CS 590/04-10-25

An Introduction to Flow Matching

Qilin Ye[1]

[1] *Department of Computer Science, Duke University*

I. BACKGROUND

A. The Main Problem & Motivations

- A critical question for decision-makers: **what medicine/treatment to prescribe to patients**?
- Challenges:
  - Only *observational* data are accessible
  - Treatment decisions evolve over time and are highly dependent on patients
- Need to:
  - *Estimate* treatment effects from observational data
  - Techniques to adjust for **time-dependent confounders**

I. BACKGROUND

B. What is a Time-Dependent Confounder?

- Patient covariates affected by *previous* treatments may influence *future* treatment choices and outcomes.
- Hard to...
  - Isolate the confounders due to their dynamic nature
  - Accurately measure the true effect of any treatment

I. BACKGROUND

B. Time-Dependent Confounder: Example

- Suppose Treatment A is administered when a patient's WBC count is abnormal for days.
- ... but WBC count may be influenced by a prior administration of Treatment B.
- Observation: treatment A leads to higher probability of death.
- Q: Should we declare Treatment A harmful?

... the analysis must account for *both* the cumulative *and* the interdependent effects of treatment sequences.

I. BACKGROUND & PRELIMINARIES

C. Dataset & Benchmarking

All papers in this presentation use a well-known Pharmacokinetic-Pharmacodynamic model of tumor growth [cite]. It simulates the combined effects of chemotherapy and radiotherapy in lung cancer patients:

$$V(t+1) = \left(1 + \underbrace{\rho \log\left(\frac{K}{V(t)}\right)}_{\text{tumor growth}} - \underbrace{\beta_c C(t)}_{\text{chemo}} - \underbrace{(\alpha_r d(t) + \beta_r d(t)^2)}_{\text{radio}} + \underbrace{\epsilon_t}_{\text{noise}}\right) V(t). \tag{1}$$

- Time-varying confounding: set chemo/radiotherapy assignment as Bernoulli r.v.'s.
- intensity of confounding: governed by two additional parameters for chemo/radiotherapy.

I. BACKGROUND & PRELIMINARIES

C. Dataset & Benchmarking

- Synthetic patient trajectories with pre-specified parameters.
- Contains both factual and counterfactual outcomes.
- Metrics:
  - Predictive accuracy: *How well does the model predict future tumor volumes?*
  - Timing decision accuracy: *Does the model select optimal treatment with appropriate timing?*

I. BACKGROUND & PRELIMINARIES

D. List of Models to be Presented

- Recurrent Marginal Structural Network (RMSN)
- Counterfactual Recurrent Network (CRN)
- Causal Transformer (CT)
- G-Transformer

I. BACKGROUND

E. Some Notations... Subject to Minor Changes Later

Each patient X will be associated with the following:

- Time-dependent covariates L_t at time t (e.g. health condition at time t)
- (With abuse of notation) static covariates $X = \{X_i\}$ (e.g. gender, genetic information)
- Treatment a_t applied at time t
- Time-dependent outcomes Y_t at time t.

Let $H_t = \langle (L_1, \ldots, L_t), (a_1, \ldots, a_{t-1}), X \rangle$ patient's medical history. Given H_t and $(a_t, \ldots, a_{t+\tau-1})$ we want to define $g(t, \tau) = g(H_t, (a_t, \ldots, a_{t+\tau-1}))$ that approximates the following ground truth:

$$\mathbb{E}[Y_{t+\tau} \mid H_t, (a_t, \ldots, a_{t+\tau-1})]. \tag{2}$$

II. RECURRENT MARGINAL STRUCTURAL NETWORK (RMSN)

A. Inverse Probability of Treatment Weighting (IPTW)

- Intuition: treatment is based on individual's clinical conditions, so a non-uniform weighting scheme is needed to analyze treatment effect.
- IPTW provides a way to "normalize" a clinically biased population.
- Correction for selection bias:
  - *Example: sicker patients are more likely to receive treatment A.*
  - *Without IPTW, hard to isolate the treatment effect of A. (Sicker? Or effective?)*
- More math next slide...

