STAT 24620=FINM 34700, STAT 32950
Multivariate Data Analysis
Lecture 9: Classification

Jingshu Wang

The University of Chicago

Outline

From unsupervised groups to supervised labels

- Lecture 5: clustering looked for groups without labels.
- Lecture 6: mixture models gave a probabilistic model for latent groups.
- Lecture 8: multivariate regression shifted attention toward prediction.

Today: the groups are known in the training data, and we want to predict labels for new observations.

Goal: use features X to predict a class label Y.

A motivating example: default prediction

Suppose each credit card customer has features

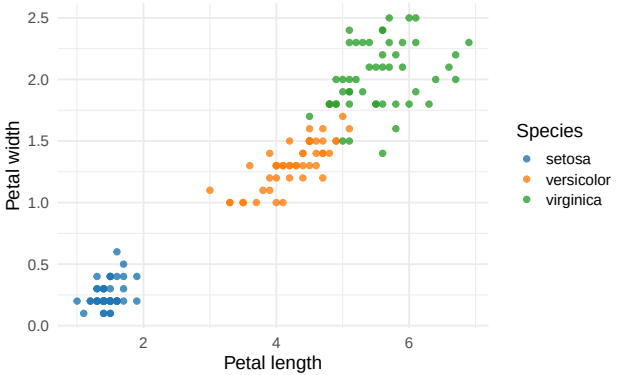
$$\mathbf{x}_i = \text{income, balance, utilization, payment history, } \ldots$$

and a label (whether the customer will default on credit card payment)

$$y_i \in \{\text{no default, default}\}.$$

- We want to predict the label for a new customer.
- Often we also want a probability of default, not just yes/no.

Another multiclass example



- Here $Y \in \{\text{setosa}, \text{versicolor}, \text{virginica}\}$ are three species of the iris flower.
- Some classes are easily separated, others overlap.
- Categories take values in an unordered set.

Classification: target and output

For a new observation with features $\mathbf{x}$, the optimal classification rule is

$$C^*(\mathbf{x}) = \arg\max_k \Pr(Y = k \mid X = \mathbf{x}),$$

which minimizes misclassification probability under 0–1 loss.

- In practice, the conditional probabilities $\Pr(Y \mid X)$ are unknown.
- A classifier may estimate these probabilities, or directly output a class label.
- A common decision rule is to predict the class with the largest estimated probability.
- The probabilities provide more information than the label alone (e.g., uncertainty).

Many classification methods can be viewed as estimating $\Pr(Y \mid X)$ and then applying a decision rule.

Binary classification as a special case

We now focus on a special case of classification: binary outcomes, where

$$Y \in \{0, 1\}.$$

- Our goal is to model
$$\Pr(Y = 1 \mid \mathbf{x}).$$
- Once we estimate this probability, we can classify by comparing it to a threshold (e.g., $1/2$).

Logistic regression provides a simple and widely used model for $\Pr(Y = 1 \mid \mathbf{x})$.

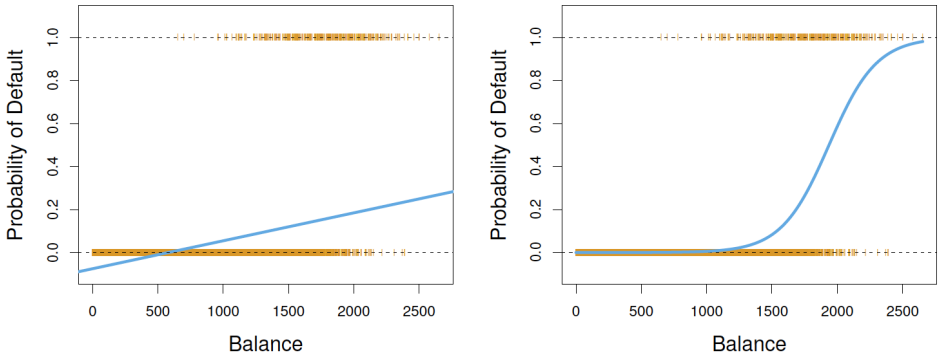
A binary classification example

- Data on Pima Indian women, used to predict diabetes status.
- $Y = 1$ (diabetes), $Y = 0$ (no diabetes); predictors include glucose, BMI, age, etc.
- The classes overlap in predictor space: no perfect separation.
- We therefore estimate $\Pr(Y = 1 \mid \mathbf{x})$ and base decisions on it.

