Learning and Prediction over Light-Weight Spatio-Temporal Data

BlackSwan Team
chenyitian@jd.com

August 16, 2017
Yang Guo, Yitian Chen, Research engineer@JD.COM

Jie Lin, Master Candidate, Nanjing University of Science and Technology

Hao Lin, Research engineer@Tencent

Jie Zhou, PHD Candidate, East China Normal University
Motivation

- Help authorities do better decision making.
- Expedite the toll collection process.
- Streamline future traffic flow and reduce congestion.
**Task Formulation**

**Task:** Given 5 tollgate-direction pairs and previous two-hour vehicle records, predict the traffic volume of every 20-minute time window for the next 2 hours.

**Data**

- **Testing days:** previous 2-hour vehicle records.
- **Training days:** vehicle records of all days (24 hours).
- **Weather:** humidity, precipitation, wind....

**Note:** the traffic volume for a given tollgate-direction pair is the total volume of all vehicles that enter/exit the tollgate in that time window. Each 20-minute time window is defined as a right half-open interval, e.g., [2016-09-21 8:00:00, 2016-09-21 8:20:00).
Basic ML approach: Use previous 6 20-minute time-window volume points to predict the next 6 points.

\[ X = \begin{bmatrix}
V_{[7:40,8:00]} & V_{[7:20,7:40]} & \cdots & V_{[6:00,6:20]} & \text{tsDistance} & \text{otherFeatures} \\
V_{[7:40,8:00]} & V_{[7:20,7:40]} & \cdots & V_{[6:00,6:20]} & 20 & \cdots \\
V_{[7:40,8:00]} & V_{[7:20,7:40]} & \cdots & V_{[6:00,6:20]} & 40 & \cdots \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
V_{[7:40,8:00]} & V_{[7:20,7:40]} & \cdots & V_{[6:00,6:20]} & 120 & \cdots \\
\end{bmatrix}, \quad Y = \begin{bmatrix}
V_{[8:00,8:20]} \\
V_{[8:20,8:40]} \\
V_{[8:40,9:00]} \\
\cdots \\
V_{[9:40,10:00]} \\
\end{bmatrix} \]

Objective: minimize the \( \text{MAPE}(\hat{Y}, Y) : \frac{1}{n} \sum_{t=1}^{n} \left| \frac{Y_t - \hat{Y}_t}{Y_t} \right| \). Works to do:

- Feature engineering: transfer raw data into features better represent the problem (Refer GIT for details).
- Modeling: design model framework/policy given the specific problem (focused in this talk).

GIT: https://github.com/12190143/Black-Swan
Challenges

Challenge-1 The vehicle volume of a route varies a lot depending on.
  ▶ Time of day.
  ▶ Day of the week.
  ▶ Holidays vs normal days.

Challenge-2 Small dataset: Only 29 (36 for stage 2) days’ data of 5 tollgate-direction pairs is provided. And it’s very noisy.

Challenge-3 Evaluation metrics (MAPE): Most regression loss functions do not minimize APE(Absolute percentage error) directly.
  ▶ MSE (Gaussian distribution): \( \text{loss} = \frac{1}{2}(y - \hat{y})^2 \).
  ▶ MAE (Laplace/Quantile distribution): \( \text{loss} = |y - \hat{y}| \).
**Strategy**

**Data-Augmentation:** Augment the data by sliding time windows from $w(t)$ to $w(t + \pi)$.

<table>
<thead>
<tr>
<th></th>
<th>previous-2-hour</th>
<th>to-be-predicted</th>
</tr>
</thead>
<tbody>
<tr>
<td>original</td>
<td>[6:00,6:20], [6:20,6:40],..., [7:40,8:00]</td>
<td>[8:00,8:20], [8:20,8:40],..., [9:40,10:00]</td>
</tr>
<tr>
<td>$\pi = 5$</td>
<td>[6:05,6:25], [6:25,6:45],..., [7:45,8:05]</td>
<td>[8:05,8:25], [8:25,8:45],..., [9:45,10:05]</td>
</tr>
<tr>
<td>$\pi = -5$</td>
<td>[5:55,6:15], [6:15,6:35],..., [7:35,7:55]</td>
<td>[7:55,8:15], [8:15,8:35],..., [9:35,9:55]</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

