Introduction to Classification (Logistic Regression, Interpretation, ROC)

Daniel Lawson University of Bristol

Lecture 05.1.1 (v1.0.2)

Signposting

- We have wrapped up **classic statistics** with a discussion on non-parametrics, kernels, and a practical on missing data and outliers.
- The remainder of the course changes the focus towards machine-learning - especially the background of the key tools that are used in practice.
- It is important to emphasise that classification is statistics, though we use the parlance of machine learning.
  - Most of machine learning is also modern statistics.
  - The main distinction is about use: whether we use the results only for prediction, or for understanding.
  - Which ultimately is no distinction at all…

Signposting (2)

- This is part 1 of Lecture 5.1, which is split into:
  - 5.1.1 covers a Classification Introduction and Interpretation
  - 5.1.2 covers kNN, LDA, SVM
- In 5.2 we cover boosting and ensemble methods
- In 6 we cover Tree and Forest methods

Intended Learning Outcomes

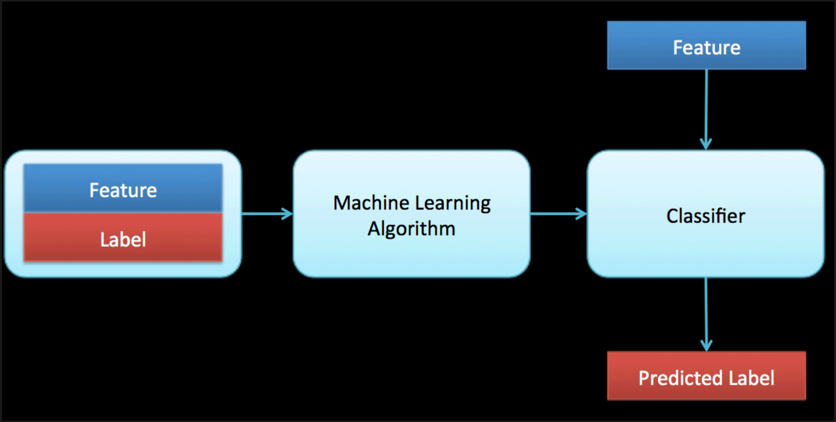
- ILO2 Be able to **use and apply basic machine learning** tools
- ILO3 Be able to make and report appropriate inferences from the results of applying basic tools to data

Types of machine learning

- Unsupervised: **no labels**. For example,
  - Clustering
  - Dimensionality reduction
  - Smoothing
- Supervised: exploits **labels**. For example,
  - Classification
  - Regression
- Other types:
  - Semi-supervised: **some labels** are available
  - Active: can **choose which labels** to obtain
  - Reinforcement: **reward based**. explore vs exploit?
  - etc.

Classification

- Machine Learning classification is about how to make good predictions of **classes** based on previous experience of how **features** relate to **classes**.



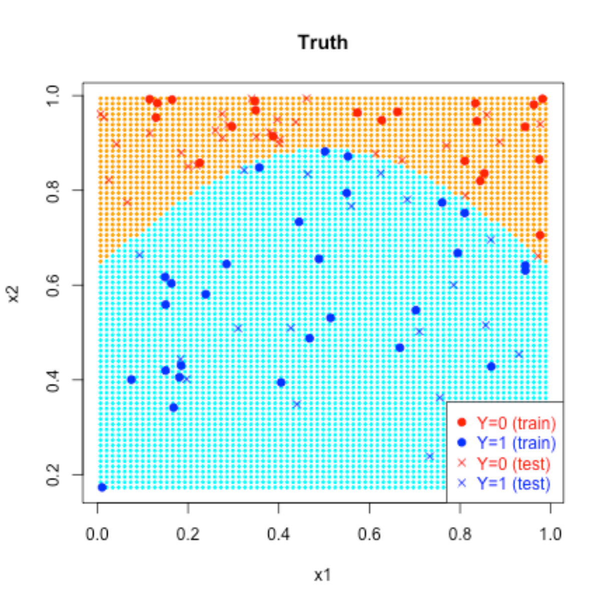
Examples of classification

- **Spam** filtering (spam/not spam)
- **Face detection** (image classification)
- **Speech recognition**
- **Handwriting recognition**
- **Turing test** … though that is human, not machine!
- In cyber:
  - **Malware detection** … when comparing to historical malware
  - **Intruder detection** … when comparing to intrusion models
- Classification is broadly the "detection, recognition, recall of **prior experience**".