Why not use ordinary linear regression?

- A linear regression fit to a binary outcome can produce fitted values below 0 or above 1.
- But a probability must stay in the interval $[0, 1]$.
- We want a model that is smooth, interpretable, and tied to class probabilities.

Linear regression (left) vs logistic regression (right)



Logistic regression

We model the conditional probability as

$$\Pr(Y = 1 \mid \mathbf{x}) = \frac{\exp(\beta_0 + \mathbf{x}^\top \boldsymbol{\beta})}{1 + \exp(\beta_0 + \mathbf{x}^\top \boldsymbol{\beta})}.$$

Equivalently, the log-odds are linear:

$$\log \frac{\Pr(Y = 1 \mid \mathbf{x})}{\Pr(Y = 0 \mid \mathbf{x})} = \beta_0 + \mathbf{x}^\top \boldsymbol{\beta}.$$

- The model ensures probabilities lie in $(0, 1)$.
- Linear structure is imposed on the log-odds scale.

Interpreting logistic regression

- Probabilities lie in $(0, 1)$, but log-odds range over all real numbers.
- A one-unit difference in x_j is associated with a change of β_j in the log-odds.

$$\exp(\beta_j)$$

is the multiplicative change in the odds associated with a one-unit difference in x_j.

- Example: if $\beta_j = 0.7$, then $\exp(\beta_j) \approx 2$, so the odds roughly double.

From probability to classification

To classify, we compare the estimated probability to a threshold:

$$\text{predict class 1 if } \Pr(Y = 1 \mid \mathbf{x}) > 0.5.$$

This is equivalent to

$$\beta_0 + \mathbf{x}^\top \boldsymbol{\beta} > 0.$$

- The default decision boundary is linear.
- The threshold can be adjusted if different errors have different costs.
- The predicted probability conveys uncertainty beyond the class label.

Maximum likelihood for logistic regression

We estimate the parameters by maximizing the likelihood

$$\prod_{i=1}^{n} p_i^{y_i}(1-p_i)^{1-y_i}, \quad p_i = \Pr(Y_i = 1 \mid \mathbf{x}_i).$$

Equivalently, we maximize the log-likelihood

$$\ell(\beta_0, \boldsymbol{\beta}) = \sum_{i=1}^{n} \left[y_i \log p_i + (1-y_i) \log(1-p_i) \right].$$

Taking derivatives gives the score equation (define $\boldsymbol{\theta} = (\beta_0, \boldsymbol{\beta}), \tilde{\mathbf{x}}_i = (1, \mathbf{x}_i)$)

$$\frac{\partial \ell}{\partial \boldsymbol{\theta}} = \sum_{i=1}^{n} \tilde{\mathbf{x}}_i (y_i - p_i) = \tilde{X}^\top (y - p).$$

- Unlike least squares, p_i depends nonlinearly on $\boldsymbol{\beta}$.
- So there is no closed-form solution.
- We solve this using iterative numerical methods.

Logistic regression with more than two classes

We extend logistic regression to $K > 2$ classes by modeling

$$\Pr(Y = k \mid \mathbf{x}), \quad k = 1, \ldots, K.$$

A common form is the softmax model:

$$\Pr(Y = k \mid \mathbf{x}) = \frac{\exp(\beta_{k0} + \mathbf{x}^\top \boldsymbol{\beta}_k)}{\sum_{\ell=1}^{K} \exp(\beta_{\ell 0} + \mathbf{x}^\top \boldsymbol{\beta}_\ell)}.$$

- Each class has its own linear score.
- Probabilities are positive and sum to 1.
- The binary case ($K = 2$) reduces to standard logistic regression.