II. RECURRENT MARGINAL STRUCTURAL NETWORK (RMSN)

A. IPTW and Math...

- The building block of IPTW is $w_i = \dfrac{\mathbb{P}(\text{treatment})}{\mathbb{P}(\text{treatment} \mid \text{trait})}$. High fraction means this population is significant to the outcome, so more upweight.

- Stabilized weight in its full form (where f is the treatment distribution):

$$\mathbf{SW}(t, \tau) = \prod_{n=t}^{t+\tau} \frac{f(A_n \mid A_{n-1})}{f(A_n \mid H_n)} = \prod_{n=t}^{t+\tau} \frac{\prod_{\text{treatments k}} f(A_n(k) \mid A_{n-1})}{\prod_{\text{treatments } k} f(A_n(k) \mid H_n)} \tag{3}$$

  (observe H_n contains both A_{n-1} and personal traits).

- With censoring (requires complete trajectories):

$$\mathbf{SW}^*(t, \tau) = \prod_{n=t}^{t+\tau} \frac{f(A_n \mid A_{n-1}, \text{no censoring by time } n)}{f(A_n \mid H_n, \text{no censoring by time } n)}. \tag{4}$$

II. RECURRENT MARGINAL STRUCTURAL NETWORK (RMSN)

A. IPTW and Math...

Final loss component (with further normalized $\mathbf{SW}(t,\tau)$):

$$e(i,t,\tau) = \underbrace{\mathbf{SW}(t,\tau-1)}_{\text{normalizes bias}} \cdot \underbrace{\mathbf{SW}^*(t,\tau-1)}_{\text{wants full trajectory}} \cdot \| \underbrace{Y_{t+\tau,i}}_{\text{ans}} - \underbrace{g(t,\tau)}_{\text{pred}} \|^2. \tag{5}$$

- Encourages simulation of a randomized experiment:
  - Treatment weight upweights less common treatment decisions
  - Censoring weight upweights observations with early termination
- TL;DR: $e(i,t,\tau)$ discourages model from "ignoring" cases that are critical for unbiased causal inference.

II. RECURRENT MARGINAL STRUCTURAL NETWORK (RMSN)

B. Training RMSN: Architecture

- **Propensity networks** used to estimate conditional probabilities in (3) and (4).
- Prediction netowrk - **encoder**: standard LSTM.
  - Input: patient history $H_t = \langle (L_1, \ldots, L_t), (a_1, \ldots, a_{t-1}), X \rangle$ (patient traits and past treatments).
  - Output: a hidden state h_t, and a one-step-ahead prediction $\hat{Y}_{t+1}$.
- Prediction network - **decoder**: another standard LSTM.
  - Input: an initial decoder state z_t transformed from h_t; and a future treatment sequence $(a_t, \ldots, a_{t+\tau-1})$.
  - Output: a predicted desponse for each future horizon $g(t, \tau)$ for $1 \leqslant \tau \leqslant \tau_{\max}$.

II. RECURRENT MARGINAL STRUCTURAL NETWORK (RMSN)

B. Training RMSN: Loss Objective

- Propensity network is trained using standard CE loss.
- Encoder: (weighted) MSE for one-step predictions, defined as

$$\mathcal{L}_{\text{enc}} = \sum_{i,t} e(i,t,1) = \sum_{i,t} [\mathbf{SW}(t,0) \cdot \mathbf{SW}^*(t,0) \cdot \|Y_{t+1,i} - g(t,1)\|^2].$$

- Decoder: multi-step weighted MSE, defined as

$$\mathcal{L}_{\text{dec}} = \sum_{i=1}^{I} \sum_{t=1}^{T_i} \sum_{\tau=2}^{\min(T_i - t, \tau_{\max})} e(i,t,\tau).$$

III. COUNTERFACTUAL RECURRENT NETWORK (CRN)

A. Intuition: Balancing Representations

- Want: given history, predict outcome: $H_t \to Y_{t+1}$.

