x (cách tìm hệ số ở Chương 3). Bảng ngay sau tính sai số từng tuần. Tổng bình phương sai số là 23,5, chia 5 được MSE huấn luyện = 4,7.

*Hà My has 5 training weeks: advertising spend x (million dong) and new customers y. The learned model is ŷ = 10.5 + 4.25 ∗x (how to find the coefficients is in Chapter 3). The table that follows computes each week's error. The squared errors add up to 23.5; dividing by 5 gives a training MSE = 4.7.*

| **Tuần** *Week* | **x** | **y** | **ŷ** | **y − ŷ** | **(y − ŷ)²** |
|---|---|---|---|---|---|
| 1 | 2 | 18 | 19 | −1 | 1 |
| 2 | 4 | 30 | 27,5 | 2,5 | 6,25 |
| 3 | 6 | 33 | 36 | −3 | 9 |
| 4 | 8 | 47 | 44,5 | 2,5 | 6,25 |
| 5 | 10 | 52 | 53 | −1 | 1 |

Giữ nguyên mô hình, Hà My chấm 4 tuần mới mà mô hình chưa thấy. Tổng bình phương sai số là 123,75, chia 4 được MSE kiểm tra = 30,94.

*Keeping the model fixed, Hà My marks 4 new weeks the model has not seen. The squared errors add up to 123.75; dividing by 4 gives a test MSE = 30.94.*

| **Tuần** *Week* | **x** | **y** | **ŷ** | **y − ŷ** | **(y − ŷ)²** |
|---|---|---|---|---|---|
| 1 | 3 | 20 | 23,25 | −3,25 | 10,5625 |
| 2 | 5 | 37 | 31,75 | 5,25 | 27,5625 |
| 3 | 7 | 31 | 40,25 | −9,25 | 85,5625 |
| 4 | 9 | 49 | 48,75 | 0,25 | 0,0625 |

Đọc kết quả: sai số trên tuần mới (30,94) gấp hơn 6 lần sai số trên tuần đã học (4,7). Nếu Hà My chỉ báo "MSE là 4,7" cho chị Linh thì chị sẽ tin mô hình tốt hơn thực tế nhiều.

*Reading the result: the error on new weeks (30.94) is more than 6 times the error on weeks already learned (4.7). If Hà My told Chị Linh only "the MSE is 4.7", Chị Linh would trust the model far more than it deserves.*

Còn một câu hỏi: 30,94 là tốt hay dở? Để trả lời, ta cần một **mô hình cơ sở** (baseline model), tức cách đoán ngây thơ nhất. Với số khách mới, cách đó là luôn đoán trung bình của y trên tập huấn luyện, ở đây 36 khách. Trên 4 tuần kiểm tra, cách đoán này có MSE = 112,75. Mô hình của Hà My (30,94) tốt hơn cơ sở rõ rệt, nên đáng giữ.

*One question remains: is 30.94 good or bad? To answer it we need a **baseline model**, the most naive guess. For new customers that is always guessing the training mean of y, here 36*

*customers. On the 4 test weeks this guess has an MSE = 112.75. Hà My's model (30.94) beats the baseline clearly, so it is worth keeping.*

> **LỖI HAY GẶP / COMMON MISTAKE –** Chấm mô hình trên chính tập huấn luyện rồi báo con số đó cho chị Linh. MSE huấn luyện là điểm đề cũ: nó thấp gần như bất kể mô hình có dùng được hay không.
> *Marking the model on the training set and reporting that number to Chị Linh. The training MSE is the old-paper score: it is low almost whatever the model is worth.*

## 2.3. Quá khớp và chưa khớp: hình chữ U – *2.3. Overfitting and underfitting: the U shape*

Mô hình có càng nhiều "nút vặn" thì càng uốn theo dữ liệu được nhiều. Mức uốn này gọi là **độ mềm dẻo** (flexibility). Đường thẳng chỉ uốn được một kiểu. Đa thức bậc 8 uốn được ở nhiều chỗ.

*The more "knobs" a model has, the more it can bend to the data. This amount of bending is called **flexibility**. A straight line bends in one way only. A polynomial of degree 8 can bend in many places.*

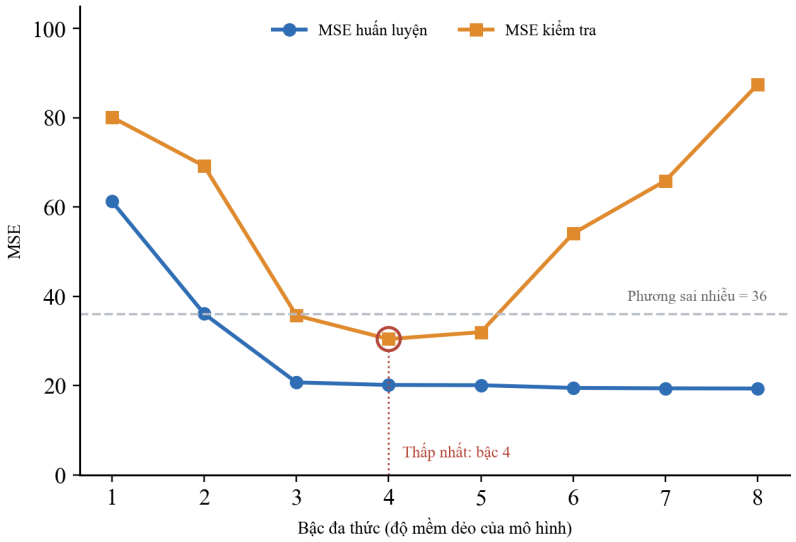
Hà My dựng thử bài toán chi quảng cáo. Biến vào x là chi quảng cáo (triệu đồng mỗi tuần, từ 0 đến 10), biến ra y là số khách mới. Dữ kiện của bài: y = 30 + 22 ∗ sin(x / 1,8) + nhiễu chuẩn có độ lệch chuẩn 6, nên quy luật thật là một đường cong và phương sai nhiễu là 36. Tập huấn luyện có 30 tuần, tập kiểm tra có 60 tuần. Hà My khớp đa thức bậc 1 đến bậc 8 (mã ở mục 2.11).

*Hà My sets up the advertising problem. The input x is advertising spend (million dong per week, from 0 to 10) and the output y is new customers. Problem data: y = 30 + 22 ∗ sin(x / 1.8) + normal noise with standard deviation 6, so the true rule is a curve and the noise variance is 36. The training set has 30 weeks and the test set has 60 weeks. Hà My fits polynomials of degree 1 to 8 (code in section 2.11).*

| **Bậc**<br>*Degree* | **MSE huấn luyện**<br>*Training MSE* | **MSE kiểm tra**<br>*Test MSE* |
|---|---|---|
| 1 | 61,3 | 80,1 |
| 2 | 36,1 | 69,2 |
| 3 | 20,7 | 35,7 |
| 4 | 20,2 | 30,4 |
| 5 | 20,1 | 32,0 |
| 6 | 19,5 | 54,1 |
| 7 | 19,4 | 65,9 |
| 8 | 19,4 | 87,4 |

Đọc bảng: MSE huấn luyện giảm đều từ 61,3 xuống 19,4. MSE kiểm tra giảm đến bậc 4 (còn 30,4, thấp hơn 36 do tập kiểm tra chỉ có 60 tuần nên con số dao động) rồi tăng trở lại, lên 87,4 ở bậc 8. Mô hình cơ sở có MSE kiểm tra 294,8, nên mọi bậc đều hơn cơ sở.

