Running a DM Project

Jingpeng Li

A Typical DM Project

- The client asks if you can use their data to build a system for predicting or classifying things in the future
- They say they have ‘plenty’ of data and they send you a file
- The data is incomplete, unsuitable to the task and would lead to a poor result
- The end

Sometimes, However

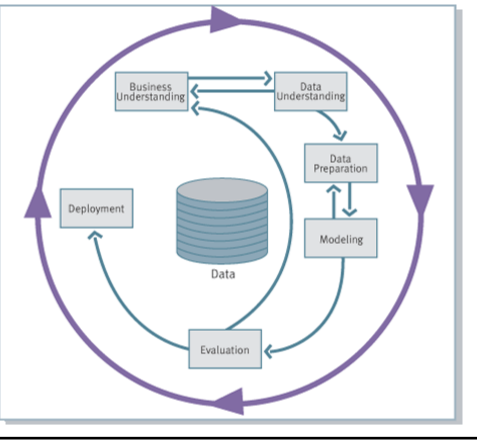
The data is suitable in quality and quantity and the project proceeds as follows:

- You obtain a base line for performance
- You spend a lot of time preparing the data
- You use the data to train several different models to see which is most suitable
- You choose the technique that led to the best model and build several to verify the robustness of the model

© University of Stirling 2019

CRISP DM Standard

- CRoss Industry Standard Process for Data Mining



Data Preparation

- We have a whole lecture on this topic, but in summary:
- Clean the data
  - Remove rows with missing values
  - Remove rows with obvious data entry errors – e.g. Age = 200
  - Recode obvious data entry inconsistencies – e.g. If Gender = M or F, but occasionally Male
  - Remove rows with minority values

Data Quantity

- Choose the variables to be used for the model
- Look at the distributions of the chosen values
- Look at the level of noise in the data
- Decide whether or not there are sufficient examples in the data
- Treat unbalanced data

Consider Error Costs

- Imagine a system that classifies input patterns into one of several possible categories
- Sometimes it will get things wrong, how often depends on the problem:
  - Direct mail targeting – very often
  - Credit risk assessment – quite often
  - Medical reasoning – very infrequently

Error Costs

- An error in one direction can cost more than an error in the opposite direction
  - Blood test
    - Recommending a blood test based on a false positive is better than missing an infection due to a false negative
  - Insurance fraud
    - Missing a case of insurance fraud is more costly than flagging a claim to be double checked

Model Building

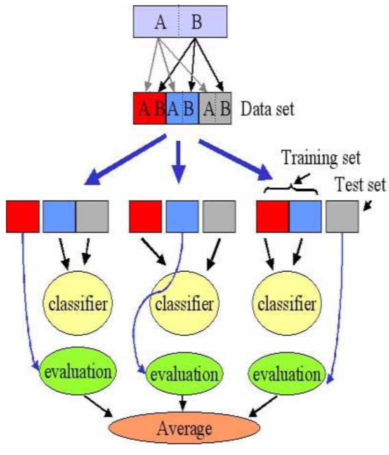
- Choose a number of techniques suitable to the task:
  - Neural network (MLP or RBF for prediction or classification)
  - Decision tree for classification
  - Rule induction for classification
  - Regression for prediction
  - Bayesian network for classification

Train Models

- For each technique:
  - Run a series of experiments with different parameters, e.g. number of hidden units in a MLP
  - Each experiment should use around 70% of the data for training and the rest for testing
  - When a good solution is found, use cross validation (10 fold is a good choice) to verify the result

Cross Validation

- Split the data into ten subsets, then train 10 models – each one using 9 of the 10 subsets as training data and the 10th as test. The score is the average of all 10.
- This is a more accurate representation of how well the data may be modelled, as it reduces the risk of getting a lucky test set



© University of Stirling 2019

Assess Models

- You can measure the success of your model in a number of ways
  - Mean Squared error – not always meaningful
  - Percentage correct for classification
  - Confusion matrix for classification

n=220

Output=	True	False
True	80	30
False	20	90

© University of Stirling 2019

Probability Outputs

- Most classification techniques provide a score with the classification – either a probability or some other measure
- This can be used:
  - – Allow an answer of “unsure” for cases where no single class has a high enough probability
  - – Weighting outputs to allow for unequal cost of outcomes
  - – Cumulative Gains charts and ROC curves

Cumulative Gains Chart

- Let’s say we want to identify prospects for a mailing campaign
- A model could score every prospect in a large set
- Sorting that set by score would place the best prospects at the top
- Imagine we mail them all and see who responds
- We should find that the top 1000 produces more responses that the last 1000 do

Cumulative Gains Chart

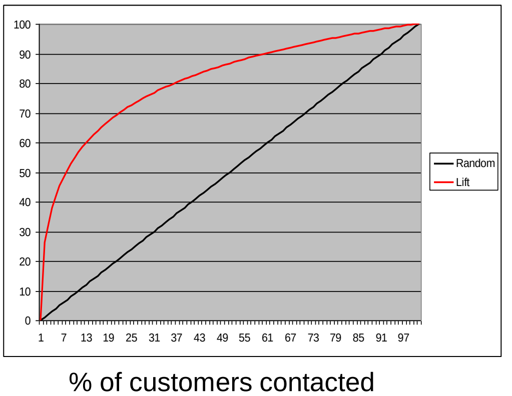
- Splitting the data into same sized bands and counting the number of respondents (correct predictions) in each produces a diminishing set of counts
- Plotting the cumulative total of these counts produces what is known as a lift curve