Pima example: fitting logistic regression

- Use `MASS::Pima.tr` ($n = 200$) as training data.
- Outcome: `type = Yes/No`, with `Yes` meaning diabetes.
- The fitted model uses 7 predictors: `npreg`, `glu`, `bp`, `skin`, `bmi`, `ped`, `age`.
- Fit

$$\Pr(Y = 1 \mid \mathbf{x}) = \frac{\exp(\beta_0 + \mathbf{x}^\top \beta)}{1 + \exp(\beta_0 + \mathbf{x}^\top \beta)}$$

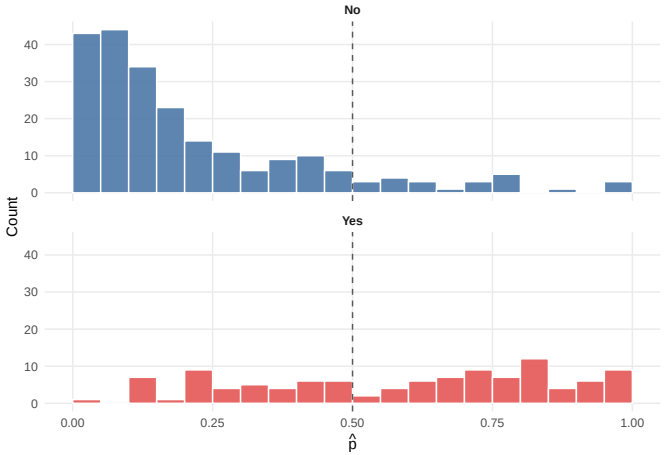
  using `glm(..., family = binomial)`.
- This is an associational predictive model, not a causal one.

Selected lines from R output

Term	Estimate	p-value
`npreg`	0.103	0.111
`glu`	0.032	< 0.001
`bmi`	0.084	0.051
⋮	⋮	⋮
`ped`	1.820	0.006
`age`	0.041	0.062

Held-out predicted probabilities on Pima.te

Same 7-variable fit, now applied to `Pima.te` ($n = 332$). The dashed line marks the 0.5 threshold.



From Pima predictions to a confusion matrix

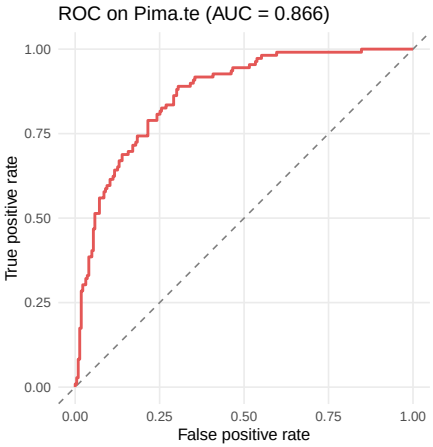
Using the 0.5 threshold on the held-out Pima test set:

	Predicted No	Predicted Yes
True No	200	23
True Yes	43	66

- This table is the confusion matrix.
- True positives (TP) $= 66$, true negatives (TN) $= 200$.
- False positives (FP) $= 23$, false negatives (FN) $= 43$.
- These quantities form the basis for common evaluation metrics.

Evaluation metrics (Pima test set)

- **Accuracy:** $(\mathrm{TP} + \mathrm{TN})/n = 0.801$
  overall proportion correct
- **Sensitivity (recall):** $\mathrm{TP}/(\mathrm{TP} + \mathrm{FN}) = 0.606$
  detects class 1 when present
- **Specificity:** $\mathrm{TN}/(\mathrm{TN} + \mathrm{FP}) = 0.897$
  correctly rejects class 1
- Different metrics emphasize different errors.
- **ROC:** sensitivity vs false positive rate as the threshold varies
  top-left is best (diagonal corresponds to random guessing); summarizes performance across thresholds



Test error is the relevant predictive target.

