Generative machine learning for discrete-continuous choice data

Bilal Farooq

Laboratory of Innovations in Transportation,
Ryerson University, Toronto, Canada

July 21, 2020

Wong, M., & Farooq, B. (2020). A bi-partite generative model framework for analyzing and simulating large scale multiple discrete-continuous travel behaviour data. Transportation Research Part C: Emerging Technologies, 110, 247-268.

https://arxiv.org/abs/1901.06415

Table of Contents

Table of Contents

Introduction

Opportunities

- New large scale ubiquitous multidimensional travel data sources (a.k.a. Big Data)
  - Increased size and complexity
  - Representative of the population behaviour
  - Contain rich latent information

Introduction

Challenges

- High fidelity data (images, videos, GPS etc.) contain useful information that may not be easily modelled the traditional way
- Necessitates exploring new "data-driven" modelling techniques
  - Flexible in representing the underlying heterogeneities in rich datasets
  - Improved estimation methods
  - Useful inference and interpretation

Introduction

Generative Approach

- Constructing the model of underlying distribution of the data
  - Using semi-supervised learning
  - Generate new data
    With similar stochastic variations as the population

Generative Approach

Basic notion

- Interested in describing the generation of the data by some unknown stochastic process
- Describe in probabilistic terms, how a set of latent/hidden variables could have generated the data by representing the underlying distribution

Table of Contents

Generative Bi-Partite Framework

Observed dataset

- $\mathbf{x} = x_{1:K} \in \mathbb{R}^{\mathcal{D}}$
  - $\mathbf{x}_D = \big(\underbrace{x_1, \ldots, x_{D_{\text{cont}}}}_{\text{continuous}}, \underbrace{x_{D_{\text{cont}+1}}, \ldots, x_{D_{\text{cont}}+D_{\text{cat}}}}_{\text{discrete}}\big)$

Generative Bi-Partite Framework

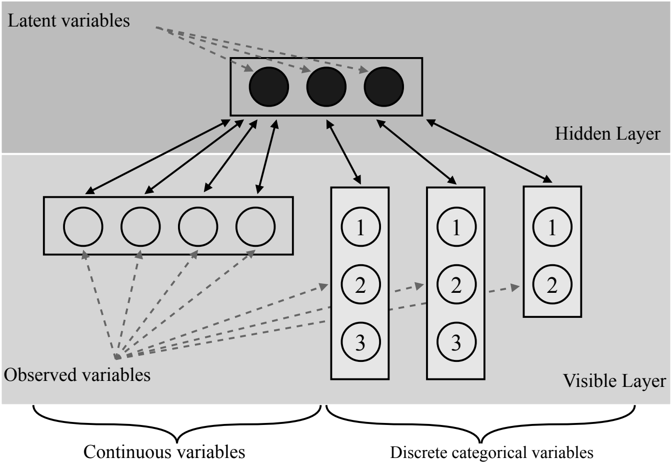
Observed dataset

- $\mathbf{x} = x_{1:K} \in \mathbb{R}^{\mathcal{D}}$
  - $\mathbf{x}_D = (\underbrace{x_1, \ldots, x_{D_{\text{cont}}}}_{\text{continuous}}, \underbrace{x_{D_{\text{cont}+1}}, \ldots, x_{D_{\text{cont}}+D_{\text{cat}}}}_{\text{discrete}})$

Latent/hidden variables

- $\mathbf{s} = s_{1:J} \in \{0, 1\}$
- Set of binary hidden random variables
- Independent and identically distributed (i.i.d.)

