Networks Structure and Dynamics
6. Link prediction using data mining

Lionel Tabourier, Fabien Tarissan

LIP6 – CNRS and Université Pierre et Marie Curie

first_name.last_name@lip6.fr

October 25th 2016

Outline

1. Introduction – Context
2. A statistical classification problem
3. Application to link prediction

The link prediction problem

Problem description

V is a fixed set of node,

- interactions known between t_0 and t'_0
- which links appear(/disappear) between t_1 and t'_1

Relevance?

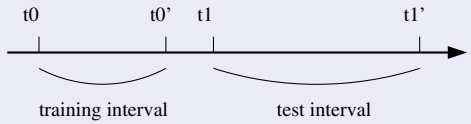
f: fraction of new links with new nodes
$\Rightarrow$ we should have: $f \ll 1$

FaceBook new friendships in 1 month?
New IP addresses with traceroute measurements in 1 month?

The link prediction problem

Principle

Liben-Nowell, Kleinberg - *JASIST, 2007*



Predict links of $G[t_1, t'_1]$ which are not in $G[t_0, t'_0]$:

Use properties correlated with the probability for a link to appear

The missing link problem

Principle

- suppose that the data crawling process missed links
- ⇒ detect unseen links using same methods

Imbalance

In large networks,
unconnected pairs much more frequent than connected pairs
impact for link prediction and missing link detection...

The missing link problem

Principle

- suppose that the data crawling process missed links
- ⇒ detect unseen links using same methods

Imbalance

In large networks,
unconnected pairs much more frequent than connected pairs
impact for link prediction and missing link detection...

Outline

1. Introduction – Context
2. A statistical classification problem
3. Application to link prediction

What is a statistical classification problem?

What is statistical classification?

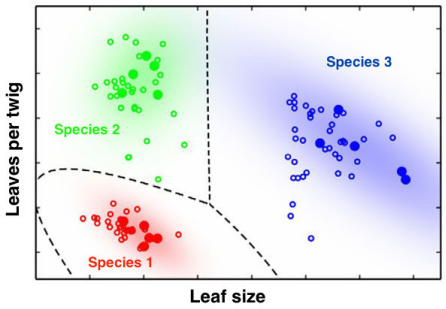
- classification:
  fixed number of groups, a group for each item
- statistical:
  based on comparison of the item features to a population of items already classified

Examples

- Family of animals according to size, appearance
- Medical diagnosis using symptoms
- Characters recognition

What is a statistical classification problem?

Example: species of plants



And here?

The classification problem

- **items:** pairs of nodes
  2 classes: exists (link) or not (unconnected pair)
- **information sources:**
  graph structure, features of nodes, features of existing links

Unsupervised or supervised learning?

Unsupervised learning:
no data already classified (ex: clustering problems)

Supervised learning:
we have examples where the result is already known
more controlled and more efficient

In general, we can use the existing network for prediction
⇒ supervised formulation possible

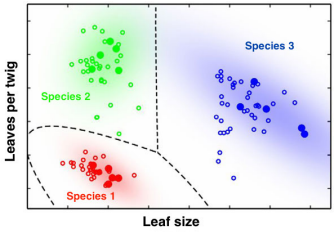
Unsupervised or supervised learning?

Unsupervised learning:
no data already classified (ex: clustering problems)

Supervised learning:
we have examples where the result is already known
more controlled and more efficient

In general, we can use the existing network for prediction
⇒ supervised formulation possible

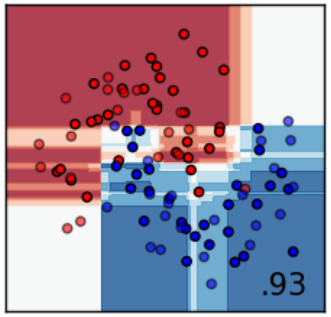
Questions to solve

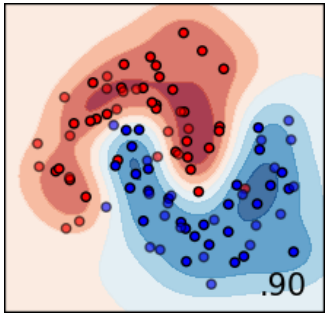


- how to draw frontiers?
- how many parameters?
- how to evaluate results?

How to draw frontiers?

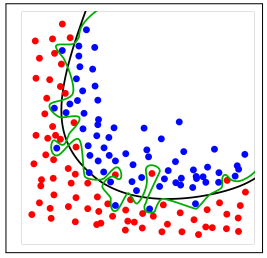
Model selection





How many parameters?

The problem of over-fitting *(fr: surapprentissage)*



Frontier particular to the training set

How to evaluate results?

Elementary measures (for 2 classes classification):

	prediction: +	prediction -
reality: +	true positive	false negative
reality: -	false positive	true negative

The cost of a wrong classification

Spam detection : false positive $\gg$ false negative
Cancer detection : false positive $\ll$ false negative

How to evaluate results?