*Reading the table: the training MSE falls steadily from 61.3 to 19.4. The test MSE falls until degree 4 (down to 30.4, below 36 because the test set has only 60 weeks and the figure wobbles) and then rises again, to 87.4 at degree 8. The baseline has a test MSE of 294.8, so every degree beats the baseline.*




**Hình 2.1.** Sai số trên tập huấn luyện (xanh) giảm mãi khi tăng bậc. Sai số trên tập kiểm tra (cam) giảm đến bậc 4 rồi tăng lại: vòng tròn đỏ đánh dấu bậc nên chọn. Đường xám là phương sai nhiễu 36, mức mà không mô hình nào xuống thấp hơn được về trung bình.

***Figure 2.1.*** *Training error (blue) keeps falling as the degree rises. Test error (orange) falls until degree 4 and then rises again: the red circle marks the degree to choose. The grey line is the noise variance of 36, a level no model can beat on average.*

Hình có hai vùng. Bên trái, mô hình **chưa khớp** (underfitting): nó quá đơn giản nên sai cả trên dữ liệu đã học. Bên phải, mô hình **quá khớp** (overfitting): nó học luôn cả nhiễu của 30 tuần huấn luyện nên MSE huấn luyện thấp mà MSE kiểm tra cao. Điểm thấp nhất của đường kiểm tra là độ mềm dẻo nên chọn.

*The figure has two zones. On the left the model **underfits**: it is too simple and is wrong even on the data it has learned. On the right it **overfits**: it learns the noise of the 30 training weeks as well, so the training MSE is low and the test MSE is high. The lowest point of the test curve is the flexibility to choose.*

| **Dấu hiệu**<br>*Sign* | **Chưa khớp**<br>*Underfitting* | **Quá khớp**<br>*Overfitting* |
|---|---|---|
| MSE huấn luyện<br>*Training MSE* | Cao<br>*High* | Thấp<br>*Low* |
| MSE kiểm tra<br>*Test MSE* | Cao, gần MSE huấn luyện<br>*High, close to the training MSE* | Cao hơn MSE huấn luyện nhiều<br>*Much higher than the training MSE* |

| **Dấu hiệu**<br>*Sign* | **Chưa khớp**<br>*Underfitting* | **Quá khớp**<br>*Overfitting* |
|---|---|---|
| Ví dụ ở bảng<br>*Example in the table* | Bậc 1 và 2<br>*Degrees 1 and 2* | Bậc 6 đến 8<br>*Degrees 6 to 8* |
| Cách xử lý<br>*Remedy* | Tăng độ mềm dẻo, thêm biến<br>*Add flexibility, add variables* | Giảm độ mềm dẻo, thêm dữ liệu, chính quy hóa, tức phạt mô hình quá phức tạp (Chương 6)<br>*Reduce flexibility, add data, regularise, i.e. penalise over-complex models (Chapter 6)* |

Chọn độ mềm dẻo có một lỗ hổng: nếu chọn bậc bằng chính tập kiểm tra thì tập này đã bị "nhìn lén" và con số cuối không còn trung thực. Chương 5 sẽ sửa bằng kiểm định chéo (cross-validation), tức chia tập huấn luyện thành nhiều phần để chọn mức mềm dẻo mà không đụng tập kiểm tra.

*Choosing the flexibility has a loophole: if the degree is chosen with the test set itself, that set has been peeked at and the final number is no longer honest. Chapter 5 fixes this with cross-validation, which splits the training set into several parts to choose the flexibility without touching the test set.*

> **LỖI HAY GẶP / COMMON MISTAKE –** Tưởng MSE huấn luyện thấp nghĩa là mô hình đúng. Ở bảng trên, bậc 8 có MSE huấn luyện thấp nhất mà MSE kiểm tra lại cao hơn bậc 4 gấp gần 3 lần.
> *Believing a low training MSE means the model is right. In the table above degree 8 has the lowest training MSE, yet its test MSE is nearly 3 times that of degree 4.*

Các ví dụ ở đây nhằm học cách đo mô hình, không nhằm dự báo giá cổ phiếu hay lãi đầu tư.

*The examples here are for learning how to measure a model; they are not meant to forecast share prices or investment returns.*

## 2.4. Đánh đổi độ chệch – phương sai – *2.4. The bias–variance trade-off*

Hình chữ U có một lời giải thích gọn. Hình dung Hà My bắn cung năm lần. Mỗi lần cô dùng một bộ dữ liệu huấn luyện khác để huấn luyện mô hình, và hồng tâm là giá trị thật $f(x_0)$ tại một điểm $x_0$. Nếu năm mũi tên cụm lại nhưng lệch xa hồng tâm, mô hình có **độ chệch** cao. Nếu năm mũi tên tản rộng quanh hồng tâm, mô hình có **phương sai** cao.

*The U shape has a neat explanation. Picture Hà My shooting five arrows. Each time she trains the model on a different training set, and the bull's-eye is the true value $f(x_0)$ at a point $x_0$. If the five arrows cluster but sit far from the bull's-eye, the model has high **bias**. If the five arrows scatter widely around the bull's-eye, the model has high **variance**.*

Nói chính xác hơn: **độ chệch** (bias) là chênh lệch giữa giá trị $\hat{f}(x_0)$ trung bình (qua nhiều bộ huấn luyện) và $f(x_0)$. Nó xuất hiện khi mô hình quá đơn giản so với thực tế, như vẽ đường thẳng cho một quan hệ cong. **Phương sai** (variance) là mức dao động của $\hat{f}(x_0)$ khi

đổi bộ huấn luyện. Nó cao khi mô hình bám theo nhiễu. Phần còn lại là **sai số không giảm được** (irreducible error) Var(ε), chính là nhiễu.

*More precisely: **bias** is the gap between the average $\hat{f}(x_0)$ (over many training sets) and $f(x_0)$. It appears when the model is too simple for reality, like drawing a straight line through a curved relationship. **Variance** is how much $\hat{f}(x_0)$ swings when the training set changes. It is high when the model chases noise. The remainder is the **irreducible error** Var(ε), which is the noise itself.*

$$E(y_0 - \hat{f}(x_0))^2 = \text{Var}(\hat{f}(x_0)) + [\text{Bias}(\hat{f}(x_0))]^2 + \text{Var}(\varepsilon) \qquad (2.2)$$

Đọc công thức: vế trái là MSE kiểm tra kỳ vọng tại $x_0$, tức MSE ta sẽ thấy nếu lặp lại việc chọn bộ huấn luyện nhiều lần rồi lấy trung bình. Vế phải là ba số không âm cộng lại. Vì vậy MSE kiểm tra không bao giờ thấp hơn Var(ε), và muốn nó nhỏ thì phải có cả phương sai lẫn độ chệch nhỏ.