Generative Bi-Partite Framework



Model Structure

Joint distribution

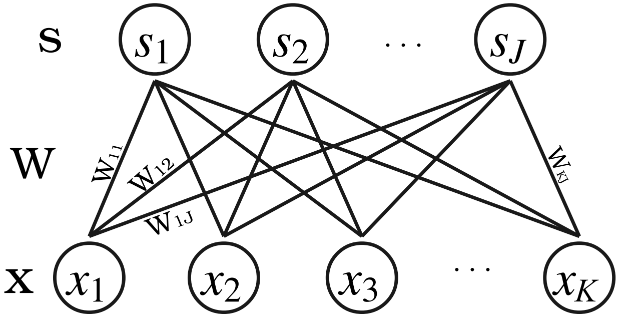
- $p(\mathbf{x}, \mathbf{s})$ over the set of observed $\mathbf{x} = x_{1:K} \in \mathbb{R}^{\mathcal{D}}$ and *binary* hidden random $\mathbf{s} = s_{1:J} \in \{0, 1\}$
- (Restricted) Boltzmann probability distribution

$$p(\mathbf{x}, \mathbf{s}) = \frac{e^{-E(\mathbf{x},\mathbf{s})}}{\sum_{x,s} e^{-E(\mathbf{x},\mathbf{s})}} \tag{1}$$

Ryerson University

LITrans

Model Structure



Model Structure

Boltzmann Energy Function

- $p(\mathbf{x}, \mathbf{s})$ as RBM with:

$$E(\mathbf{x}, \mathbf{s}) = -\mathbf{x}^\top \mathbf{W}\mathbf{s} - \mathbf{b}^\top \mathbf{x} - \mathbf{c}^\top \mathbf{s} \tag{2}$$

- $\mathbf{W} \in \mathbb{R}^{K \times J}$ is the weight matrix, connecting $\mathbf{s} = (s_1, s_2, ..., s_J)$ and $\mathbf{x} = (x_1, x_2, .., x_K)$
- $\mathbf{b}$ and $\mathbf{c}$ are the parameters for the visible and hidden layer

Model Structure

Observed variables (discrete)

- For $x_{D_{\text{cat}}} = (x_{D_{\text{cat}_1}}, ..., x_{D_{\text{cat}_k}})$, with $x_{D_{\text{cat}_k}} = 1$ i.e. k alternative for variable $x_{D_{\text{cat}}}$ is chosen:

$$p(x_{D_{\text{cat}_k}} = 1) = \frac{e^{f_k(\mathbf{s};\theta)}}{\sum_{k'} e^{f_{k'}(\mathbf{s};\theta)}}$$

Ryerson University

LITrans

Model Structure

Observed variables (continuous)

- $x_{D_{\mathrm{cont}}}$ is drawn from a Gaussian $\mathcal{N}(W, \Sigma^2)$
- To accommodate positive values only, stepped sigmoidal is used:

$$\sum_{i=1}^{\infty} \sigma(\mathbf{s} - i) \approx \ln(1 + e^{s})$$

Model Structure

Latent/Hidden variables

- With prior $p(\mathbf{s})$, we can quantify how $\mathbf{x}$ is related to $\mathbf{s}$ via likelihood function $p(\mathbf{x}|\mathbf{s})$
- Posterier distribution:

$$p(\mathbf{s}|\mathbf{x}) = \frac{p(\mathbf{x}, \mathbf{s})}{p(\mathbf{x})} \propto p(\mathbf{x}|\mathbf{s})p(\mathbf{s})$$

Ryerson University

LITrans

Model Estimation

Estimation problem

- Obtaining the posterior belief $p(\mathbf{s}|\mathbf{x})$
  - $\arg\max_{\theta} p(\mathbf{x})$ (Max Likelihood of data)
- $p(\mathbf{x}) = \int_{\mathbf{s}} p(\mathbf{x}|\mathbf{s})p(\mathbf{s})d\mathbf{s}$

Model Estimation

Estimation algorithm

- MCMC algorithms could be a solution
- High computational cost
- Posterior approximation may be difficult with large datasets and complex distributions

