Blue Book for Bulldozers

Hajic, Havranek, Taufer, Tomasek
Problem - description

- https://www.kaggle.com/c/bluebook-for-bulldozers
- The goal of the contest is to predict the sale price of heavy equipment at auction
Source data

- all data are stored in simple csv
- but there is huge amount of noise in these data
  - some bulldozers are made in year 1000
  - different YearMades attached to the same MachineID
  - strange MachineHoursCurrentMeter values
    - example:
      - SalesID 2318649
      - Value 2 483 300
      - Year made 2005
      - \((2013-2005) \times 24 \times 365 = 70 080 :\)
Evaluation

- Root Mean Squared Logarithmic Error ("RMSLE")

\[ \epsilon = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (\log(p_i + 1) - \log(a_i + 1))^2} \]

- \( p_i \) - your predicted value
- \( a_i \) - real value
- \( n \) - count
Source data - relevancy

- show excel description
- is fork type or transmission relevant for final price?
- how can we find out?
- can we find it out manually or using some magic machine learning?
Possible solutions

- **Question-form**
  - FHS style
  - ask people in Prague
- **Genetic programming**
- **Neural networks**
Statistics - observations

- 3/4 only once
- one piece sold 26 times
- data aren't complete

...
Statistics - solution

● Regression
  ○ According to observation linear is not enough
  ○ Polynomial is needed
    ■ grade about 3-4 will be enough
Statistic - what's completed

- Parsing script
- Analyzing scripts
- Observation picture diagrams generator
  - Currently running in lab
Solution?

- We don't know how to solve this problem
  - Let's cultivate the solution -> genetic programming
    - The buyer will be product of evolution
  - Inspiration / literature:
    - Tomáš Křen: Genetic functional programming presentation
    - Genetic programming research group http://www.genetic-programming.com
Genetic programming

- **Population**
  - Member = *Price calculation function*
    - Tree of functions :: [Price] -> Price
      - Arithmetical / logical / load / SQL history aggregation
  - Fit function = difference from actual price in DB
    - same as the official
  - Reproduction
    - Switch subtrees on random layer
    - ... picture diagram
  - Mutation
    - change function in specific node
Genetic solution - data

- Input data (training data)
  - Structured in SQL database
  - Special nullary function nodes access the data
  - Bulldozers table
    - Stores known bulldozers specification and price

- Input object
  - Bulldozer for auction
  - Structure = database table row without price specified
    - [Int] numeric values
    - [Enum] enum values
Genetic solution - node functions

- **Constant**
  - :: Price

- **Arithmetical**
  - Classical operations
  - :: [Price] -> Price
    - Price is numeric type - double/real

- **Logical**
  - if-then-else
    - <, <= ...
  - :: [Price] -> Price
Genetic solution - node functions

- **Load**
  - :: Price
  - Loads specific cell from input object
    - number value
      - mask as price and returns for next operation (usually arith.)
    - enum value
      - mask as price for only logical functions

- **SQL Aggregation**
  - :: Price
  - Selects from history database values
    - using aggregation function (count, max, sum...)
    - using where based on input object
Genetic solution - convergence

- Solution is very generic
  - Needs optimizations, heuristics, constraints...
Genetic solution - subproblems

● Not all data columns are dependent
● Split price calculation by column groups
  ○ k separated evolution runs with smaller members
    ■ using only few columns for loading and sql agg. functions
  ○ One small function for aggregation

● Columns
  ○ globals
  ○ specials
  ○ ... picture diagram
Genetic solution - confidence

- During the process is calculated confidence of returned price
  - effects final aggregation
  - effects selection in evolution process
- Example
  - confidence is low when database history doesn't contain data similar to input object
Genetic solution - constraints

- **Constants**
  - Take from final universum
    - example: equally taken subset of $[0,1]$

- **Type constraints**
  - Input object
    - arithmetical operations for number values
    - for enum values only logical
      - special switch

- **Generic**
  - Max deep
Genetic solution - heuristic

- Startup population member
  - Not only random generated
  - Based on human rational guess
    - From SQL agg. uses only avg, median...
  - Based on other team member's results
- Small column groups
Genetic solution - what's done

- Team foundation server
- Generic node abstraction
- Arithmetical nodes
- Data parsing in SQL
Jakub's presentation

http://www.youtube.com/watch?v=SJI5v9QoPus