Anomaly Detection
Finding the Unusual

Poul Petersen
CIO, BigML, Inc
What is Anomaly Detection?

- An **unsupervised** learning technique
  - No labels necessary
  - Useful for finding unusual instances
  - Filtering, finding mistakes, 1-class classifiers
- Finds instances that do not match
  - Customer: big or small spender for profile
  - Medical: healthy patient despite indicative diagnostics
- Defines each unusual instance by an “**anomaly score**”
  - in BigML: 0 = normal, 1 = unusual, and 0.7 ≫ 0.6 > 0.5
  - Standard deviation, distributions, etc
## Clusters

<table>
<thead>
<tr>
<th>date</th>
<th>customer</th>
<th>account</th>
<th>auth</th>
<th>class</th>
<th>zip</th>
<th>amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mon</td>
<td>Bob</td>
<td>3421</td>
<td>pin</td>
<td>clothes</td>
<td>46140</td>
<td>135</td>
</tr>
<tr>
<td>Tue</td>
<td>Bob</td>
<td>3421</td>
<td>sign</td>
<td>food</td>
<td>46140</td>
<td>401</td>
</tr>
<tr>
<td>Tue</td>
<td>Alice</td>
<td>2456</td>
<td>pin</td>
<td>food</td>
<td>12222</td>
<td>234</td>
</tr>
<tr>
<td>Wed</td>
<td>Sally</td>
<td>6788</td>
<td>pin</td>
<td>gas</td>
<td>26339</td>
<td>94</td>
</tr>
<tr>
<td>Wed</td>
<td>Bob</td>
<td>3421</td>
<td>pin</td>
<td>tech</td>
<td>21350</td>
<td>2459</td>
</tr>
<tr>
<td>Wed</td>
<td>Bob</td>
<td>3421</td>
<td>pin</td>
<td>gas</td>
<td>46140</td>
<td>83</td>
</tr>
<tr>
<td>Thr</td>
<td>Sally</td>
<td>6788</td>
<td>sign</td>
<td>food</td>
<td>26339</td>
<td>51</td>
</tr>
</tbody>
</table>
Clusters

<table>
<thead>
<tr>
<th>date</th>
<th>customer</th>
<th>account</th>
<th>auth</th>
<th>class</th>
<th>zip</th>
<th>amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mon</td>
<td>Bob</td>
<td>3421</td>
<td>pin</td>
<td>clothes</td>
<td>46140</td>
<td>135</td>
</tr>
<tr>
<td>Tue</td>
<td>Bob</td>
<td>3421</td>
<td>sign</td>
<td>food</td>
<td>46140</td>
<td>401</td>
</tr>
<tr>
<td>Tue</td>
<td>Alice</td>
<td>2456</td>
<td>pin</td>
<td>food</td>
<td>12222</td>
<td>234</td>
</tr>
<tr>
<td>Wed</td>
<td>Sally</td>
<td>6788</td>
<td>pin</td>
<td>gas</td>
<td>26339</td>
<td>94</td>
</tr>
<tr>
<td>Wed</td>
<td>Bob</td>
<td>3421</td>
<td>pin</td>
<td>tech</td>
<td>21350</td>
<td>2459</td>
</tr>
<tr>
<td>Wed</td>
<td>Bob</td>
<td>3421</td>
<td>pin</td>
<td>gas</td>
<td>46140</td>
<td>83</td>
</tr>
<tr>
<td>Thr</td>
<td>Sally</td>
<td>6788</td>
<td>sign</td>
<td>food</td>
<td>26339</td>
<td>51</td>
</tr>
</tbody>
</table>
## Anomaly Detection

<table>
<thead>
<tr>
<th>date</th>
<th>customer</th>
<th>account</th>
<th>auth</th>
<th>class</th>
<th>zip</th>
<th>amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mon</td>
<td>Bob</td>
<td>3421</td>
<td>pin</td>
<td>clothes</td>
<td>46140</td>
<td>135</td>
</tr>
<tr>
<td>Tue</td>
<td>Bob</td>
<td>3421</td>
<td>sign</td>
<td>food</td>
<td>46140</td>
<td>401</td>
</tr>
<tr>
<td>Tue</td>
<td>Alice</td>
<td>2456</td>
<td>pin</td>
<td>food</td>
<td>12222</td>
<td>234</td>
</tr>
<tr>
<td>Wed</td>
<td>Sally</td>
<td>6788</td>
<td>pin</td>
<td>gas</td>
<td>26339</td>
<td>94</td>
</tr>
<tr>
<td>Wed</td>
<td>Bob</td>
<td>3421</td>
<td>pin</td>
<td>tech</td>
<td>21350</td>
<td>2459</td>
</tr>
<tr>
<td>Wed</td>
<td>Bob</td>
<td>3421</td>
<td>pin</td>
<td>gas</td>
<td>46140</td>
<td>83</td>
</tr>
<tr>
<td>Thr</td>
<td>Sally</td>
<td>6788</td>
<td>sign</td>
<td>food</td>
<td>26339</td>
<td>51</td>
</tr>
</tbody>
</table>
Anomaly Detection