*Reading the formula: the left side is the expected test MSE at $x_0$, the MSE we would see if we repeated the choice of training set many times and averaged. The right side is three non-negative numbers added together. So the test MSE never drops below Var(ε), and to make it small both the variance and the bias must be small.*

Ví dụ tính tay. Tại một mức chi quảng cáo $x_0$, số khách thật là $f(x_0) = 50$ và nhiễu có phương sai 36. Hà My huấn luyện hai mô hình trên 5 bộ dữ liệu khác nhau và ghi lại giá trị mỗi mô hình đoán tại $x_0$.

*A hand example. At one advertising level $x_0$ the true number of customers is $f(x_0) = 50$ and the noise has variance 36. Hà My trains two models on 5 different data sets and records what each model predicts at $x_0$.*

| **Lần**<br>*Run* | **Mô hình đơn giản $\hat{f}(x_0)$**<br>*Simple model $\hat{f}(x_0)$* | **Mô hình mềm dẻo $\hat{f}(x_0)$**<br>*Flexible model $\hat{f}(x_0)$* |
|---|---|---|
| 1 | 46 | 50 |
| 2 | 53 | 62 |
| 3 | 49 | 41 |
| 4 | 44 | 55 |
| 5 | 48 | 42 |
| Trung bình<br>*Mean* | 48 | 50 |
| Chệch = TB − 50<br>*Bias = mean − 50* | −2 | 0 |
| Chệch²<br>*Bias squared* | 4 | 0 |
| Phương sai<br>*Variance* | 9,2 | 62,8 |

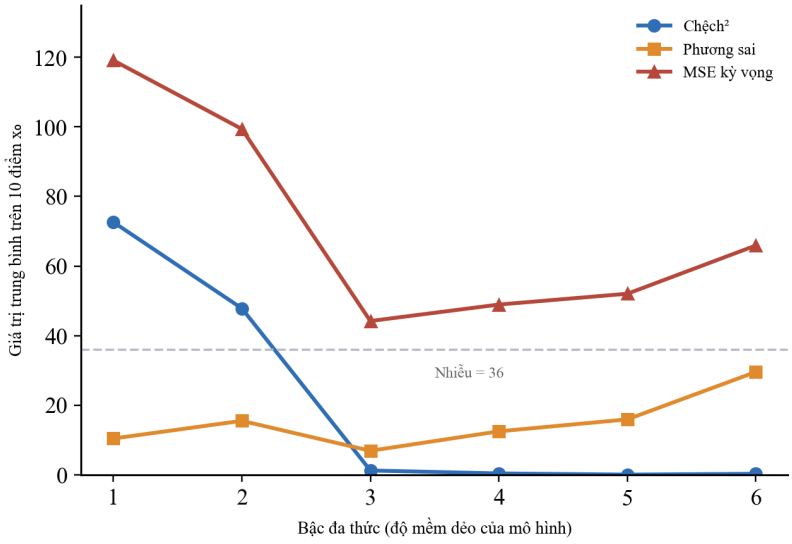
| **Lần**<br>*Run* | **Mô hình đơn giản $\hat{f}(x_0)$**<br>*Simple model $\hat{f}(x_0)$* | **Mô hình mềm dẻo $\hat{f}(x_0)$**<br>*Flexible model $\hat{f}(x_0)$* |
|---|---|---|
| Chệch² + phương sai + 36<br>*Bias² + variance + 36* | 49,2 | 98,8 |

Cách làm: lấy trung bình 5 giá trị, trừ 50 để được độ chệch; lấy trung bình bình phương độ lệch của 5 giá trị quanh trung bình của chúng để được phương sai (chia cho 5); cộng ba phần. Đọc kết quả: mô hình đơn giản chệch −2 nhưng ổn định (phương sai 9,2); mô hình mềm dẻo không chệch nhưng đổi dáng mạnh (phương sai 62,8). Cộng lại, mô hình đơn giản có MSE kỳ vọng 49,2, tốt hơn mô hình mềm dẻo (98,8).

*Method: average the 5 values and subtract 50 to get the bias; average the squared deviations of the 5 values around their own mean to get the variance (divide by 5); add the three parts. Reading the result: the simple model is biased by −2 but stable (variance 9.2); the flexible model is unbiased but changes shape a lot (variance 62.8). Added up, the simple model has an expected MSE of 49.2, better than the flexible model (98.8).*

Làm lại trên dữ liệu chi quảng cáo của mục 2.3: Hà My sinh 100 bộ huấn luyện, mỗi bộ 30 tuần, khớp đa thức bậc 1 đến 6 và đo ba phần tại 10 điểm $x_0$ (mục 2.11). Ở bậc 1, chệch² là 72,6 còn phương sai chỉ 10,5: mô hình chưa khớp. Từ bậc 3, chệch² còn 1,3 hoặc nhỏ hơn, nhưng phương sai tăng dần lên 29,6 ở bậc 6. MSE kỳ vọng thấp nhất ở bậc 3, bằng 44,2.

*Repeating on the advertising data of section 2.3: Hà My generates 100 training sets of 30 weeks each, fits polynomials of degree 1 to 6, and measures the three parts at 10 points $x_0$ (section 2.11). At degree 1 the squared bias is 72.6 while the variance is only 10.5: the model underfits. From degree 3 the squared bias is 1.3 or less, but the variance climbs to 29.6 at degree 6. The expected MSE is lowest at degree 3, at 44.2.*




**Hình 2.2.** Chệch² (xanh) giảm nhanh khi mô hình bớt đơn giản, phương sai (cam) nhìn chung tăng. Tổng của hai đường cộng với nhiễu 36 là MSE kỳ vọng (đỏ), thấp nhất ở bậc 3. Bậc 2 có phương sai cao hơn bậc 3 vì mô hình còn lệch nhiều nên đường cong đổi dáng theo từng bộ dữ

liệu.

