What is Topic Modeling?

- Unsupervised algorithm
- Learns only from text fields
- Finds hidden topics that model the text

Questions:
- How is this different from the Text Analysis that BigML already offers?
- What does it output and how do we use it?
- Unsupervised... model?
Text Analysis

1. Stem Words -> Tokens
2. Remove tokens that occur too often
3. Remove tokens that do not occur often enough
4. Count occurrences of remaining “interesting” tokens

**great**: appears 4 times
Be not afraid of greatness: some are born great, some achieve greatness, and some have greatness thrust upon ‘em.

The token “great” occurs more than 3 times

The token “afraid” occurs no more than once
Hodor!
Topic Model Demo #1
TA vs TM

Text Analysis

- Creates thousands of hidden token counts
- Token counts are independently uninteresting
- No semantic importance
- No measure of co-occurrence

Topic Model

- Creates tens of topics that model the text
- Topics are independently interesting
- Semantic meaning extracted
- Support for bigrams
Generating Documents

• "Machine" that generates a random word with equal probability with each pull.

• Pull random number of times to generate a document.

• All documents can be generated, but most are nonsense.
Intuition:  
• Written documents have meaning - one way to describe meaning is to assign a topic.
• For our random machine, the topic can be thought of as increasing the probability of certain words.

<table>
<thead>
<tr>
<th>Topic: travel</th>
</tr>
</thead>
<tbody>
<tr>
<td>word</td>
</tr>
<tr>
<td>travel</td>
</tr>
<tr>
<td>airplane</td>
</tr>
<tr>
<td>mars</td>
</tr>
<tr>
<td>mantle</td>
</tr>
<tr>
<td>...</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Topic: space</th>
</tr>
</thead>
<tbody>
<tr>
<td>word</td>
</tr>
<tr>
<td>space</td>
</tr>
<tr>
<td>airplane</td>
</tr>
<tr>
<td>mars</td>
</tr>
<tr>
<td>mantle</td>
</tr>
<tr>
<td>...</td>
</tr>
</tbody>
</table>
• Each text field in a row is concatenated into a document

• The documents are analyzed to generate "k" related topics

• Each topic is represented by a distribution of term probabilities

1. Each text field in a row is concatenated into a document.
2. The documents are analyzed to generate "k" related topics.
3. Each topic is represented by a distribution of term probabilities.
Topic Model Demo #2
Intuition:  
- Any given document is likely a mixture of the modeled topics...
- This can be represented as a distribution of topic probabilities

**Topic: travel**
- Word: travel, probability: 23.55%
- Word: airplane, probability: 2.33%
- Word: mars, probability: 0.003%
- Word: mantle, probability: ε
- ... probability: ε

**Topic: space**
- Word: space, probability: 38.94%
- Word: airplane, probability: ε
- Word: mars, probability: 13.43%
- Word: mantle, probability: 0.05%
- ... probability: ε

Will 2020 be the year that humans will embrace space exploration and finally travel to Mars?
Topic Model Demo #3
Clustering?

Unlabelled Data → Clustering → Batch Centroid

Text Fields

Unlabelled Data → Topic Model → Batch Topic Distribution

Prob

Prob

Prob

Prob
Topic Model Demo #4
Topic Model Use Cases

- As a preprocessor for other techniques
  - Building better models
- Bootstrapping categories for classification
- Recommendation
- Discovery in large, heterogeneous text datasets
Topic Model Tips

• Setting $k$
  • Much like k-means, the best value is data specific
  • Too few will agglomerate unrelated topics, too many will partition highly related topics
  • I tend to find the latter more annoying than the former

• Tuning the Model
  • Remove common, useless terms
  • Set term limit higher, use bigrams
Your Turn!

- Create a Source and a Dataset from the StumbleUpon tsv
- Configure a Topic Model (not a 1-click) using:
  - Maximum n-grams=2
  - Exclude non-dictionary words
  - Exclude non-language characters
  - Removing HTML tags
  - Exclude numeric digits
- What is the primary topic for the phrase boilerplate = “No soup for you!”