Business Data Analytics

Lecture 13

The slides are available under creative common license. The original owner of these slides is the University of Tartu
Fraud Detection

Wrongful act for financial gain
Fraud (google returns)
Fraud is an uncommon, well-considered, imperceptibly concealed, time-evolving and often carefully organized crime which appears in many types and forms.
Fraud

Fraud is when trickery is used to gain a dishonest advantage, which is often financial, over another person.

There are many words used to describe fraud: Scam, con, swindle, extortion, sham, double-cross, hoax, cheat, ploy, ruse, hoodwink, confidence trick.

These are just a few words you might hear in relation to fraud. Fraud can be committed against individuals or businesses. Have a look in our A-Z of fraud for information about different types of fraud.

Cyber crime is any criminal act dealing with computers and networks (called hacking). Additionally, cyber crime also includes traditional crimes conducted through the Internet.

Facts about fraud and cyber crime

The Annual Fraud Indicator 2016 estimates the cost of fraud to the UK is £193bn a year.

Source: https://www.actionfraud.police.uk/what-is-fraud
Fraud: where it happens?

- Fraud detection is a topic applicable to many industries including:
  - banking and financial sectors,
  - Insurance,
  - government agencies
  - law enforcement,
  - Actually all ....

Source: http://www.statsoft.com/Textbook/Fraud-Detection
Impact of Fraud and Examples

• Some of the Frauds across the world:
  • http://www.fraud-magazine.com/article.aspx?id=4294974524
  • https://www.actionfraud.police.uk/fraud-az-government-agency-scam
Examples

Caterpillar slides following a report that accused it of tax and accounting fraud
Examples

Caterpillar slides following a report that accused it of tax and accounting fraud

VW executive gets seven years for U.S. emissions fraud

DETROIT (Reuters) - A U.S.-based Volkswagen AG (VOWG_p.DE) executive who oversaw emissions issues was sentenced to seven years in prison and fined $400,000 by a judge on Wednesday for his role in a diesel emissions scandal that has cost the German automaker as much as $30 billion.
Type of Frauds

• First-party bank fraud
• Money laundering
• Insurance claim fraud
• E-commerce fraud
• Tax Evasion (by declaring the company bankrupt)
• Credit card transactions
• Opinion fraud

• Procurement fraud
• Tax fraud
• Benefits fraud
• Improper Payments
• Social security fraud
• Call behavior fraud
• Scientific fraud
• Terrorism
Traditional approaches to solve Fraud

Discrete data analysis

Pros: Simple. Works with rookie fraudsters.
But (usually) not always possible

• Normal behavior
  • Tendency of fraudsters to blend into the surroundings (camouflage)
  • No longer fraud by opportunity, but carefully planned.
• How to detect Mr. Hyde?
Advanced approaches to solve Fraud

• Need for new techniques:
  • Evolving from *descriptive statistics* towards *predictive statistics* (data mining)
  • Learning from historical data to judge upon new observations,
  • Detection of *patterns* that initially seem to comply to normal behavior, but in reality instigate fraudulent activities
A simple model

Discrete Data Analysis

Layer 1: endpoint-centric
- Analysis of users and their endpoints

Layer 2: navigation-centric
- Analysis of navigation behavior and suspect patterns

Layer 3: account-centric
- Analysis of anomaly behavior on a per-channel

Gartner’s Layered Fraud Prevent

Gartner at http://www.gartner.com/newsroom/id/1695014
A simple model

**Gartner’s Layered Fraud Prevention Approach**

Gartner at [http://www.gartner.com/newsroom/id/1695014](http://www.gartner.com/newsroom/id/1695014)
Is fraud a social phenomenon?