***Figure 2.2.*** *Squared bias (blue) falls fast as the model becomes less simple, while variance (orange) generally rises. The two plus the noise of 36 give the expected MSE (red), lowest at degree 3. Degree 2 has higher variance than degree 3 because the model is still biased, so its curve changes shape with each data set.*

| **Tiêu chí**<br>*Criterion* | **Độ chệch cao**<br>*High bias* | **Phương sai cao**<br>*High variance* |
|---|---|---|
| Nguyên nhân thường gặp<br>*Usual cause* | Mô hình quá đơn giản<br>*Model too simple* | Mô hình quá mềm dẻo hoặc quá ít dữ liệu<br>*Model too flexible or too little data* |
| Dấu hiệu<br>*Symptom* | MSE huấn luyện và MSE kiểm tra đều cao<br>*Training and test MSE both high* | MSE huấn luyện thấp, MSE kiểm tra cao<br>*Training MSE low, test MSE high* |
| Thêm dữ liệu<br>*More data* | Ít giúp<br>*Helps little* | Giúp<br>*Helps* |
| Làm mô hình mềm dẻo hơn<br>*Making the model more flexible* | Giúp<br>*Helps* | Làm tệ hơn<br>*Makes it worse* |

## 2.5. Phân lớp: tỷ lệ lỗi và bộ phân lớp Bayes – *2.5. Classification: the error rate and the Bayes classifier*

Khi đoán nhãn, ví dụ hồ sơ vỡ nợ (1) hay trả đủ (0), MSE không còn hợp. Thay vào đó ta đếm số lần đoán sai. **Tỷ lệ lỗi** (error rate) là phần trăm quan sát bị đoán sai. Giống MSE, ta có tỷ lệ lỗi huấn luyện và tỷ lệ lỗi kiểm tra, và con số tỷ lệ lỗi kiểm tra mới là con số đáng báo.

*When we guess a label, such as default (1) or repaid (0), the MSE no longer fits. Instead we count wrong guesses. The **error rate** is the percentage of observations guessed wrongly. Like the MSE, there is a training error rate and a test error rate, and only the test error rate is worth reporting.*

$$\text{Tỷ lệ lỗi} = (1/n)\ \Sigma_{i=1}^{n}\ I(y_i \neq \hat{y}_i) \qquad (2.3)$$

Ở đây I là hàm chỉ báo: I bằng 1 khi điều kiện trong ngoặc đúng (đoán sai) và bằng 0 khi sai (đoán đúng). Cộng các số 0 và 1 chính là đếm số lần đoán sai.

*Here I is an indicator function: I equals 1 when the condition in brackets holds (a wrong guess) and 0 otherwise (a right guess). Adding the 0s and 1s simply counts the wrong guesses.*

Giả sử ta biết hẳn xác suất P(vỡ nợ | x) của mỗi hồ sơ có đầu vào x. Cách đoán tốt nhất là chọn nhãn có xác suất lớn hơn. Cách đoán đó gọi là **bộ phân lớp Bayes** (Bayes classifier). Với hai lớp, nó báo vỡ nợ khi xác suất vượt 0,5.

*Suppose we know exactly the probability P(default | x) of each applicant with input x. The best guess is to pick the label with the larger probability. That rule is called the **Bayes classifier**. With two classes, it flags default when the probability exceeds 0.5.*

$$\hat{y} = \text{lớp j có } P(Y = j \mid X = x_0) \text{ lớn nhất} \quad (2.4)$$

Tập các điểm mà xác suất đúng bằng 0,5 là đường phân chia hai vùng, gọi là **ranh giới Bayes** (Bayes decision boundary). Tỷ lệ lỗi của bộ phân lớp Bayes gọi là **sai số Bayes** (Bayes error rate). Công thức: lấy xác suất của lớp thắng tại mỗi điểm, lấy trung bình, rồi trừ khỏi 1.

*The set of points where the probability is exactly 0.5 is the line separating the two regions, called the **Bayes decision boundary**. The error rate of the Bayes classifier is called the **Bayes error rate**. In a formula: take the probability of the winning class at each point, average it, and subtract from 1.*

$$\text{Sai số Bayes} = 1 - E[\max_j P(Y = j \mid X)] \quad (2.5)$$

Sai số Bayes lớn hơn 0 vì nhãn thật cũng có nhiễu: hai hồ sơ giống hệt nhau vẫn có thể một vỡ nợ, một không. Không phương pháp nào thắng được mức này. Vì ta không biết xác suất thật, bộ phân lớp Bayes chỉ làm thước chuẩn; KNN, hồi quy logistic, đoán xác suất bằng một đường cong hình chữ S (Chương 4) và các phương pháp khác đều cố ước lượng xác suất đó.

*The Bayes error is above 0 because real labels carry noise: two identical applicants can still differ, one defaulting and one not. No method can beat this level. Since we do not know the true probabilities, the Bayes classifier serves only as a yardstick; KNN, logistic regression, which predicts a probability with an S-shaped curve (Chapter 4) and others all try to estimate those probabilities.*

> **VÍ DỤ / EXAMPLE –** Ví dụ tính tay. Lúa Vàng chia khách thành 3 nhóm. Nhóm 1 chiếm 50% hồ sơ, xác suất vỡ nợ 0,15. Nhóm 2 chiếm 30%, xác suất 0,40. Nhóm 3 chiếm 20%, xác suất 0,70. Bộ phân lớp Bayes báo vỡ nợ chỉ ở nhóm 3. Xác suất đoán sai là 0,15; 0,40 và 0,30. Sai số Bayes = $0{,}5 * 0{,}15 + 0{,}3 * 0{,}40 + 0{,}2 * 0{,}30 = 0{,}255$. Cách luôn đoán "trả đủ" sai $0{,}5 * 0{,}15 + 0{,}3 * 0{,}40 + 0{,}2 * 0{,}70 = 0{,}335$. Nghĩa là cứ 1.000 hồ sơ, bộ phân lớp tốt nhất vẫn sai 255 hồ sơ, và cách đoán ngây thơ sai 335.
>
> *A hand example. Lúa Vàng splits customers into 3 groups. Group 1 is 50% of applications, default probability 0.15. Group 2 is 30%, probability 0.40. Group 3 is 20%, probability 0.70. The Bayes classifier flags default only in group 3. The probability of a wrong guess is 0.15, 0.40 and 0.30. Bayes error = $0.5 * 0.15 + 0.3 * 0.40 + 0.2 * 0.30 = 0.255$. Always guessing "repaid" errs with $0.5 * 0.15 + 0.3 * 0.40 + 0.2 * 0.70 = 0.335$. Out of every 1,000 applications, even the best classifier is wrong on 255, and the naive guess on 335.*

Giờ làm trên bộ dữ liệu hồ sơ vay mà Hà My tự sinh. Mỗi hồ sơ có hai đặc trưng chuẩn hóa về đoạn từ 0 đến 1: u là thu nhập (0 là thấp nhất) và v là dư nợ (1 là cao nhất). Dữ kiện của bài: xác suất vỡ nợ thật theo công thức sau.

*Now work on the loan data set that Hà My generates herself. Each applicant has two features scaled to the range 0 to 1: u is income (0 is the lowest) and v is debt (1 is the highest). Problem data: the true default probability follows this formula.*

$$P(\text{vỡ nợ} \mid u, v) = 1 \,/\, (1 + e^{-(-1 + 6(v - u))}) \quad (2.6)$$

Xác suất bằng 0,5 khi $-1 + 6(v - u) = 0$, tức $v - u = 1/6 \approx 0{,}167$. Đó là ranh giới Bayes: đường thẳng $v = u + 0{,}167$. Tính trên lưới $40 \times 40$ điểm (mục 2.11), tỷ lệ vỡ nợ chung là

37,4% và sai số Bayes là 17,4%. Luôn đoán "không vỡ nợ" sai 37,4%. Nghĩa là dùng thông tin thu nhập và dư nợ giảm lỗi từ 37,4% xuống tối đa 17,4%.

