Preemptive Shipping
How Gilt Predicts Which Customers Will Buy Products It Has Never Sold Before

Igor Elbert, Principal Data Scientist, Gilt.com
Before Gilt – sample sales
Gilt pioneered online “flash sales” in US
Can we shorten shipping time?

Shipping times from Las Vegas

Shipping times from Kentucky

What if we move items closer to potential buyers?
Goal: Predict which items will be sold ‘West’

How is it different from what retailers have always been doing?
Challenges:

- Volatile preferences, context
- Regional differences
Challenges:

• Narrow decision window

• No sales history for new products
But we have a lot of data…

• Orders
• Product Information
• Clickstream

and tools to handle it:
What would be a good model?

For this use-case Precision is more important than Recall.

<table>
<thead>
<tr>
<th>Actual</th>
<th>Predicted</th>
</tr>
</thead>
<tbody>
<tr>
<td>‘West’</td>
<td>‘West’</td>
</tr>
<tr>
<td>‘Non-West’</td>
<td>‘non-West’</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>‘West’</td>
<td>50 😊</td>
</tr>
<tr>
<td>‘Non-West’</td>
<td>15 😞</td>
</tr>
<tr>
<td>‘West’</td>
<td>15 🙁</td>
</tr>
<tr>
<td>‘Non-West’</td>
<td>20</td>
</tr>
</tbody>
</table>
What would be a good model?

For this use-case Precision is more important than Recall

Maximize ratio of True-Positives to False-Positives-No-Sale

<table>
<thead>
<tr>
<th></th>
<th>Predicted</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>‘West’</td>
<td>‘East’</td>
<td>No-sale</td>
</tr>
<tr>
<td>‘West’</td>
<td>50 😊</td>
<td>6 😞</td>
<td>9 😞</td>
</tr>
<tr>
<td>‘East’</td>
<td>3 $ 😞</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>No-sale</td>
<td>12 $$$</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>
Propensity thresholds
Initial approach:

Color
Size
Material
Brand
Category
Price
Margin
Discount
Other product metadata
Data Mining and the Big Picture

- Random Forests (various implementations)
- Weighted Subspace Random Forest
- AdaBoost
- NNet
- Others
Bigger picture – Data is the King

a. Dealing with high cardinality
b. When throwing away data is a good thing
c. Wonders of cheating
d. Let’s be subjective here
High cardinality attributes

Error in randomForest.default(m, y, ...) :
Can not handle categorical predictors with more than 53 categories.

Reducing cardinality:
  0. **Clean thy data:**

```sql
REGEXP_REPLACE(TRIM(COALESCE(LOWER(material), '')), '
  (elastace|elatsane|elastine?|elastan[^e]|elastan$)',
  'elastane', 'g')
```

1. **Clean it again:** Levenshtein Distances:

<table>
<thead>
<tr>
<th>target_cnt</th>
<th>source_cnt</th>
<th>target</th>
<th>source</th>
<th>distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>1</td>
<td>100% nylon</td>
<td>1005 nylon</td>
<td>1</td>
</tr>
<tr>
<td>1368</td>
<td>3</td>
<td>100% cotton</td>
<td>100 % cotton</td>
<td>1</td>
</tr>
<tr>
<td>60</td>
<td>4</td>
<td>95% cotton 5% elastane 95 cotton 5% elastane</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1368</td>
<td>4</td>
<td>100% cotton</td>
<td>1005 cotton</td>
<td>1</td>
</tr>
<tr>
<td>....</td>
<td>....</td>
<td>....</td>
<td>....</td>
<td>....</td>
</tr>
<tr>
<td>21</td>
<td>11</td>
<td>leather suede</td>
<td>leather or suede</td>
<td>3</td>
</tr>
<tr>
<td>80</td>
<td>15</td>
<td>suede leather</td>
<td>suede or leather</td>
<td>3</td>
</tr>
</tbody>
</table>
High cardinality attributes (cont.)

2. **Cut the long tail.** Algorithm specific.  
   If algorithm does not complain about hard cardinality it does not mean it benefits from it. Use either top N attribute values by number of cases or as many values as needed to cover X% of cases.

3. **Cheat** – picking into test data is now allowed.  
   **Before:** Model training took a long time. So model was built to score many data sets.  
   **Now:** Model training is quick. It can be adopted to the testing set. Knowing which values are in the test set we can build better model.

4. **Slice the data set** – each subset will have its own set of attribute values

5. **Rough it up:** ‘royal blue’ is ‘blue’  
   Clustering helps
…helps with new brands too
‘Subjective’ attributes:

- Who is Gilt’s buyer for the item?
- Who curated the sale?
- Who set the price?
- Who is the photographer?
- ...

Side effects: business insights, best practices

(use simple methods do describe and explain)
More subjectivity: asking strangers for data

Template for Amazon’s Mechanical Turk
Using Mechanical Turk:

• Good quality responses

• Quick turn-around

• Produced several good predictors
  - expected and unexpected

• User-generated content (waiting or analysis):

  “This dress looks like it would work well mainly for hourglass figures. The banding looks very atypical, so it would probably get the dress more attention than without the banding.”

• Easy to automate (MTurkR R package)
This dress will sell
.... on a good day

- Day of the week
- Day of the year
  - previous, next holiday
- What else is on sale – possible ‘halo effect’
- Total number of items on sale
- Price line-up
- Proxies of traffic – more visitors, higher chances to sell
- Time of sale – competing with other shoppers
Rewards:

- Dry runs look promising – we expect significant reduction in days-in-transit
- Useful side effects:
  - Propensity to sell – pick-and-pack optimization
  - Propensity to not sell – insights on pricing, merchandizing, inventory
- ‘Stress-test’ for company’s logistics – order routing, sale creation, shipping

Conclusions:

- Shop at Gilt - we need data
- See big picture
- Clean and trim the data
- Use subjective data
Questions?

Igor Elbert
ielbert@gilt.com