Credit card transaction fraud:
- Stolen credit cards (yellow nodes) are often used in the same stores (blue nodes)
- Store itself also processes legitimate transactions to cover their fraudulent activities

Source: www.dataminingapps.com
Insurance Fraud – “Whiplash for Cash”

• What is it?
  – Fake car accidents. “Paper Collisions”: complete with fake drivers, fake passengers and fake witnesses

https://www.youtube.com/watch?v=vgE8Buqmry0
Insurance Fraud – “Whiplash for Cash”

• What is it?
  – Fake car accidents. “Paper Collisions”: complete with fake drivers, fake passengers and fake witnesses

https://www.youtube.com/watch?v=vgE8Buqmry0
Tax evasion fraud

- Nodes were companies
- Links: shared resources
- Shared resources:
  - Shared members of board
  - Shared employees
Corruption in Estonia

- Data about relations between people, organizations and public tenders.
- Corruption detection map into problem of finding network structures containing
  - 1) an n-clique containing a corrupted person linked to a victim organization.
  - 2) a network component with many paths between the victim and a beneficiary outside victim organization.
Types of models

Input

Logistic regression
Decision Tree

Output

Input

Black box
Random Forest
Boosting Algorithms

Output
Decision Tree

Step 1: Take the whole dataset

Step 2: Take the feature with more important information gain and divide the data

Step 3: Take the next feature with more important information gain and divide the data again and so on..

<table>
<thead>
<tr>
<th>F1</th>
<th>F2</th>
<th>F3</th>
<th>Salary</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>?</td>
</tr>
</tbody>
</table>
Random Forest

1) Multiple parallel Trees
2) Bootstrap dataset
3) Random Selection of features at every step
4) Voting

<table>
<thead>
<tr>
<th></th>
<th>F1</th>
<th>F2</th>
<th>F3</th>
<th>Salary</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

?
Gradient Boost builds fixed sized trees based on previous trees error.

Gradient Boost scales all the tree by the same amount.

Gradient Boost builds another tree based on the errors made by the previous tree.

Gradient Boost scales the tree.

Gradient Boost builds another tree based on the errors made by the previous tree.

Gradient Boost scales the tree.
Gradient Boosting Machine

Gradient Boost builds fixed sized trees based on previous trees error.

Gradient Boost scales all the tree by the same amount.

Gradient Boost builds another tree based on the errors made by the previous tree.

Gradient Boost scales the tree.

Gradient Boost builds another tree based on the errors made by the previous tree.

Gradient Boost scales the tree.

STOPING CONDITION: Gradient Boost continues to build trees in this fashion until
1) it has made the number of trees you asked for or
2) addition trees fail to improve the fit.
Gradient Boosting

First Attempt

<table>
<thead>
<tr>
<th>F1</th>
<th>F2</th>
<th>F3</th>
<th>Salary</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>88</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>76</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>56</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>73</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>77</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>57</td>
</tr>
</tbody>
</table>

Average salary = 71.2

Error of Tree1 = observed – predicted values
(Actual - (71.2) salary value)

Pseudo Residual

<table>
<thead>
<tr>
<th>Pseudo Residual</th>
</tr>
</thead>
<tbody>
<tr>
<td>16.8</td>
</tr>
<tr>
<td>4.8</td>
</tr>
<tr>
<td>-15.2</td>
</tr>
<tr>
<td>1.8</td>
</tr>
<tr>
<td>5.8</td>
</tr>
<tr>
<td>-14.2</td>
</tr>
</tbody>
</table>

NOTE: we are working on at the most 4 leaves (for simplicity). Generally this value is between 8 to 32.
Gradient Boosting

Build a tree to predict residual

<table>
<thead>
<tr>
<th>F1</th>
<th>F2</th>
<th>F3</th>
<th>Salary</th>
<th>Pseudo Residual</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>88</td>
<td>16.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>76</td>
<td>4.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>56</td>
<td>-15.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>73</td>
<td>1.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>77</td>
<td>5.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>57</td>
<td>-14.2</td>
</tr>
</tbody>
</table>

Average = -14.7  
Average = 3.8
Gradient Boosting

Build a tree to predict residual

<table>
<thead>
<tr>
<th>F1</th>
<th>F2</th>
<th>F3</th>
<th>Salary</th>
<th>Pseudo Residual</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>88</td>
<td>16.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>76</td>
<td>4.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>56</td>
<td>-15.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>73</td>
<td>1.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>77</td>
<td>5.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>57</td>
<td>-14.2</td>
</tr>
</tbody>
</table>
Gradient Boosting

