Advanced ML in Google Cloud (2)

Abhay Agarwal (MS Design ‘19)
Agenda

- ‘Productizing’ analytics
- Data wrangling
- Data fundamentals
- Data studio vs datalab vs colab
‘Productizing’

● What does it mean to ‘productize’ your ML?
Pitfalls in Productizing

- My algorithm has a 95% accuracy -- is it ready for production?
- My algorithm has a 95% accuracy and 95% precision -- is it ready for production?
- My algorithm has a 95% accuracy, 95% precision, and my training data is roughly sampled from real examples -- is it ready for production?
- My algorithm has a 95% accuracy, 95% precision, training data sampled from real examples, and my algorithm tests hypotheses that match the use cases -- is it ready for production?
Data wrangling
DATA COLLECTION FUNDAMENTALS
Key Concepts

Quantity

Quality

Cost

Freshness

Structure
Quantity

• Breadth
  • Number of entities or observations
  • E.g., People, companies, stars, shopping trips,…
  • Ideally: comprehensive

• Depth
  • Data gathered on each entity or observation
### World Bank Development Indicators

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>millions</td>
<td>sq. km thousands</td>
<td>people per sq. km</td>
<td>$ billions 2017</td>
<td>$ billions 2017</td>
<td>$ billions 2017</td>
<td>$ billions 2017</td>
</tr>
<tr>
<td>Afghanistan</td>
<td>35.5</td>
<td>652.9</td>
<td>54</td>
<td>29.2</td>
<td>570</td>
<td>71.1</td>
<td>2,000</td>
</tr>
<tr>
<td>Albania</td>
<td>2.9</td>
<td>28.8</td>
<td>105</td>
<td>12.4</td>
<td>4,320</td>
<td>34.8</td>
<td>12,120</td>
</tr>
<tr>
<td>Algeria</td>
<td>41.3</td>
<td>2,381.7</td>
<td>17</td>
<td>163.5</td>
<td>3,960</td>
<td>621.9</td>
<td>15,050</td>
</tr>
<tr>
<td>American Samoa</td>
<td>0.1</td>
<td>0.2</td>
<td>278</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>Andorra</td>
<td>0.1</td>
<td>0.5</td>
<td>164</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>Argentina</td>
<td>29.8</td>
<td>1,246.7</td>
<td>24</td>
<td>991</td>
<td>3,330</td>
<td>180.5</td>
<td>6,090</td>
</tr>
<tr>
<td>Aruba</td>
<td>0.1</td>
<td>0.4</td>
<td>232</td>
<td>1.4</td>
<td>14,170</td>
<td>2.3</td>
<td>22,980</td>
</tr>
<tr>
<td>Armena</td>
<td>2.9</td>
<td>29.7</td>
<td>103</td>
<td>11.7</td>
<td>4,000</td>
<td>29.5</td>
<td>10,060</td>
</tr>
<tr>
<td>Austria</td>
<td>8.8</td>
<td>83.9</td>
<td>107</td>
<td>1,263.5</td>
<td>51,360</td>
<td>1,160</td>
<td>47,160</td>
</tr>
<tr>
<td>Azerbaijan</td>
<td>9.9</td>
<td>86.6</td>
<td>119</td>
<td>402</td>
<td>4,080</td>
<td>164.2</td>
<td>16,650</td>
</tr>
<tr>
<td>Bahamas, The</td>
<td>0.4</td>
<td>13.9</td>
<td>39</td>
<td>11.5</td>
<td>28,170</td>
<td>11.8</td>
<td>29,790</td>
</tr>
<tr>
<td>Bahrain</td>
<td>1.5</td>
<td>0.8</td>
<td>1,536</td>
<td>192</td>
<td>20,240</td>
<td>64.1</td>
<td>42,930</td>
</tr>
<tr>
<td>Bangladesh</td>
<td>164.7</td>
<td>147.6</td>
<td>1,265</td>
<td>242.8</td>
<td>1,470</td>
<td>664.5</td>
<td>4,040</td>
</tr>
<tr>
<td>Barbados</td>
<td>0.3</td>
<td>0.4</td>
<td>664</td>
<td>44</td>
<td>15,540</td>
<td>5.1</td>
<td>17,830</td>
</tr>
</tbody>
</table>
Graph Data

Graphs arise naturally in many settings

Many interesting techniques
e.g., Page Rank, community detection
Data Quality

- **Errors**
  - E.g., human labeling mistakes

- **Missing data**
  - E.g., missing addresses in customer records

- **Bias**
  - Sample bias, measurement bias, prejudice/stereotype
Data Quality: Sample Bias

Day Driving vs Night Driving

Tank recognition
Data Quality: Prejudice/Stereotype Bias
Algorithmic Law Enforcement

But what about perpetuating bias against minorities?
Data Quality: Measurement Bias
Data Freshness

Rate of data collection must match rate of change of underlying phenomenon
Data manipulation in Google Cloud

- Data Studio
- Datalab
- Colab
- (offline!)
Data Studio

- Data Studio - glorified spreadsheets with a few integrations to Google Cloud to pull data
- Use cases: excel-like functions, simple visualizations (e.g. geographic)
Datalab

- Datalab - hosted Jupyter instance with preset libraries
- Use cases: python scripting, visualization, ML pipelining, some long-running scripting, versioned scripts and models
Colab

- Colab - Shared, no-setup version of Datalab that is designed around sharing
- Use cases: creating publicly accessible work, collaboration, but no long-running scripting