Some Important Classifiers

- **Logistic Regression** (Block 2 and 5)
- **K-Nearest Neighbours** (Block 4 and 5)
- **Linear Discriminant Analysis** (Block 5)
- **Support Vector Machines** (Block 5)
- **Decision Trees** (Block 6)
- **CART**: Classification and Regression Trees (Block 6)
- **Random Forests** (Block 6)
- **Naive Bayes** (Block 7)
- **Neural Networks** (Block 9)

Classification



From Regression to Classification

- In Week 3 we discussed **linear regression**, i.e. obtaining solutions to:

$$y_i = \vec{x}_i \cdot \beta + e_i$$

- in scalar form, where we have p' covariates and have $\vec{x}_i = (1, x_{1,i}, \cdots, x_{p',i})$, so $\vec{x}_i$ and β are both vectors of length $p = p' + 1$, and e_i are the residuals whose squared-sum is minimised.
- Logistic regression instead solves for the probability that a binary outcome y is 1:

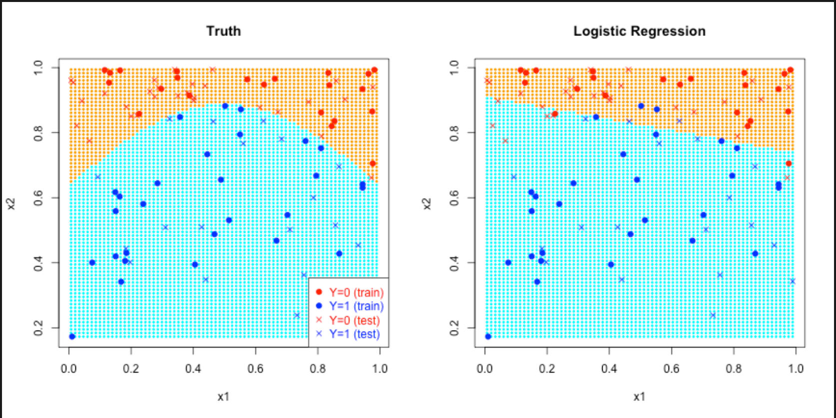
$$\text{logit}(p(y_i)) = \ln\left(\frac{p(y_i)}{1 - p(y_i)}\right) = \vec{x}_i \cdot \beta + e_i$$

- The model then assumes $y_i \sim \text{Bern}(p(y_i))$. The prediction is the **log-odds** ratio, with values > 0 predicting a 1 and values < 0 predicting a 0.

Logistic Regression fitting

- Logistic regression is an example of a **generalised linear model** or **GLM**.
- In general these cannot be directly solved with Linear Algebra. Options include:
- **Maximum likelihood** estimation:
    - A numerical procedure can be used to maximise the likelihood in terms of the parameters β, and σ the variance of e.
- Iteratively Reweighted **least squares** (IRLS):
    - The non-linearity can be adopted into weights, and a linear algebra solution reached.
    - Then the weights are updated, and the procedure iterated.
- Co-estimation tends to be relatively computationally costly (higher dimensional space) but to have better estimation properties.
- In both cases we look for sub-problems that can be efficiently solved.

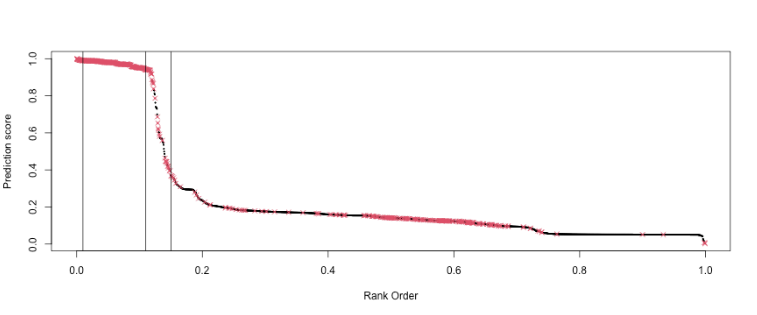
Logistic Regression example



Classification Performance

- We can always compute **training** and **test** dataset accuracy.
- However, we should only ever compare performance on **test** data, to prevent over-fitting.
- Classifiers are understood through their **Confusion Matrix**, that is a comparison between:
  - Ground truth class, and
  - Predicted classes.
- For binary classes, we summarise using (true/false)(positive/negative) outcomes.
- Binary classification is particularly convenient as most classifiers can provide **scores** rather than **class predictions**.
  - Scores are **ordered**. So we can choose a threshold to control the total proportion of **positive predictions**.
  - This provides a **relationship** between **Positive Claims** and **True Positives**.