**Regression-Objective:** We consider two ways of approximating the evaluation objective.

- **Logarithm-Transform:**
  \[
  |\log \hat{y} - \log y| = |\log \frac{\hat{y}}{y}| = |\log(1 + \frac{\hat{y} - y}{y})| \approx |\frac{\hat{y} - y}{y}| \text{ (APE)}.\]

- **Quantile-Regression:** Minimize $|y - \hat{y}|$, a quantile point a little smaller than 0.5 (prediction a little smaller than median).
Solution Framework

- **Original data**: 573140 vehicle records of 5 tollgate-direction pairs from Sep 19 to Oct 24.
- **Data augmentation**: Sliding time-window with different timestamps.
- **Feature engineering**: Vehicle records aggregation, weather data preprocessing; reformulate the data for ML training.
- **Model training**: Train different models with absolute-loss or log-transform.
- **Model ensemble**: Weighted average of multiple model results.
## Experiment: Data Augmentation

### Table

<table>
<thead>
<tr>
<th>( \pi (\text{Lead}) )</th>
<th>mean</th>
<th>sd</th>
<th>aveCor</th>
<th>( \pi (\text{Lag}) )</th>
<th>mean</th>
<th>sd</th>
<th>aveCor</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>3</td>
<td>0.157</td>
<td>5.53</td>
<td>0.954</td>
<td>-3</td>
<td>-0.199</td>
<td>5.61</td>
<td>0.954</td>
</tr>
<tr>
<td>5</td>
<td>0.224</td>
<td>7.39</td>
<td>0.915</td>
<td>-5</td>
<td>-0.327</td>
<td>7.41</td>
<td>0.919</td>
</tr>
<tr>
<td>10</td>
<td>0.358</td>
<td>11.18</td>
<td>0.812</td>
<td>-10</td>
<td>-0.693</td>
<td>11.51</td>
<td>0.838</td>
</tr>
<tr>
<td>15</td>
<td>0.562</td>
<td>14.67</td>
<td>0.717</td>
<td>-15</td>
<td>-1.079</td>
<td>14.45</td>
<td>0.737</td>
</tr>
<tr>
<td>20</td>
<td>0.714</td>
<td>17.89</td>
<td>0.621</td>
<td>-20</td>
<td>-1.47</td>
<td>17.42</td>
<td>0.638</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
**Experiment**: Take last 7 days (Oct 18 to Oct 24) as test data (leaderboard), the remains as training data.

<table>
<thead>
<tr>
<th>Model</th>
<th>Data</th>
<th>Approach</th>
<th>MAPE</th>
</tr>
</thead>
<tbody>
<tr>
<td>LR</td>
<td>original</td>
<td>Log-Trans</td>
<td>0.146</td>
</tr>
<tr>
<td>KNN</td>
<td>original</td>
<td>Abs-Dist</td>
<td>0.140</td>
</tr>
<tr>
<td>GBDT</td>
<td>original</td>
<td>Gaussian</td>
<td>0.147</td>
</tr>
<tr>
<td>LightGBM</td>
<td>original</td>
<td>Log-Trans</td>
<td>0.133</td>
</tr>
<tr>
<td>LightGBM</td>
<td>original, ±1, ±5, ±10</td>
<td>Log-Trans</td>
<td>0.1222</td>
</tr>
<tr>
<td>NN</td>
<td>original, ±3, ±5</td>
<td>Quantile</td>
<td>0.1219</td>
</tr>
<tr>
<td>Ensemble</td>
<td></td>
<td>Weighted Average</td>
<td>0.1150</td>
</tr>
</tbody>
</table>
Experiment

Performance of Neural Network:
- Laplace/Quantile regression for absolute loss $|\hat{y} - y|$.
- Two NN Model results, eg, "with-dropout" $A$ and "without-dropout" $B$.
- $\alpha \cdot \min(A, B) + (1 - \alpha) \cdot \max(A, B)$ ($\alpha \in [0.5, 0.7]$).
- Apply cross-validation to find the best $\alpha$ ($\alpha \in \text{seq}(0.5, 0.7, by = 0.05)$).
- Train with "adadelta" for adaptive learning rate.