LDA: a generative alternative to logistic regression

- Logistic regression modeled $\Pr(Y \mid X)$ directly.
- LDA takes a generative route: model the feature distribution within each class,

$$f_k(\mathbf{x}) = f(\mathbf{x} \mid Y = k), \qquad \pi_k = \Pr(Y = k).$$

- Then Bayes' rule gives

$$\Pr(Y = k \mid X = \mathbf{x}) \propto \pi_k f_k(\mathbf{x}).$$

- This should feel familiar from Lecture 6: same Gaussian-mixture ingredients, but now the class labels are observed, not latent.

A 1D Gaussian example

- Suppose

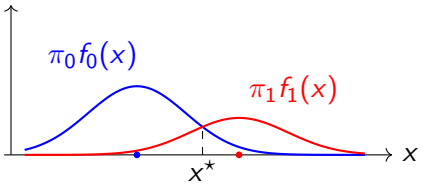
$$X \mid Y = 0 \sim N(\mu_0, \sigma^2),$$
$$X \mid Y = 1 \sim N(\mu_1, \sigma^2),$$

  with $\mu_1 > \mu_0$.

- Classify by comparing

$$\pi_0 f_0(x) \quad \text{and} \quad \pi_1 f_1(x).$$

- The boundary is where these two quantities are equal.
- With common variance, that boundary is a single threshold in 1D.



The LDA model

Assume that for class k,

$$\mathbf{X} \mid (Y = k) \sim \mathcal{N}(\boldsymbol{\mu}_k, \Sigma), \qquad \pi_k = \Pr(Y = k),$$

with a common covariance Σ across classes.

Then the posterior is obtained from

$$\Pr(Y = k \mid \mathbf{X} = \mathbf{x}) \propto \pi_k f_k(\mathbf{x}),$$

so we predict the class with the largest score (derivation in next slide)

$$\delta_k(\mathbf{x}) = \mathbf{x}^\top \Sigma^{-1} \boldsymbol{\mu}_k - \frac{1}{2} \boldsymbol{\mu}_k^\top \Sigma^{-1} \boldsymbol{\mu}_k + \log \pi_k.$$

- The Gaussian assumption is a modeling approximation.
- The shared-Σ assumption is the key step: the decision boundary is linear.
- So LDA is a probabilistic model that still produces hyperplane boundaries.

Why is the LDA boundary linear?

For class k,

$$\log\{\pi_k f_k(\mathbf{x})\} = \log \pi_k - \frac{1}{2}(\mathbf{x} - \mu_k)^\top \Sigma^{-1}(\mathbf{x} - \mu_k) + c.$$

Expanding the quadratic term gives

$$-\frac{1}{2}\mathbf{x}^\top \Sigma^{-1}\mathbf{x} + \mathbf{x}^\top \Sigma^{-1}\mu_k - \frac{1}{2}\mu_k^\top \Sigma^{-1}\mu_k + \log \pi_k + c.$$

- The term $-\frac{1}{2}\mathbf{x}^\top \Sigma^{-1}\mathbf{x}$ does not depend on k.
- It cancels when we compare classes.
- What remains is $\delta_k(\mathbf{x})$, which is linear in $\mathbf{x}$.

How do we fit LDA?