Model Estimation

Variational Bayesian Inference

- There exists a tractable distribution $q(\mathbf{s})$ that approximates the exact posterior $p(\mathbf{s}|\mathbf{x})$
- We search over the set of distributions that minimizes the Kullback-Leibler (KL) divergence objective function:

$$\begin{aligned} \arg\min \quad & D_{KL}[q(\mathbf{s})||p(\mathbf{s}|\mathbf{x})] \\ s.t. \quad & \frac{p(\mathbf{s}|\mathbf{x})}{q(\mathbf{s})} > 0, \\ & D_{KL}[q(\mathbf{s})||p(\mathbf{s}|\mathbf{x})] = 0 \iff q(\mathbf{s}) = p(\mathbf{s}|\mathbf{x}) \end{aligned} \tag{3}$$

Ryerson University

LITrans

Model Estimation

Variational Bayesian Inference

- In our case:

$$(D_{KL}[q(\mathbf{s})||p(\mathbf{s}|\mathbf{x})] = -\int_{\mathbf{s}} q(\mathbf{s}) \ln \frac{p(\mathbf{s}|\mathbf{x})}{q(\mathbf{s})} d\mathbf{s})$$

- Where:

$$q(\mathbf{s}) = \prod_{j=1}^{J} q(s_j) \approx \prod_{j=1}^{J} p(s_j|\mathbf{x}), \quad \mathbf{s} = \{s_1, s_2, ..., s_J\}$$

- Product of Expert Model (PoE), where each expert has tractable closed form solution $q(s_j) = (1 + e^{-Wx-c})^{-1}$.

Ryerson University

LITrans

Model Estimation

Variational Bayesian Inference

- From Eq 3, using change-of-measure technique, $D_{KL}[q(\mathbf{s})||p(\mathbf{s}|\mathbf{x})]$:

$$= \int q(\mathbf{s}) \ln q(\mathbf{s}) d\mathbf{s} - \int q(\mathbf{s}) \ln p(\mathbf{x}, \mathbf{s}) d\mathbf{s} + \ln p(\mathbf{x}) \int q(\mathbf{s}) d\mathbf{s}$$
$$= -\mathcal{F} + \ln p(\mathbf{x})$$

Model Estimation

Variational Bayesian Inference

- $\mathcal{F}$ is the variational free energy and:
$$\arg\min D_{KL}[q(\mathbf{s})||p(\mathbf{s}|\mathbf{x})] = \arg\max F$$
- Variational free energy objective is the lower bound approximation to log-likelihood of data as $\ln p(\mathbf{x}) \geq \mathcal{F}$

Learning Algorithm

Learning $q(s)$ using $\mathcal{F}$

$$
\begin{aligned}
\nabla_{q(\mathbf{s};\theta)} F &= \nabla_{q(\mathbf{s};\theta)} \ln \sum_{s} p(\mathbf{x}, \mathbf{s}; \theta) && (4) \\
&= \nabla_{q(\mathbf{s};\theta)} \ln \frac{\sum_{s} e^{-E(\mathbf{x},\mathbf{s};\theta)}}{\sum_{x,s} e^{-E(\mathbf{x},\mathbf{s}\theta)}} && (5) \\
&= \nabla_{q(\mathbf{s};\theta)} \Bigg(\underbrace{\ln \sum_{s} e^{-E(\mathbf{x},\mathbf{s};\theta)}}_{\text{utility } U} - \underbrace{\ln \sum_{x,s} e^{-E(\mathbf{x},\mathbf{s};\theta)}}_{\text{entropy } \mathcal{H}} \Bigg) && (6)
\end{aligned}
$$

Ryerson University

LITrans

Learning Algorithm

Learning $q(s)$ using $\mathcal{F}$

$$q(\mathbf{s}) := \max_{q(\mathbf{s})} F \iff$$

$$\nabla_{q(\mathbf{s};\theta)} F = 0, \text{ for any } \theta^* \in \arg\max_{x \in \mathbb{D}} \ln p(\mathbf{x}; \theta^*)$$

Ryerson University

LITrans

Learning Algorithm

Learning $q(s)$ using $\mathcal{F}$

Using stochastic gradient descent

$$\theta_t \leftarrow \theta_{t-1} - \frac{1}{A_\tau}\eta \sum_{A_\tau} \nabla_{q(\mathbf{s};\theta)} - \mathcal{F}_{A_\tau} \qquad \forall A_\tau \in \mathcal{D}, \tau = 1, \ldots T$$