Elementary measures (for 2 classes classification):

	prediction: +	prediction -
reality: +	true positive	false negative
reality: -	false positive	true negative

The cost of a wrong classification

Spam detection : false positive $\gg$ false negative
Cancer detection : false positive $\ll$ false negative

Example of model: classification trees

Dataset: set of points (item, class): $E_{tot} = (x_1, x_2, ..., x_n, y)$

Principle

Build a tree from data such that:

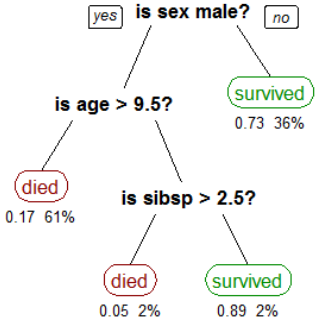
- the **root** of the tree is the whole set E_{tot}
- each **node** i is a subset E_i
- each **branch** from i is a partition of E_i according to a condition of the form: $x_i \leq \alpha$ or $x_i > \alpha$
- **branching: each son as homogeneous as possible**

Example of model: classification trees

How to ...

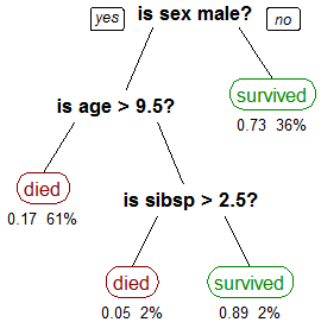
- ... split into homogeneous subsets?
  partition criterion
  ex: ***minimize Gini coefficient****:* $\Sigma_k f_k(1 - f_k)$
  with f_k *fraction of elements in class k*
- ... to stop the process?
  tree leaves **homogeneity** criterion
  ex: if a leaf has 90% of its elements in one class

Example: Titanic survival classification tree



Many other models: k Nearest Neighbors, Support Vector Machines, Neural Networks, ...

Example: Titanic survival classification tree



Many other models: k Nearest Neighbors, Support Vector Machines, Neural Networks, . . .

Outline

1 Introduction – Context

2 A statistical classification problem

3 Application to link prediction

Typical problem

Liben-Nowell, Kleinberg - *JASIST, 2007*

Datasets

Scientific collaboration networks:

- node = authors, link = co-publication
- publications in *DBLP, arXiv, Medline...*
- number of articles: a few thousands per year
- number of authors: a few thousands

Protocol

Year A to predict **new** collaborations in year $A+1$

Prediction features (part 1)

Local structural characteristics

- Number of common neighbors:

$$|\mathcal{N}(i) \cap \mathcal{N}(j)|$$

- Jaccard index:

$$\frac{|\mathcal{N}(i) \cap \mathcal{N}(j)|}{|\mathcal{N}(i) \cup \mathcal{N}(j)|}$$

- Adamic-Adar index:

$$\sum_{k \in \mathcal{N}(i) \cap \mathcal{N}(j)} \frac{1}{log(\delta(k))}$$

- etc...

Prediction features (part 2)

Global structure features

- Katz index:

$$\sum_{L=1}^{\infty} \beta^L \nu_{ij}(L)$$

  ν_{ij}: number of paths of length L from i to j
  β : parameter < 1
- preferential attachment index: $|\mathcal{N}(i)|.|\mathcal{N}(j)|$
- *hitting time* in j starting from i
- etc...

Non-structural characteristics

- similarity index between nodes i and j *(age, gender, for scientists: field of expertise...)*

Assess the quality of the prediction

Elementary measures (for 2 classes classification):

	prediction: +	prediction -
reality: +	true positive	false negative
reality: -	false positive	true negative

Usual measures:

- **precision**, $\mathbf{Pr} = \frac{\#tp}{\#tp+\#fp}$
- **recall** (*fr: rappel*), $\mathbf{Rc} = \frac{\#tp}{\#tp+\#fn}$
- **F-score**, $\mathbf{F} = \frac{2.\mathbf{Pr}.\mathbf{Rc}}{\mathbf{Pr}+\mathbf{Rc}}$ (trade-off between Pr and Rc)
- and others (*fall-out*$= \frac{\#fp}{\#fp+\#tn}$, *ROC* curve,...)

Quality assessment for link prediction

Prediction in large networks, **class imbalance problem**:
high risk of FP
$\Rightarrow$ precision often low

A basic protocol

- set N_{new}, the number of new links that appear
- keep the N_{new} top scoring items according to each feature
- compare to a random prediction

Quality assessment for link prediction

Prediction in large networks, **class imbalance problem**:
high risk of FP
$\Rightarrow$ precision often low

A basic protocol

- set N_{new}, the number of new links that appear
- keep the N_{new} top scoring items according to each feature
- compare to a random prediction

Results

Predictor		astro-ph	cond-mat	gr-qc	hep-ph	hep-th
probability that a random prediction is correct		0.475%	0.147%	0.341%	0.207%	0.153%
graph distance (all distance-2 pairs)		*9.4*	*25.1*	*21.3*	*12.0*	*29.0*
common neighbors		**18.0**	**40.8**	**27.1**	**26.9**	**46.9**
preferential attachment		4.7	6.0	7.5	*15.2*	7.4
Adamic/Adar		*16.8*	**54.4**	**30.1**	**33.2**	**50.2**
Jaccard		*16.4*	**42.0**	19.8	**27.6**	*41.5*
SimRank	$\gamma = 0.8$	*14.5*	*39.0*	*22.7*	*26.0*	*41.5*
hitting time		6.4	23.7	*24.9*	3.8	13.3
hitting time—normed by stationary distribution		5.3	23.7	11.0	11.3	21.2

Results