Build a tree to predict residual

<table>
<thead>
<tr>
<th>F1</th>
<th>F2</th>
<th>F3</th>
<th>Salary</th>
<th>Pseudo Residual</th>
</tr>
</thead>
<tbody>
<tr>
<td>--</td>
<td>--</td>
<td>--</td>
<td>88</td>
<td>16.8</td>
</tr>
<tr>
<td>76</td>
<td></td>
<td></td>
<td></td>
<td>4.8</td>
</tr>
<tr>
<td>56</td>
<td></td>
<td></td>
<td></td>
<td>-15.2</td>
</tr>
<tr>
<td>73</td>
<td></td>
<td></td>
<td></td>
<td>1.8</td>
</tr>
<tr>
<td>77</td>
<td></td>
<td></td>
<td></td>
<td>5.8</td>
</tr>
<tr>
<td>57</td>
<td></td>
<td></td>
<td></td>
<td>-14.2</td>
</tr>
</tbody>
</table>

Predicted weight = Average weight + 16.8 = 88

**NOTE**: Low Bias but high variance (overfitting)
Gradient Boosting

Build a tree to predict residual

<table>
<thead>
<tr>
<th>F1</th>
<th>F2</th>
<th>F3</th>
<th>Salary</th>
<th>Pseudo Residual</th>
</tr>
</thead>
<tbody>
<tr>
<td>--</td>
<td>--</td>
<td>--</td>
<td>88</td>
<td>16.8</td>
</tr>
<tr>
<td></td>
<td>76</td>
<td></td>
<td>4.8</td>
<td></td>
</tr>
<tr>
<td>56</td>
<td></td>
<td></td>
<td>-15.2</td>
<td></td>
</tr>
<tr>
<td>73</td>
<td></td>
<td></td>
<td>1.8</td>
<td></td>
</tr>
<tr>
<td>77</td>
<td></td>
<td></td>
<td>5.8</td>
<td></td>
</tr>
<tr>
<td>57</td>
<td></td>
<td></td>
<td>-14.2</td>
<td></td>
</tr>
</tbody>
</table>

Predicted weight = Average weight + θ*16.8

θ is the learning rate, to scale the contribution from the new tree and has a value between 0 and 1.
### Gradient Boosting

<table>
<thead>
<tr>
<th>F1</th>
<th>F2</th>
<th>F3</th>
<th>Salary</th>
<th>Pseudo Residual</th>
</tr>
</thead>
<tbody>
<tr>
<td>--</td>
<td>--</td>
<td>--</td>
<td>88</td>
<td>16.8</td>
</tr>
<tr>
<td>76</td>
<td></td>
<td></td>
<td></td>
<td>4.8</td>
</tr>
<tr>
<td>56</td>
<td></td>
<td></td>
<td></td>
<td>-15.2</td>
</tr>
<tr>
<td>73</td>
<td></td>
<td></td>
<td>5.8</td>
<td>1.8</td>
</tr>
<tr>
<td>77</td>
<td></td>
<td></td>
<td>5.8</td>
<td>-14.2</td>
</tr>
<tr>
<td>57</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Predicted weight = Average weight + $\Theta \times 16.8$

Let $\Theta = 0.1$

Predicted weight = $71.2 + 0.1 \times 16.8 = 72.9$
Gradient Boosting

Build a tree to predict residual

<table>
<thead>
<tr>
<th>F1</th>
<th>F2</th>
<th>F3</th>
<th>Salary</th>
<th>PR1</th>
<th>PR2</th>
</tr>
</thead>
<tbody>
<tr>
<td>--</td>
<td>--</td>
<td>--</td>
<td>88</td>
<td>16.8</td>
<td>15.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>76</td>
<td>4.8</td>
<td>4.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>56</td>
<td>-15.2</td>
<td>-13.7</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>73</td>
<td>1.8</td>
<td>1.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>77</td>
<td>5.8</td>
<td>5.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>57</td>
<td>-14.2</td>
<td>-12.7</td>
</tr>
</tbody>
</table>

Let $\Theta = 0.1$
Predicted weight = $71.2 + 0.1 \times 16.8 = 72.9$
New residual error = $88 - 72.9 = 15.1$