Learning Algorithm

Input : RBM data sample $\mathcal{D} = \{\mathbf{x}_1, ..., \mathbf{x}_n\}$, batch sample $A_i \subset \mathcal{D}, i = 1, ..., d$, learning rate η, iteration steps T
Output: gradient approximation $\theta = (\mathbf{W}, \mathbf{c}, \mathbf{b})$.

init: $\theta = 0$, $\tau = 1$;
forall $A_\tau \in \mathcal{D}, \tau = 1, ..., T$ **do**
  forall $(\mathbf{x}_n) \in A_\tau$ **do**
    for $t = 1$ **to** N **do**
      CD_t: iterate over Gibbs chain
      positive phase
      $\mathbf{x}^0 \leftarrow \mathbf{x}_n$
      $\mathbf{s}^0 \sim \prod_{j=1}^{H} p(s_j|\mathbf{x}^0)$
      negative phase
      $\mathbf{x}^t \sim \prod_{i=1}^{I} p(x_i|\mathbf{s}^0)$
      $\mathbf{s}^t \sim \prod_{j=1}^{H} p(s_j|\mathbf{x}^t)$
    end
  end
  % Variational free energy term
  $\nabla_{q(\mathbf{s};\theta)}(-\mathcal{F})_{A_\tau} \approx (\langle \mathbf{x}^t \mathbf{s}^t \rangle - \langle \mathbf{x}^0 \mathbf{s}^0 \rangle)$
  % parameter update step
  for $\theta \in \theta$ **do**
    $\theta_{\tau+1} \leftarrow \theta_\tau - \eta \nabla_{q(\mathbf{s};\theta)}(-\mathcal{F})_{A_\tau}$;
  end
end

Inference Using RBM

Simple Example

- Two observed variables $[x, y]$ connected by a single hidden unit s_j
- Boltzmann Energy:
  $E(x, y, s) = -\sum_{s_j} xW_{1,j}s_j - \sum_{s_j} yW_{1,j}s_j - b_1 x - \sum_{s_j} c_j s_j - b_2 y$

Ryerson University

LITrans

Inference Using RBM

Simple Example

- Then for $P(y|x) = \dfrac{e^{-F(x,y)}}{\sum_{y'} e^{-F(x,y')}}$

$$
\begin{aligned}
F(x,y) &= -\ln \sum_{s_j \in \{0,1\}} e^{-E(x,y,s_j)} \\
&= -b_1 x - d_2 y - \ln(1 + e^{-xW_{1,j} - yW_{1,j} - c_j})
\end{aligned}
$$

Inference Using RBM

Simple Example

- Suppose $y = \{y^1, y^2, y^3\}$

$$
\begin{aligned}
F(x_1, y_1^1) &= -\Big(b_1 x_1 + d_2^1 \cdot (y_1^1 = 1) + d_2^2 \cdot (y_1^2 = 0) \\
&+ d_2^3 \cdot (y_1^3 = 0) + \ln(1 + e^{-x_1 W_{1,j} - y_1 W_{1,j} - c_j})\Big) \\
&= -\Big(b_1 x_1 + d_2^1 + \underbrace{\ln(1 + e^{-x_1 W_{1,j} - y_1 W_{1,j} - c_j})}_{\text{single correction term}}\Big)
\end{aligned}
$$