<table>
<thead>
<tr>
<th>Model</th>
<th>Data</th>
<th>Approach</th>
<th>MAPE</th>
</tr>
</thead>
<tbody>
<tr>
<td>3-Layers</td>
<td>original</td>
<td>gaussian</td>
<td>0.144</td>
</tr>
<tr>
<td>3-Layers</td>
<td>original</td>
<td>new</td>
<td>0.138</td>
</tr>
<tr>
<td>3-Layers</td>
<td>original, ±3, ±5</td>
<td>...</td>
<td>0.133</td>
</tr>
<tr>
<td>2-Layers</td>
<td>original, ±3, ±5</td>
<td>...</td>
<td>0.1277</td>
</tr>
<tr>
<td>1-Layer</td>
<td>original, ±3, ±5</td>
<td>...</td>
<td>0.1219</td>
</tr>
</tbody>
</table>
Decision Making: which models can we trust?

Normal model results.

Huge differences among model results.
**ETC introduction guide**

**The ETC in-vehicle device**

ETC systems can be used simply by inserting the ETC card into the in-vehicle device. The ETC in-vehicle device is equipped with a function which wirelessly communicates with the antenna set up at the toll booths to send and receive vehicle information necessary for paying the appropriate toll fare.
ETC introduction guide

The ETC in-vehicle device

ETC systems can be used simply by inserting the ETC card into the in-vehicle device. The ETC in-vehicle device is equipped with a function which wirelessly communicates with the antenna set up at the toll booths to send and receive vehicle information necessary for paying the appropriate toll fare.
Traffic control: vehicles without ETC devices from tollgate direction pair 2-0 enter.

Traffic control: Only allow vehicles with ETC devices to pass: is_etc.
Business Assumption

- Total traffic volumes entering tollgate 1,2 are stable (1-0, 2-0).
- When tollgate 2 only allow is etc vehicles to enter, those without ETC devices will turn to tollgate 1 (1-0).

Model Selection

- Build new baseline models with data:
  - 2-0: is etc volume data.
  - 1-0: total volume + no etc volume of 2-0.
- Bagging of selected model result(s) close to the baseline model results.
About JD.COM

- China’s largest retailer, online or offline—236.5 million shoppers.
- Renowned for our zero-fakes policy and amazingly fast delivery.

Y Business Units: Focus on retailing service and smart supply chain, working out demand forecasting, inventory optimization, dynamic pricing with artificial intelligence and operation research technologies.
Thank you!
Appendix: Quantile regression approach

- We are trying to minimize the APE (absolute percentage error) $|\hat{y} - y|/y$ for every training sample.
- Laplace/quantile regression (with quantile_alpha=0.5) minimize the absolute error $|\hat{y} - y|$.
- Assume the true value $y$ randomly distributed around the predicted value $\hat{y}$.
- We want to do optimal decision $\hat{y}_2$ that minimize the expected APE.
- A decision variable of $\hat{y}_2$ which is a little smaller than $\hat{y}$ achieve better (smaller) absolute percentage error.

Example: prediction value $\hat{y} = 100$, true value follows discrete uniform distribution in the interval $[60, 140]$. 

---

![Simulation of Expected APE](image_url)
Appendix: Neural network training details

- Apply randomized grid search with parameters list below.
- Bagging of blend of top-K results.

<table>
<thead>
<tr>
<th>parameters</th>
<th>list</th>
</tr>
</thead>
<tbody>
<tr>
<td>activation</td>
<td>&quot;Rectifier&quot;,&quot;Tanh&quot;</td>
</tr>
<tr>
<td>l1,l2 regularization</td>
<td>c(0, 1e-3, 1e-4, 1e-5, 1e-6, 1e-7)</td>
</tr>
<tr>
<td>input_dropout_opt</td>
<td>c(0,0.05,0.1,0.2)</td>
</tr>
<tr>
<td>max_w2_opt</td>
<td>10,20,30,40,50</td>
</tr>
<tr>
<td>adadelta-epsilon</td>
<td>c(1e-4,1e-6,1e-8,1e-10)</td>
</tr>
<tr>
<td>adadelta-rho</td>
<td>c(0.9,0.95,0.99,0.999)</td>
</tr>
</tbody>
</table>

grid search parameters list