*The probability equals 0.5 when $-1 + 6(v - u) = 0$, that is $v - u = 1/6 \approx 0.167$. That is the Bayes boundary: the straight line $v = u + 0.167$. Computed on a 40 × 40 grid (section 2.11), the overall default rate is 37.4% and the Bayes error is 17.4%. Always guessing "no default" errs 37.4% of the time. So using income and debt cuts the error from 37.4% down to at best 17.4%.*

| **Ranh giới thử $v - u >$ ...**<br>*Trial boundary $v - u >$ ...* | **Tỷ lệ lỗi (%)**<br>*Error rate (%)* |
|---|---|
| −0,2 | 32,8 |
| 0 | 20,6 |
| 0,167 (ranh giới Bayes)<br>*0.167 (Bayes boundary)* | 17,4 |
| 0,4 | 23,0 |
| 0,8 | 35,7 |

Đọc bảng: lỗi thấp nhất đúng ở ranh giới Bayes. Dịch ranh giới sang hai phía đều làm lỗi tăng; ở mức 0,8 gần như không hồ sơ nào bị báo vỡ nợ và lỗi gần bằng cách luôn đoán "không vỡ nợ".

*Reading the table: the error is lowest exactly at the Bayes boundary. Moving the boundary either way raises the error; at 0.8 almost no applicant is flagged and the error is close to always guessing "no default".*

Một lưu ý về công bằng: mô hình tín dụng của Lúa Vàng chỉ dùng biến liên quan đến khả năng trả nợ như thu nhập, dư nợ, số lần trễ hạn. Giới tính, dân tộc, tôn giáo không được dùng làm biến dự báo.

*A word on fairness: Lúa Vàng's credit model uses only variables related to the ability to repay, such as income, debt and number of late payments. Gender, ethnicity and religion are not used as predictors.*

## 2.6. KNN: phân lớp bằng k láng giềng gần nhất – *2.6. KNN: classifying with the k nearest neighbours*

Bộ phân lớp Bayes cần biết xác suất thật, mà ta không biết. **KNN** (k láng giềng gần nhất) ước lượng xác suất bằng cách đời thường: để đoán hồ sơ mới vỡ nợ hay không, hãy hỏi k hồ sơ cũ giống nó nhất và cho họ bỏ phiếu. Nhãn nào đông phiếu hơn thì thắng.

*The Bayes classifier needs the true probabilities, which we do not have. **KNN** (k-nearest neighbours) estimates them in a everyday way: to guess whether a new applicant defaults, ask the k old applicants most like them and let them vote. The label with more votes wins.*

Cần dữ liệu gì: tập huấn luyện có nhãn, các cột đầu vào là số, và một cách đo "giống nhau" bằng khoảng cách. Cách quen nhất là khoảng cách Euclid, tức độ dài đường thẳng nối hai điểm.

*What you need: a labelled training set, numeric input columns, and a way to measure "alike" by distance. The most familiar is the Euclidean distance, the length of the straight line joining two points.*

$$d = \sqrt{(x_1 - x'_1)^2 + (x_2 - x'_2)^2} \quad (2.7)$$

1. Chọn k, thường là số lẻ để phiếu không hòa (với hai lớp).
   *Choose k, usually odd so that votes do not tie (with two classes).*
2. Nếu các cột có thang đo khác nhau, chuẩn hóa từng cột: trừ trung bình, chia cho độ lệch chuẩn của cột đó.
   *If the columns have different scales, standardise each column: subtract its mean and divide by its standard deviation.*
3. Tính khoảng cách từ hồ sơ mới đến từng hồ sơ trong tập huấn luyện.
   *Compute the distance from the new applicant to every training applicant.*
4. Sắp xếp khoảng cách từ nhỏ đến lớn và lấy k hồ sơ đầu.
   *Sort the distances from small to large and take the first k applicants.*
5. Đếm nhãn trong k hồ sơ đó. Nhãn đa số là dự báo; tỷ lệ phiếu là ước lượng xác suất.
   *Count the labels among those k. The majority label is the prediction; the share of votes is the probability estimate.*

$$\hat{P}(Y = j \mid x_0) = (1/k) \sum_{i \in N_0} I(y_i = j) \quad (2.8)$$

Trong công thức, $N_0$ là nhóm k hồ sơ gần $x_0$ nhất.
*In the formula, $N_0$ is the group of the k applicants nearest to $x_0$.*

**VÍ DỤ / EXAMPLE –** Ví dụ tính tay. Lúa Vàng có 8 hồ sơ cũ với thu nhập (triệu đồng mỗi tháng) và dư nợ (triệu đồng). Hồ sơ mới có thu nhập 16 và dư nợ 20. Khoảng cách từ hồ sơ mới đến hồ sơ 1 là $\sqrt{(16-10)^2 + (20-30)^2} = \sqrt{136} \approx 11{,}66$. Làm tương tự cho cả 8 hồ sơ ta được bảng sau; hai cột đầu có cùng thang đo nên không cần chuẩn hóa.
*A hand example. Lúa Vàng has 8 old applicants with income (million dong per month) and debt (million dong). The new applicant has income 16 and debt 20. The distance to applicant 1 is $\sqrt{(16-10)^2 + (20-30)^2} = \sqrt{136} \approx 11.66$. Doing the same for all 8 gives the table that follows; the first two columns share a scale, so no standardising is needed.*

| **Hồ sơ** *Applicant* | **Thu nhập** *Income* | **Dư nợ** *Debt* | **Nhãn** *Label* | **Khoảng cách** *Distance* | **Hạng** *Rank* |
|---|---|---|---|---|---|
| 1 | 10 | 30 | vỡ nợ *default* | 11,66 | 5 |
| 2 | 12 | 8 | trả đủ *repaid* | 12,65 | 6 |
| 3 | 14 | 26 | vỡ nợ *default* | 6,32 | 2 |
| 4 | 18 | 10 | trả đủ *repaid* | 10,2 | 3 |

| Hồ sơ *Applicant* | Thu nhập *Income* | Dư nợ *Debt* | Nhãn *Label* | Khoảng cách *Distance* | Hạng *Rank* |
|---|---|---|---|---|---|
| 5 | 20 | 24 | vỡ nợ *default* | 5,66 | 1 |
| 6 | 24 | 12 | trả đủ *repaid* | 11,31 | 4 |
| 7 | 26 | 28 | trả đủ *repaid* | 12,81 | 7 |
| 8 | 30 | 6 | trả đủ *repaid* | 19,8 | 8 |

Hạng 1 đến 3 là hồ sơ 5 (vỡ nợ), 3 (vỡ nợ) và 4 (trả đủ). Ta thử bốn giá trị k.