<table>
<thead>
<tr>
<th>date</th>
<th>customer</th>
<th>account</th>
<th>auth</th>
<th>class</th>
<th>zip</th>
<th>amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mon</td>
<td>Bob</td>
<td>3421</td>
<td>pin</td>
<td>clothes</td>
<td>46140</td>
<td>135</td>
</tr>
<tr>
<td>Tue</td>
<td>Bob</td>
<td>3421</td>
<td>sign</td>
<td>food</td>
<td>46140</td>
<td>401</td>
</tr>
<tr>
<td>Tue</td>
<td>Alice</td>
<td>2456</td>
<td>pin</td>
<td>food</td>
<td>12222</td>
<td>234</td>
</tr>
<tr>
<td>Wed</td>
<td>Sally</td>
<td>6788</td>
<td>pin</td>
<td>gas</td>
<td>26339</td>
<td>94</td>
</tr>
<tr>
<td>Wed</td>
<td>Bob</td>
<td>3421</td>
<td>pin</td>
<td>tech</td>
<td>21350</td>
<td>2459</td>
</tr>
<tr>
<td>Wed</td>
<td>Bob</td>
<td>3421</td>
<td>pin</td>
<td>gas</td>
<td>46140</td>
<td>83</td>
</tr>
<tr>
<td>Thr</td>
<td>Sally</td>
<td>6788</td>
<td>sign</td>
<td>food</td>
<td>26339</td>
<td>51</td>
</tr>
</tbody>
</table>

- Amount $2,459 is higher than all other transactions
- It is the only transaction
  - In zip 21350
  - for the purchase class "tech"
Use Cases

- Unusual instance discovery - "exploration"
- Intrusion Detection - "looking for unusual usage patterns"
Intrusion Detection

GOAL: Identify unusual command line behavior per user and across all users that might indicate an intrusion.

- Dataset of command line history for users
- Data for each user consists of commands, flags, working directories, etc.
- Assumption: Users typically issue the same flag patterns and work in certain directories
Use Cases

- Unusual instance discovery - "exploration"
- Intrusion Detection - "looking for unusual usage patterns"
- Fraud - "looking for unusual behavior"
Fraud

- Dataset of credit card transactions
- Additional user profile information

**GOAL:** Cluster users by profile and use multiple anomaly scores to detect transactions that are anomalous on multiple levels.
Use Cases

- Unusual instance discovery - "exploration"
- Intrusion Detection - "looking for unusual usage patterns"
- Fraud - "looking for unusual behavior"
- Identify Incorrect Data - "looking for mistakes"
- Remove Outliers - "improve model quality"
Removing Outliers

- Models need to generalize
- Outliers negatively impact generalization

**GOAL**: Use anomaly detector to identify most anomalous points and then remove them before modeling.
Use Cases

- Unusual instance discovery - "exploration"
- Intrusion Detection - "looking for unusual usage patterns"
- Fraud - "looking for unusual behavior"
- Identify Incorrect Data - "looking for mistakes"
- Remove Outliers - "improve model quality"
- Model Competence / Input Data Drift
After putting a model into production, data that is being predicted can become statistically different than the training data.

Train an anomaly detector at the same time as the model.

**GOAL**: For every prediction, compute an anomaly score. If the anomaly score is high, then the model may not be competent and should not be trusted.
Benford’s Law

- In real-life numeric sets the small digits occur disproportionately often as leading significant digits.

- Applications include:
  - accounting records
  - electricity bills
  - street addresses
  - stock prices
  - population numbers
  - death rates
  - lengths of rivers

- Available in BigML API
Univariate Approach

• Single variable: heights, test scores, etc

• Assume the value is distributed “normally”

• Compute standard deviation

• a measure of how “spread out” the numbers are

• the square root of the variance (The average of the squared differences from the Mean.)