And so on ...
Gradient Boosting

<table>
<thead>
<tr>
<th>F1</th>
<th>F2</th>
<th>F3</th>
<th>Salary</th>
<th>PR1</th>
<th>PR2</th>
</tr>
</thead>
<tbody>
<tr>
<td>--</td>
<td>--</td>
<td>--</td>
<td>88</td>
<td>16.8</td>
<td>15.1</td>
</tr>
<tr>
<td>76</td>
<td>4.8</td>
<td>4.3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>-15.2</td>
<td>-13.7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>73</td>
<td>1.8</td>
<td>1.4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>77</td>
<td>5.8</td>
<td>5.4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>57</td>
<td>-14.2</td>
<td>-12.7</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Build a new tree to predict residual

Predicted weight = Average weight + Θ*16.8 + Θ*15.1
Gradient Boosting

Build a tree to predict residual

<table>
<thead>
<tr>
<th>F1</th>
<th>F2</th>
<th>F3</th>
<th>Salary</th>
<th>PR1</th>
<th>PR2</th>
</tr>
</thead>
<tbody>
<tr>
<td>--</td>
<td>--</td>
<td>--</td>
<td>88</td>
<td>16.8</td>
<td>15.1</td>
</tr>
<tr>
<td>76</td>
<td>4.8</td>
<td>4.3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>-15.2</td>
<td>-13.7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>73</td>
<td>1.8</td>
<td>1.4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>77</td>
<td>5.8</td>
<td>5.4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>57</td>
<td>-14.2</td>
<td>-12.7</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Predicted weight = Average weight + θ*16.8 + θ*15.1
Each time we add a tree to the Prediction, the Residuals get smaller.
Bagging and Boosting

• **Bagging:**
  - **parallel** ensemble: each model is built independently
  - aim to **decrease variance**, not bias
  - suitable for high variance low bias models (complex models)
  - an example of a tree based method is **random forest**, which develop fully grown trees (note that RF modifies the grown procedure to reduce the correlation between trees)

• **Boosting:**
  - **sequential** ensemble: try to add new models that do well where previous models lack
  - aim to **decrease bias**, not variance
  - suitable for low variance high bias models
  - an example of a tree based method is **gradient boosting**

sources:
XGBoost (**Extreme Gradient Boosting**): library (a type of implementation)

- Optimized distributed gradient boosting library.
- Uses gradient boosting (GBM) framework at core
- Does better than GBM framework alone
- XGBoost belongs to a family of boosting algorithms that convert weak learners into strong learners.
- A weak learner is one which is slightly better than random guessing.
- Boosting is a sequential process; i.e., trees are grown using the information from a previously grown tree one after the other.
- This process slowly learns from data and tries to improve its prediction in subsequent iterations.

Idea behind XGBoost (or boosting in general)

Advantages of XGBoost

- **Parallel Computing:** It is enabled with parallel processing (using OpenMP); i.e., when you run xgboost, by default, it would use all the cores of your laptop/machine.

- **Regularization:** I believe this is the biggest advantage of xgboost. GBM has no provision for regularization. Regularization is a technique used to avoid overfitting in linear and tree-based models.

- **Enabled Cross Validation:** In R, we usually use external packages such as caret and mlr to obtain CV results. But, xgboost is enabled with internal CV function (we’ll see below).

- **Missing Values:** XGBoost is designed to handle missing values internally. The missing values are treated in such a manner that if there exists any trend in missing values, it is captured by the model.

- **Flexibility:** In addition to regression, classification, and ranking problems, it supports user-defined objective functions also. An objective function is used to measure the performance of the model given a certain set of parameters. Furthermore, it supports user defined evaluation metrics as well.

- **Availability:** Currently, it is available for programming languages such as R, Python, Java, Julia, and Scala.

- **Save and Reload:** XGBoost gives us a feature to save our data matrix and model and reload it later.

- **Tree Pruning:** XGBoost grows the tree upto max_depth and then prune backward until the improvement in loss function is below a threshold.

Demo time!

https://courses.cs.ut.ee/2019/bda/spring/Main/Practice