*Ranks 1 to 3 are applicants 5 (default), 3 (default) and 4 (repaid). We try four values of k.*

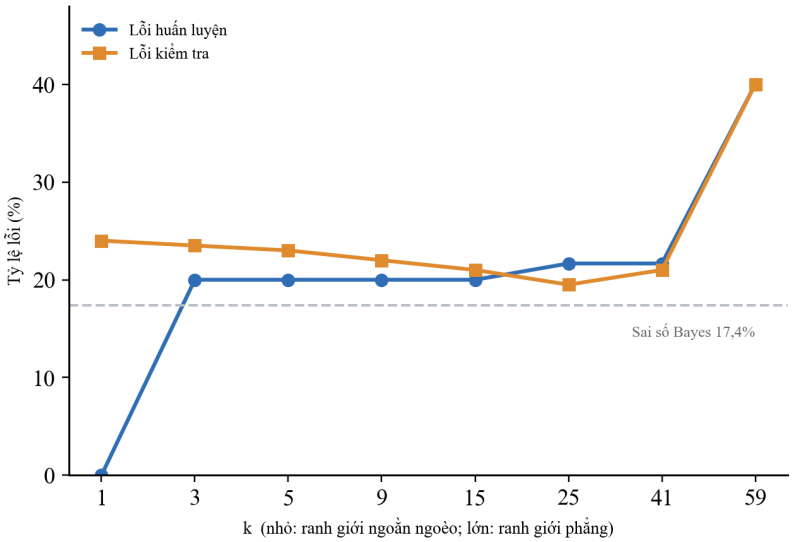
| k | Láng giềng gần nhất *Nearest neighbours* | Số vỡ nợ *Defaults* | Dự báo *Prediction* |
|---|---|---|---|
| 1 | 5 | 1/1 | vỡ nợ *default* |
| 3 | 5, 3, 4 | 2/3 ≈ 0,67 *2/3 ≈ 0.67* | vỡ nợ *default* |
| 5 | 5, 3, 4, 6, 1 | 3/5 = 0,6 *3/5 = 0.6* | vỡ nợ *default* |
| 7 | 5, 3, 4, 6, 1, 2, 7 | 3/7 ≈ 0,43 *3/7 ≈ 0.43* | trả đủ *repaid* |

Đọc bảng: với k = 1, 3, 5 hồ sơ mới bị báo vỡ nợ; với k = 7 kết luận đổi thành trả đủ vì hai hồ sơ ở xa (hồ sơ 2 và 7) kéo phiếu về phía trả đủ. Cùng một hồ sơ, k khác nhau cho kết luận khác nhau, nên k là một lựa chọn phải làm có cơ sở.

*Reading the table: for k = 1, 3, 5 the new applicant is flagged as default; for k = 7 the verdict flips to repaid because two distant applicants (2 and 7) pull the vote to repaid. The same applicant gets different verdicts for different k, so k must be chosen on evidence.*

Giờ xem ảnh hưởng của k trên dữ liệu lớn hơn. Hà My lấy 60 hồ sơ huấn luyện và 200 hồ sơ kiểm tra từ công thức ở mục 2.5, rồi chạy KNN với k từ 1 đến 59 (mã ở mục 2.11). Với k = 1, lỗi huấn luyện bằng 0 vì mỗi điểm là láng giềng gần nhất của chính nó, còn lỗi kiểm tra là 24,0%. Lỗi kiểm tra thấp nhất ở k = 25, bằng 19,5%, gần sai số Bayes 17,4%. Với k = 59, KNN gần như chỉ đoán lớp đông hơn và lỗi kiểm tra lên 40,0%.

*Now look at the effect of k on a larger data set. Hà My takes 60 training applicants and 200 test applicants from the formula of section 2.5, then runs KNN with k from 1 to 59 (code in section 2.11). With k = 1 the training error is 0 because each point is its own nearest neighbour, while the test error is 24.0%. The test error is lowest at k = 25, at 19.5%, close to the Bayes error of 17.4%. With k = 59 KNN almost only guesses the larger class and the test error climbs to 40.0%.*




**Hình 2.3.** Lỗi huấn luyện (xanh) tăng khi k tăng. Lỗi kiểm tra (cam) giảm đến k = 25 rồi tăng vọt ở k = 59. Đường xám là sai số Bayes: lỗi kiểm tra tốt nhất của KNN nằm trên đường này, gần nhưng không thấp hơn.

***Figure 2.3.*** *Training error (blue) rises as k rises. Test error (orange) falls until k = 25 and then jumps at k = 59. The grey line is the Bayes error: the best test error of KNN sits above this line, near it but not below.*

Hình là chữ U của KNN theo chiều ngược: k nhỏ là mềm dẻo (ranh giới ngoằn ngoèo, dễ quá khớp), k lớn là đơn giản (ranh giới gần phẳng, dễ chưa khớp). Chọn k bằng kiểm định chéo trên tập huấn luyện (Chương 5), không chọn bằng tập kiểm tra.

*The figure is the U shape of KNN reversed: small k is flexible (a wiggly boundary, easy to overfit), large k is simple (a nearly flat boundary, easy to underfit). Choose k by cross-validation on the training set (Chapter 5), not with the test set.*

**LỖI HAY GẶP / COMMON MISTAKE –** Quên chuẩn hóa cột trước khi chạy KNN. Nếu dư nợ ghi bằng đồng (10.000.000) còn thu nhập ghi bằng triệu đồng (6), khoảng cách giữa hai hồ sơ là $\sqrt{(6^2 + 10.000.000^2)} \approx 10.000.000$: thu nhập biến mất khỏi phép đo.

*Forgetting to standardise the columns before KNN. If debt is in dong (10,000,000) and income in million dong (6), the distance between two applicants is $\sqrt{(6^2 + 10,000,000^2)} \approx 10,000,000$: income vanishes from the measurement.*

## 2.7. Ma trận nhầm lẫn và các chỉ số phân lớp – *2.7. The confusion matrix and classification measures*

Tỷ lệ lỗi gộp mọi sai lầm làm một. Nhưng với chị Linh, hai sai lầm khác nhau rõ rệt. Cho vay nhầm người sẽ vỡ nợ thì mất tiền gốc. Từ chối nhầm người trả được thì mất khoản lãi. **Ma trận nhầm lẫn** (confusion matrix) là bảng đếm riêng từng loại kết quả để thấy mô hình sai kiểu nào.

*The error rate lumps every mistake together. For Chị Linh, though, the two mistakes are clearly different. Lending to someone who will default loses the principal. Refusing someone who would have repaid loses the interest. The **confusion matrix** is a table that counts each kind of outcome separately so you can see which way the model errs.*

Ta gọi lớp ta quan tâm, vỡ nợ, là lớp "dương". Bốn ô của bảng có tên riêng: TP (true positive, dương thật) là hồ sơ vỡ nợ thật được báo vỡ nợ; FP (false positive, dương giả) là hồ sơ trả đủ bị báo nhầm là vỡ nợ; FN (false negative, âm giả) là hồ sơ vỡ nợ thật bị bỏ sót; TN (true negative, âm thật) là hồ sơ trả đủ được báo đúng là trả đủ.