• Depending on the number of instances, choose a “multiple” of standard deviations to indicate an anomaly. A multiple of 3 for 1000 instances removes ~ 3 outliers.
Univariate Approach

- Available in BigML API
Multivariate Matters
Multivariate Matters
Human Expert

Most Unusual?
Human Expert

“Skinny” but not “smooth”

Most unusual

Key Insight
The “most unusual” object is different in some way from every partition of the features.
• Human used prior knowledge to select possible features that separated the objects.

• “round”, “skinny”, “smooth”, “corners”

• Items were then separated based on the chosen features.

• Each cluster was then examined to see which object fit the least well in its cluster and did not fit any other cluster.
Create **features** that capture these object differences

- **Length/Width**
  - greater than 1 => “skinny”
  - equal to 1 => “round”
  - less than 1 => invert

- **Number of Surfaces**
  - distinct surfaces require “edges” which have corners
  - easier to count

- **Smooth** - true or false
# Anomaly Features

<table>
<thead>
<tr>
<th>Object</th>
<th>Length / Width</th>
<th>Num Surfaces</th>
<th>Smooth</th>
</tr>
</thead>
<tbody>
<tr>
<td>penny</td>
<td>1</td>
<td>3</td>
<td>TRUE</td>
</tr>
<tr>
<td>dime</td>
<td>1</td>
<td>3</td>
<td>TRUE</td>
</tr>
<tr>
<td>knob</td>
<td>1</td>
<td>4</td>
<td>TRUE</td>
</tr>
<tr>
<td>eraser</td>
<td>2.75</td>
<td>6</td>
<td>TRUE</td>
</tr>
<tr>
<td>box</td>
<td>1</td>
<td>6</td>
<td>TRUE</td>
</tr>
<tr>
<td>block</td>
<td>1.6</td>
<td>6</td>
<td>TRUE</td>
</tr>
<tr>
<td>screw</td>
<td>8</td>
<td>3</td>
<td>FALSE</td>
</tr>
<tr>
<td>battery</td>
<td>5</td>
<td>3</td>
<td>TRUE</td>
</tr>
<tr>
<td>key</td>
<td>4.25</td>
<td>3</td>
<td>FALSE</td>
</tr>
<tr>
<td>bead</td>
<td>1</td>
<td>2</td>
<td>TRUE</td>
</tr>
</tbody>
</table>
Random Splits

Know that “splits” matter - don’t know the order

smooth?
Isolation Forest

Grow a random decision tree until each instance from a sample is in its own leaf

“easy” to isolate

Depth

“hard” to isolate

Now repeat the process several times and use average Depth to compute anomaly score: 0 (similar) -> 1 (dissimilar)
### Isolation Forest Scoring

#### For the instance, \( i_2 \)

<table>
<thead>
<tr>
<th></th>
<th>( f_1 )</th>
<th>( f_2 )</th>
<th>( f_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( i_1 )</td>
<td>red</td>
<td>cat</td>
<td>ball</td>
</tr>
<tr>
<td>( i_2 )</td>
<td>red</td>
<td>cat</td>
<td>ball</td>
</tr>
<tr>
<td>( i_3 )</td>
<td>red</td>
<td>cat</td>
<td>box</td>
</tr>
<tr>
<td>( i_4 )</td>
<td>blue</td>
<td>dog</td>
<td>pen</td>
</tr>
</tbody>
</table>

- For \( D = 3 \)
- For \( D = 6 \)
- For \( D = 2 \)

Find the depth in each tree

**Map avg depth to final score**

- \( S = 0.45 \)
Anomaly Demo
Your Turn!

- Build an Anomaly Detector for the **Diabetes 80% Training** set
- Filter out the **top 2 Anomalies**
- Build a “Clean” model from the new Dataset and Evaluate it
- Which performs better?
1-Class Classifier?

- You place an advertisement in a local newspaper
- You collect demographic information about all responders
- Now you want to market in a new locality with direct letters
- To optimize mailing costs, need to predict who will respond
- But, can not distinguish not interested from didn’t see the ad
- Train an anomaly detector on the 1-class data
- Pick the households with the lowest scores for mailing:
  - If a household has a low anomaly score, then they are “similar” to enough of your positive responders and therefore may respond as well
  - If an individual has a high anomaly score, then they are dissimilar from all previous responders and therefore are less likely to respond.