Inference Using RBM

Simple Example

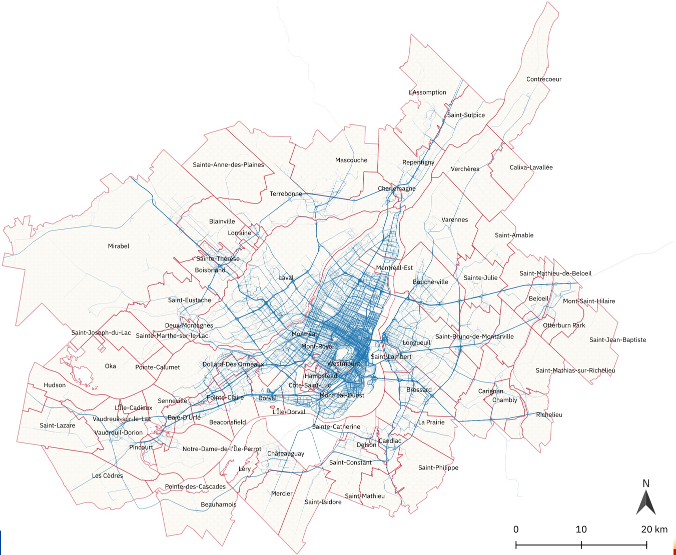
- Suppose that weights to hidden connections are zero, $W_1 = W_2 = c_j = 0$, then $F(x_1, y_1^1) = -\Big(b_1 x_1 + d_2^1 + \ln(1 + e^0)\Big) = -\underbrace{(b_1 x_1 + d_2^1)}_{\text{MNL utility}}$

Table of Contents

Montreal GPS Dataset

- 2016 MTL Trajet GPS data from the Greater Montréal Region
- Open dataset with 293,330 trip observations
- Variables considered:
  - Mode choice
  - Trip purpose
  - Trip distance
  - Origin/destination point
  - Departure/arrival time