Classification Performance



.	$Y = 1$	$Y = 0$	Condition
$\hat{Y} = 1$	TP	FP	Prediction positive
$\hat{Y} = 0$	FN	TN	Prediction negative
Claim	Truth positive	Truth Negative	.

Classification Performance Representations

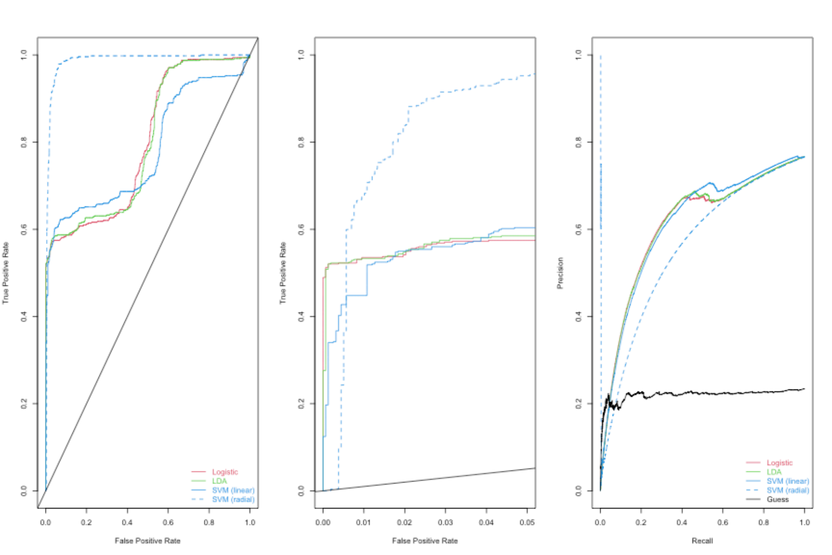
- There are many ways to represent performance
- The **Receiver-Operator-Curve (ROC)** is the most popular, as it holds regardless of the true distribution of the data.
  - X-axis: False Positive Rate (FPR) $= P(\hat{Y} = 1|Y = 0)$
  - Y-axis: True Positive Rate (TPR) $= P(\hat{Y} = 1|Y = 1)$
  - The **Area Under the Curve (AUC)** is a measure of Accuracy (0.5=guessing, 1=perfect).
  - We need to care about the region of the ROC curve that matters.
- The **Precision-Recall curve** is appropriate when we care specifically about positive cases:
  - X-axis: Precision $= P(Y = 1|\hat{Y} = 1)$
  - Y-axis: Recall=TPR $= P(\hat{Y} = 1|Y = 1)$

Some important properties

- Some nice things[1] can be said about ROC and PR curves:
- Dominance:
  - If one curve dominates (is always above) another in ROC, it dominates in PR
  - and vice-versa
- ROC curves can be linearly interpolated
  - This is "flipping a coin" to access classifiers in-between
- PR curves have a slightly more complex relationship but the same principle can be applied
- Integrating both scores leads to performance metric that can be optimized

[1] Davis and Goadrich, "The Relationship Between Precision-Recall and ROC Curves", ICML 2006.

ROC/PR Curve Example



Metrics for Classification

- Accuracy (Proportion of samples classified correctly) is a terrible metric if classes are unequal
- TPR at a given FPR is more flexible
- AUC characterises the whole ROC curve
- Area Under Precision-Recall Curve (AUPRC?) is also a thing people advocate for
- None are "right", we have to define the inference task
- Any of these and more are often optimized
    - If we optimise a parameter or perform model comparison based on test data, we need additional test data to test the meta-algorithm!

Signposting:

- Next up: Some example Classification methods: Linear Discriminant Analysis, Support Vector Machines.
- We Reflect after 5.1.2.
- **References**:
  - Stack Exchange Discussion of ROC vs PR curves.
  - Davis and Goadrich, "The Relationship Between Precision-Recall and ROC Curves", ICML 2006.
  - Rob Schapire's ML Classification features a Batman Example…
  - Chapter 4 of The Elements of Statistical Learning: Data Mining, Inference, and Prediction (Friedman, Hastie and Tibshirani).