Under the Gaussian model, we maximize the log-likelihood

$$\ell(\{\mu_k\}, \Sigma, \{\pi_k\}) = \sum_{k=1}^{K} \sum_{i:y_i=k} \left[\log \pi_k - \tfrac{1}{2}(x_i - \mu_k)^\top \Sigma^{-1}(x_i - \mu_k) \right] + c.$$

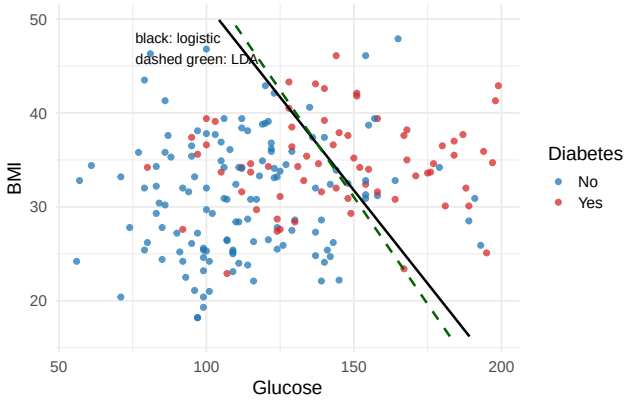
- This is a Gaussian mixture model with known (hard) labels.
- No E-step is needed; fitting reduces to a single M-step.

- Class proportions: $\hat{\pi}_k = n_k/n$
- Class means: $\hat{\mu}_k = \frac{1}{n_k} \sum_{i:y_i=k} x_i$
- Common covariance:

$$\hat{\Sigma} = \frac{1}{n-K} \sum_{k=1}^{K} \sum_{i:y_i=k} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^\top$$

These are exactly the MLEs under the Gaussian assumption.

Logistic v.s. LDA on the Pima dataset



- For visualization, this slide uses only two predictors (glucose and BMI).
- The classes overlap substantially, so no linear rule separates them perfectly.
- Logistic regression and LDA give similar but not identical boundaries.

LDA vs. logistic regression: conceptual comparison

LDA

- models $f(\mathbf{x} \mid Y)$ plus priors;
- uses Gaussian and shared covariance assumptions;
- can be more efficient when the model is well specified;
- requires estimating and inverting a covariance matrix.

Logistic regression

- models $\Pr(Y \mid \mathbf{x})$ directly;
- fewer assumptions on the distribution of X;
- often more robust to model misspecification;
- typically more stable when p is not small relative to n.

Both can give linear boundaries, but they differ in assumptions and in how difficult they are to estimate in finite samples.

Variations of LDA

- **Quadratic Discriminant Analysis (QDA):** assume

$$X \mid Y = k \sim \mathcal{N}(\mu_k, \Sigma_k)$$

  - allow each class to have its own covariance -¿ more flexible;
  - decision boundary becomes quadratic.
- **Regularized / shrinkage LDA:**
  - stabilize covariance estimation when p is large or n is small;
  - avoid overfitting from noisy covariance estimates.
- **Naive Bayes (diagonal Σ):** assume

$$X \mid Y = k \sim \mathcal{N}(\mu_k, \Sigma_k), \quad \Sigma_k \text{ is diagonal}$$

  - features are conditionally independent given the class;
  - simple and stable, especially in high dimensions.

All are Gaussian-based classifiers with different covariance assumptions.

A practical workflow for classification

1. Define the prediction target and evaluation metric.
2. Split data (train/test or cross-validation).
3. Fit candidate models (e.g., logistic regression, LDA).
4. Compare out-of-sample performance.

Class imbalance and threshold choice

- When one class is rare, accuracy can be misleading.
- A classifier may predict the majority class most of the time.
- Adjust the decision threshold (not always 0.5) to trade off sensitivity and specificity.
- Use sensitivity, specificity, or ROC instead of accuracy alone.

An R demo of classification

Notebook file: `Lecture9_demo.html`

Lecture 9 summary

- Classification predicts a discrete label from observed features.
- The Bayes classifier is the population target under 0-1 loss.
- Logistic regression models conditional class probabilities directly.
- LDA is a model-based classifier connected to Gaussian mixture ideas.
- Both often yield linear decision boundaries, but from different assumptions.
- In high dimensions, both need more careful estimation or regularization.

Suggested reading

- James, Witten, Hastie, Tibshirani (2nd edition), Chapter 4.
- Johnson & Wichern (6th Edition), Chapter 11.1-11.5, 11.7.