Lift Curve

- The lift curve below shows the return you would get if you mailed at random in black and the lift curve after modelling in red

http://www2.cs.uregina.ca/~dbd/cs831/notes/lift_chart/lift_chart.html

% of positive responses



% of customers contacted

ROC Curves

- With a two-class problem, you might set the threshold probability at 0.5
- If the probability of rain is 0.8 and the probability of it being dry is 0.2, you would, if forced to make a prediction, say it will rain
- If you want to be optimistic, you could set the threshold at 0.7 for rain, so you need to be more certain before you will predict rain

ROC Curves

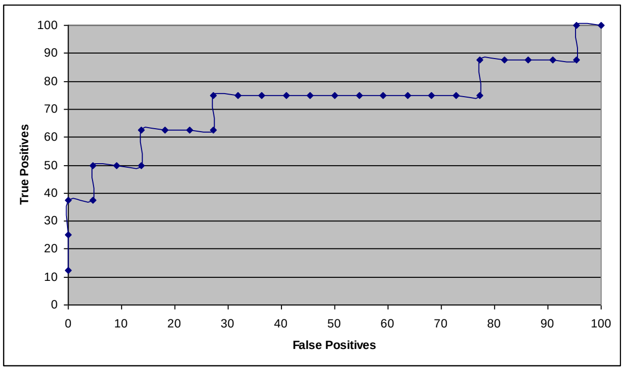
- An ROC curve (**R**eceiver **O**perating **C**haracteristic) is similar to a lift curve
- It tells you how many false positives and true positives you would get for each possible threshold
- The threshold for a positive is varied from 0 to 1 and the number of true positives and false positives counted for each

ROC Curves

- The idea is to optimise the trade-off between finding as many of the positives as possible, while wrongly including as few negatives as possible
- Every additional decision with a lower confidence (probability) is a move towards randomness
- You could get all the positives by saying “yes” to everything, but you would include all the negatives too, so the top RHS of the ROC curve is 100%, 100%

An ROC Curve

- The ROC curve below shows that the more of the positives we identify, the more false positives creep in



Example

- A client of mine sells predictions to the record industry on how well new singles will do
- I built them a 'Hit Classifier' which classes a song as a Hit or a Miss
- The cost of not releasing a hit is **low** – the record company forgets those
- The cost of releasing a miss is **high** – the song flops, everyone looks bad

Example

- The company wants to flag as many hits as it can (it can't sell a long list of misses!) without any false positives (saying Hit when it turns out to be a miss)
- We want to find the highest Hit threshold that excludes false positives
- The ROC curve tells us where this is

Implementing the System

- Once the predictor is built and tested,
- The thresholds for classifications are set
- And the client is happy with the level of performance,
- You need to make the predictions available as part of their business process

Embedding DM

- This could be a predictor behind a web site or a call centre, or any other 'customer facing part of a business', choosing offers specific to each customer
- It could be a desktop software package that a business analyst uses for planning
- It could be on a chip, built into a consumer product – washing machines, cars, microwave ovens all have them

Summary

- Collect, check and process data
- Choose several techniques. For each:
  - Choose several different parameter sets
  - Use 10 fold validation to build models
  - Choose 1 model to use
  - Plot errors and ROC curves
- Pick the best model or use a combination of several
- Embed in a live system

Commercial Data Mining

Opportunities and Challenges

Contents

Traditional Software Projects

- Selling, specifying, developing and delivering a traditional software system:
  - Specification describes exactly what it will do
  - Result can be measured against spec
  - Payment can be demanded when it is shown to be complete

Data Driven Projects

- Specification can only say what the system will try to do
- Data quality might prevent it from working
- Who takes the risk? Will the client still have to pay if it doesn't work?
- What level of accuracy is expected?
- Speculatively building solutions is risky

Barriers to Uptake Technical

- Lack of data
  - Insufficient quantity
  - Data unavailable when required
- Data does not contain the information required to support the task
- Specifying a project that relies on data for both its definition and its operation
- Handling errors – many techniques are probabilistic

Barriers to Uptake Cultural

- Explaining and selling the concept
- Proving the concept before seeing the data
- Replacing intelligent workers with ‘intelligent’ computers
- Managing expectations

Replacing Experts

- Industrialisation has made a lot of manual labour redundant
- Doing the same for human experts in cerebral jobs is a greater challenge
- They usually have more power in a company
- They would need to help you build the system that might replace them

Example – Insurance Risk

- Underwriters are skilled at assessing risk and managing the exposure of an insurance company
- An intelligent computer system would have a good chance of out performing them
- Could you persuade an insurance company to trust the most crucial aspect of their business to a computer system they can't understand?
- It would be easier to start your own insurance co.

Selling a Solution

- DM solutions can be hard to sell because:
  - You can't be sure it will work until you have seen the data
  - You can't demonstrate it working at a sales pitch (not on their data, anyway)
  - You may need to sell the power of the technology in order to make the sale, but that can set expectations too high

Barriers to Uptake Conceptual

- Computers can learn
- Computers can learn better than humans
- Computers can make mistakes based on what they have learned
- Concepts of data driven systems …

Concepts of data driven systems

- Non-linearity, often in multiple dimensions
- Confidence levels and errors
- Generalisation and over fitting
- Data quality and quantity
- Results dependent on data!!