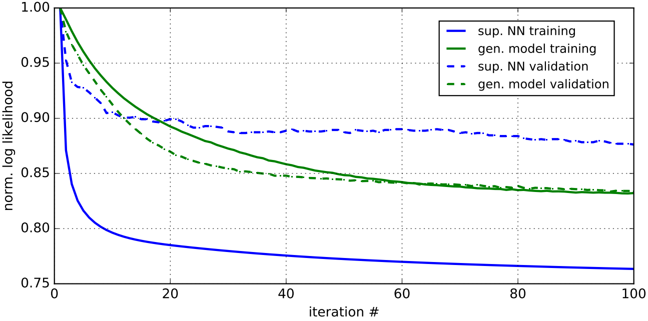
Montreal GPS Dataset



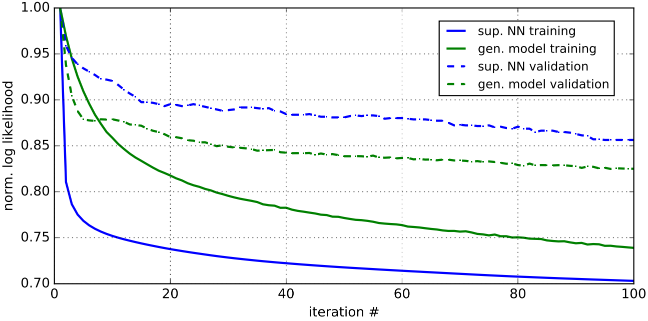
Ryerson University

LITrans

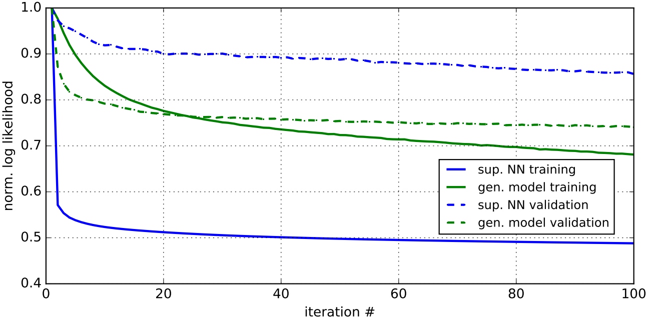
Benchmarking with Supervised NN



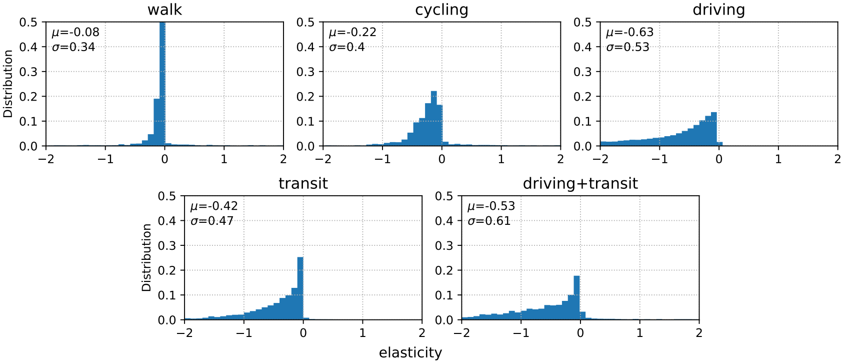
Benchmarking with Supervised NN



Benchmarking with Supervised NN

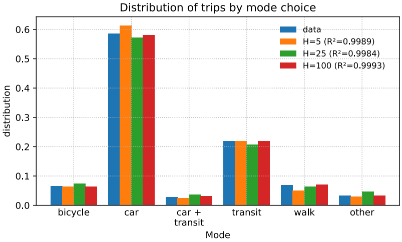


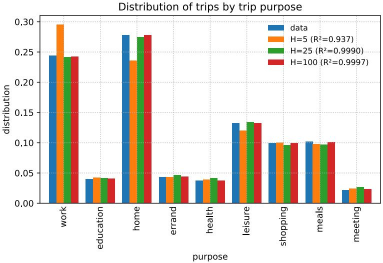
Mode Choice Elasticity to Distance

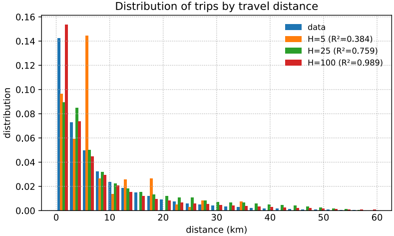


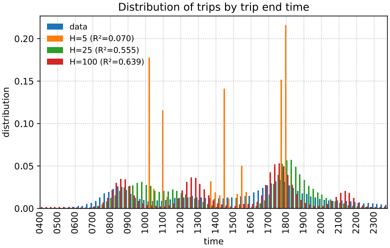
Ryerson University

Forecasting









Forecasting

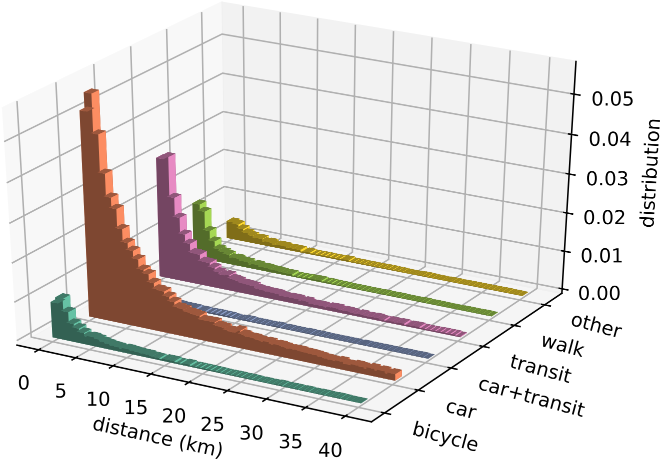


Table of Contents

Concluding Remarks

- RBM based generative model for discrete-continuous travel behaviour data
  - VBI based estimation process for retrieving the joint distribution
  - Generation of conditional probabilities and economic analysis
- Performed better in forecasting, when compared to supervised feed-forward neural networks
  - With the similar dimensionality/same number of latent variables used

Concluding Remarks

- Increase in latent variables, may cause overfitting
  - Regularization techniques can be used
- Application on other high fidelity datasets
- Explore the use in population synthesis
- Explore the use of other generative models
  - Variational Autoencorders (VAE)
  - Generative Adversarial Networks (GANs)

Thanks for listening!

Paper: `https://arxiv.org/abs/1901.06415`
Source Code: `https://github.com/LiTrans/ML-MDC`