*We call the class we care about, default, the "positive" class. The four cells have their own names: TP (true positive) is a real defaulter flagged as default; FP (false positive) is a good applicant wrongly flagged as default; FN (false negative) is a real defaulter that was missed; TN (true negative) is a good applicant correctly cleared.*

| **Thực tế**<br>*Reality* | **Mô hình báo vỡ nợ**<br>*Model says default* | **Mô hình báo trả đủ**<br>*Model says repaid* |
|---|---|---|
| Vỡ nợ thật<br>*Truly defaults* | TP | FN |
| Trả đủ thật<br>*Truly repays* | FP | TN |

Từ bốn số này tính được năm chỉ số. Mỗi chỉ số trả lời một câu hỏi, và câu hỏi nào cũng đọc được bằng "cứ 100 hồ sơ …".

*Five measures follow from these four counts. Each answers one question, and each can be read as "out of every 100 applicants …".*

- **Độ chính xác** (accuracy): trong 100 hồ sơ, mô hình đoán đúng bao nhiêu hồ sơ.
  ***Accuracy**: out of 100 applicants, how many the model gets right.*
- **Độ nhạy** (sensitivity, còn gọi recall): trong 100 hồ sơ vỡ nợ thật, mô hình bắt được bao nhiêu.
  ***Sensitivity** (also called recall): out of 100 real defaulters, how many the model catches.*
- **Độ đặc hiệu** (specificity): trong 100 hồ sơ trả đủ, mô hình để yên bao nhiêu.
  ***Specificity**: out of 100 good applicants, how many the model leaves alone.*
- **Độ chuẩn xác** (precision): trong 100 hồ sơ mô hình báo vỡ nợ, bao nhiêu hồ sơ vỡ nợ thật.
  ***Precision**: out of 100 applicants the model flags as default, how many truly default.*
- **F1**: một con số gộp độ nhạy và độ chuẩn xác theo trung bình điều hòa, kéo về phía số nhỏ hơn trong hai số.
  ***F1**: one number combining sensitivity and precision through the harmonic mean, pulled towards the smaller of the two.*

$$\text{Độ chính xác} = (\text{TP} + \text{TN}) / \text{n} \quad (2.9)$$

$$\text{Độ nhạy} = \text{TP} / (\text{TP} + \text{FN}) \quad (2.10)$$

$$\text{Độ đặc hiệu} = \text{TN} / (\text{TN} + \text{FP}) \quad (2.11)$$

$$\text{Độ chuẩn xác} = \text{TP} / (\text{TP} + \text{FP}) \quad (2.12)$$

$$\text{F1} = 2 * \text{Độ chuẩn xác} * \text{Độ nhạy} / (\text{Độ chuẩn xác} + \text{Độ nhạy}) = 2 * \text{TP} / (2 * \text{TP} + \text{FP} + \text{FN}) \quad (2.13)$$

Mô hình chấm điểm tín dụng cho mỗi hồ sơ một xác suất vỡ nợ, và ta tự chọn **ngưỡng** (threshold) c: báo vỡ nợ khi xác suất từ c trở lên. Ngưỡng 0,5 là mặc định, nhưng không có lý do gì nó là ngưỡng tốt nhất. Đổi ngưỡng làm cả bốn ô của ma trận thay đổi.

*A credit scoring model gives each applicant a default probability, and we choose the **threshold** c: flag default when the probability is c or above. A threshold of 0.5 is the default, but there is no reason it is the best one. Changing the threshold changes all four cells of the matrix.*

**VÍ DỤ / EXAMPLE –** Bài mẫu. Lúa Vàng chấm 200 hồ sơ kiểm tra, trong đó 40 hồ sơ vỡ nợ thật và 160 hồ sơ trả đủ. Ở ngưỡng 0,5 mô hình báo vỡ nợ 50 hồ sơ: TP = 28, FP = 22, FN = 12, TN = 138. Độ chính xác = (28 + 138)/200 = 0,83. Độ nhạy = 28/40 = 0,70. Độ đặc hiệu = 138/160 = 0,8625. Độ chuẩn xác = 28/50 = 0,56. F1 = $2 * 28/(2 * 28 + 22 + 12)$ = 0,6222. Nghĩa là cứ 100 hồ sơ mô hình báo vỡ nợ thì 56 hồ sơ vỡ nợ thật, và cứ 100 hồ sơ vỡ nợ thật thì mô hình bắt được 70.

*Worked example. Lúa Vàng marks 200 test applicants, of whom 40 truly default and 160 repay. At threshold 0.5 the model flags 50 as default: TP = 28, FP = 22, FN = 12, TN = 138. Accuracy = (28 + 138)/200 = 0.83. Sensitivity = 28/40 = 0.70. Specificity = 138/160 = 0.8625. Precision = 28/50 = 0.56. F1 = $2 * 28/(2 * 28 + 22 + 12)$ = 0.6222. Out of every 100 applicants the model flags, 56 truly default, and out of every 100 real defaulters the model catches 70.*

Chị Linh muốn bắt nhiều hồ sơ vỡ nợ hơn nên hạ ngưỡng xuống 0,3. Mô hình báo vỡ nợ 80 hồ sơ: TP = 33, FP = 47, FN = 7, TN = 113. Làm lại năm phép tính được độ chính xác 0,73; độ nhạy 0,825; độ đặc hiệu 0,70625; độ chuẩn xác 0,4125; F1 0,55.

*Chị Linh wants to catch more defaulters, so she lowers the threshold to 0.3. The model flags 80 applicants as default: TP = 33, FP = 47, FN = 7, TN = 113. Redoing the five calculations gives accuracy 0.73, sensitivity 0.825, specificity 0.70625, precision 0.4125 and F1 0.55.*

Đọc kết quả: hạ ngưỡng bắt thêm 5 hồ sơ vỡ nợ (độ nhạy 0,70 lên 0,825) nhưng cũng báo nhầm thêm 25 hồ sơ tốt. Không có ngưỡng nào thắng mọi chỉ số; ta phải chọn theo điều công ty sợ hơn, và mục 2.10 làm việc đó bằng chi phí.

*Reading the result: lowering the threshold catches 5 more defaulters (sensitivity 0.70 to 0.825) but also wrongly flags 25 more good applicants. No threshold wins on every measure; we must choose by what the company fears more, and section 2.10 does that with costs.*

Hà My chạy lại với 200 hồ sơ kiểm tra sinh từ công thức ở mục 2.5, dùng xác suất thật làm điểm, ở bốn ngưỡng (mã ở mục 2.11).

*Hà My reruns this on 200 test applicants generated from the formula of section 2.5, using the true probability as the score, at four thresholds (code in section 2.11).*

| **Ngưỡng** *Threshold* | **TP** | **FP** | **FN** | **TN** | **Chính xác** *Acc.* | **Nhạy** *Sens.* | **Đặc hiệu** *Spec.* | **Chuẩn xác** *Prec.* | **F1** |
|---|---|---|---|---|---|---|---|---|---|
| 0,7 | 42 | 6 | 34 | 118 | 0,800 | 0,553 | 0,952 | 0,875 | 0,677 |
| 0,5 | 55 | 16 | 21 | 108 | 0,815 | 0,724 | 0,871 | 0,775 | 0,748 |
| 0,3 | 65 | 32 | 11 | 92 | 0,785 | 0,855 | 0,742 | 0,670 | 0,751 |

| Ngưỡng *Threshold* | TP | FP | FN | TN | Chính xác *Acc.* | Nhạy *Sens.* | Đặc hiệu *Spec.* | Chuẩn xác *Prec.* | F1 |
|---|---|---|---|---|---|---|---|---|---|
| 0,2 | 70 | 45 | 6 | 79 | 0,745 | 0,921 | 0,637 | 0,609 | 0,733 |

Đọc bảng: độ chính xác gần như đứng yên (từ 0,745 đến 0,815) trong khi độ nhạy nhảy từ 0,553 lên 0,921. Độ chuẩn xác đi ngược chiều độ nhạy: từ 0,875 xuống 0,609.

*Reading the table: accuracy barely moves (from 0.745 to 0.815) while sensitivity jumps from 0.553 to 0.921. Precision runs the other way, from 0.875 down to 0.609.*

> **LỖI HAY GẶP / COMMON MISTAKE –** Nhầm độ chính xác (accuracy) với độ chuẩn xác (precision). Một mô hình có thể có độ chính xác 0,83 mà độ chuẩn xác chỉ 0,56, như ví dụ trên.
> *Mixing up accuracy and precision. A model can have an accuracy of 0.83 and a precision of only 0.56, as in the example above.*

## 2.8. Đường ROC và AUC – *2.8. The ROC curve and the AUC*

Mỗi ngưỡng cho một cặp số: độ nhạy và tỷ lệ báo nhầm. Nếu thử mọi ngưỡng rồi nối các cặp lại ta được **đường ROC**. Trục dọc là độ nhạy, còn gọi tỷ lệ dương tính thật. Trục ngang là **tỷ lệ dương tính giả** FP/(FP + TN), tức bằng 1 trừ độ đặc hiệu: phần hồ sơ tốt bị báo nhầm.

*Each threshold gives a pair of numbers: the sensitivity and the false alarm rate. Trying every threshold and joining the pairs gives the **ROC curve**. The vertical axis is sensitivity, also called the true positive rate. The horizontal axis is the **false positive rate** FP/(FP + TN), equal to 1 minus specificity: the share of good applicants wrongly flagged.*

Cách đọc: điểm (0; 0) ứng với ngưỡng cao nhất, mô hình không báo ai. Điểm (1; 1) ứng với ngưỡng thấp nhất, mô hình báo tất cả. Đường chéo là cách đoán ngẫu nhiên. Đường càng gần góc trên bên trái thì mô hình càng tốt, vì bắt nhiều ca thật mà báo nhầm ít.

*How to read it: the point (0, 0) is the highest threshold, where the model flags nobody. The point (1, 1) is the lowest threshold, where it flags everybody. The diagonal is random guessing. The closer the curve gets to the top left corner, the better the model, since it catches many real cases with few false alarms.*

**AUC** là diện tích dưới đường ROC. Giá trị 0,5 là ngẫu nhiên, giá trị 1 là hoàn hảo. Có một cách hiểu đời thường: chọn ngẫu nhiên một hồ sơ vỡ nợ thật và một hồ sơ trả đủ; AUC là xác suất mô hình cho hồ sơ vỡ nợ điểm cao hơn.

*The **AUC** is the area under the ROC curve. A value of 0.5 is random and 1 is perfect. There is an everyday reading: pick one real defaulter and one good applicant at random; the AUC is the probability that the model gives the defaulter the higher score.*

1. Sắp các hồ sơ theo điểm giảm dần.
   *Sort the applicants by score, highest first.*

2. Đặt ngưỡng bằng điểm cao nhất, rồi hạ dần qua từng điểm.
   *Set the threshold at the highest score, then lower it past one score at a time.*
3. Mỗi lần hạ, đếm TP và FP rồi tính độ nhạy = TP/P và tỷ lệ dương tính giả = FP/N (P là số ca dương, N là số ca âm).
   *At each step count TP and FP, then compute sensitivity = TP/P and false positive rate = FP/N (P is the number of positives, N the number of negatives).*
4. Chấm các điểm (tỷ lệ dương tính giả, độ nhạy) và nối lại, bắt đầu từ (0; 0).
   *Plot the points (false positive rate, sensitivity) and join them, starting from (0, 0).*
5. Tính diện tích dưới đường bằng cách cộng các hình thang, hoặc đếm số cặp (dương, âm) xếp đúng thứ tự.
   *Compute the area under the curve by adding trapezoids, or by counting the (positive, negative) pairs placed in the right order.*

**VÍ DỤ / EXAMPLE –** Ví dụ tính tay. Mô hình chấm điểm 8 hồ sơ, trong đó 4 hồ sơ vỡ nợ thật (dương) và 4 hồ sơ trả đủ (âm). Hạ ngưỡng lần lượt qua từng điểm cho Bảng sau. Diện tích dưới đường: các đoạn ngang tại độ nhạy 0,25; 0,75; 0,75 và 1 có độ rộng 0,25 mỗi đoạn, nên AUC = 0,25 ∗ (0,25 + 0,75 + 0,75 + 1) = 0,6875. Đếm cặp cho cùng kết quả: trong 16 cặp (dương, âm), có 11 cặp mà hồ sơ dương điểm cao hơn, và 11/16 = 0,6875.
*A hand example. A model scores 8 applicants, of whom 4 truly default (positive) and 4 repay (negative). Lowering the threshold past each score gives the next table. Area under the curve: the horizontal runs at sensitivity 0.25, 0.75, 0.75 and 1 each have width 0.25, so AUC = 0.25 ∗ (0.25 + 0.75 + 0.75 + 1) = 0.6875. Counting pairs gives the same: of 16 (positive, negative) pairs, 11 have the positive scored higher, and 11/16 = 0.6875.*

| **Lần** *Step* | **Ngưỡng ≥** *Threshold ≥* | **Nhãn hồ sơ vừa vào** *Label just added* | **TP** | **FP** | **Độ nhạy** *Sensitivity* | **Tỷ lệ DT giả** *FPR* |
|---|---|---|---|---|---|---|
| 1 | 0,9 | dương *positive* | 1 | 0 | 0,25 | 0 |
| 2 | 0,8 | âm *negative* | 1 | 1 | 0,25 | 0,25 |
| 3 | 0,7 | dương *positive* | 2 | 1 | 0,5 | 0,25 |
| 4 | 0,6 | dương *positive* | 3 | 1 | 0,75 | 0,25 |
| 5 | 0,5 | âm *negative* | 3 | 2 | 0,75 | 0,5 |
| 6 | 0,4 | âm *negative* | 3 | 3 | 0,75 | 0,75 |
| 7 | 0,35 | dương *positive* | 4 | 3 | 1 | 0,75 |