- Do not want: spurious path $H_t \to a_t \to Y_{t+1}$.

- Goal: a representation of H_t that is *not predictive* of treatment a_t.

- Formally: a mapping Φ where $\mathbb{P}(\Phi(H_t) \mid A_t = a_i)$ remains constant over all treatments a_i.

III. COUNTERFACTUAL RECURRENT NETWORK (CRN)

B. Training CRN: Architecture

- **Encoder**: RNN with LSTM unit.
  - Input: history H_t
  - Output: a representation $\Phi(H_t)$, and a one-step-ahead prediction $\hat{Y}_{t+1}$.
- **Decoder**: RNN with LSTM unit.
  - Input: latent representation $\Phi(H_t)$; future treatments $(a_t, \ldots, a_{t+\tau-1})$; static features X.
  - Teacher forcing during training.
  - Output: counterfactual outcomes $\hat{Y}_{t+1}, \ldots, \hat{Y}_{t+\tau}$.

III. COUNTERFACTUAL RECURRENT NETWORK (CRN)

B. Training CRN: Loss Objective

- **Outcome predictor** G_y (outcome prediction):

$$\mathcal{L}_{y;t,i} = \mathcal{L}_y = \|Y_{t+1} - G_y(\Phi(H_t))\|^2. \tag{6}$$

- **Treatment classifier** G_a (domain discrimination):

$$\mathcal{L}_{a;t,i} = \mathcal{L}_a = -\sum_{\text{treatment } j} \mathbf{1}[a_t = j] \log G_a(\Phi(H_t), a_t = j). \tag{7}$$

Encoder Φ wants to please G_y but fool G_a.

$$\mathcal{L} = \mathcal{L}_{t,i} = \sum_{\text{patient } i} [\mathcal{L}_{y;t,i} - \lambda \cdot \mathcal{L}_{a;t,i}]. \tag{8}$$

III. COUNTERFACTUAL RECURRENT NETWORK (CRN)

B. Training CRN: Loss Objective

Theorem 1. ***Adversarial training encourages*** $\Phi(H_t)$ ***to be domain indiscriminant.*** *Formally:*

Fix t. *For* $j \in [k]$ *(treatments), let* P_j *denote the distribtion of* H_t *conditioned on* $a_t = j$. *Let* G_a^j *denote the output of* G_a *given* $a_t = j$. *Let* P_j^Φ *denote the distribution of* $\Phi(H_t)$ *given* $a_t = j$. *The minimax game defined by*

$$\min_{\Phi} \max_{G_a} \sum_{\textit{treatment } j} \mathbb{E}_{H_t \sim P_j}[\log G_a^j(\Phi(H_t))] \qquad \textit{subject to} \qquad \sum_{\textit{treatment } j} G_a^j(\Phi(H_t)) = 1$$

has a global minimum that uniquely correpsonds to when all P_j^Φ *agree, i.e., when the learned representations are invariant across all treatments.*

IV. CAUSAL TRANSFORMERS (CT)

A. Motivation

LSTM struggles to capture complex, long-range dependencies... which patient histories can be.

Attention!

IV. CAUSAL TRANSFORMERS (CT)

A. Motivation & Preliminaries

Building blocks of a transformer block: each token is associated with three embeddings:

- **Query** (Q): the “search” feature of a token — what information is sought at a given position.
- **Key** (K): the “contents” of a token, so its relevance can be measured by...
- **Value** (V): the actual information that is aggregated according to attention weights from QK-similarity.

$$\mathrm{Attn}(Q, K, V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) V. \tag{9}$$

IV. CAUSAL TRANSFORMERS (CT)

B. CT: Architecture

- Three paralellel **transformer subnetworks**, each dedicated to one of the following:
  - Time-varying covariates X_t
  - Past outcomes Y_t
  - Treatment history a_t.

  Output of each subnetwork: *sequence* of hidden states representing each one above.
- Standard transformer tricks: masked multi-head self-attention; cross-attention layers; relative positional encoding, etc.
- For each time step t, fusion the three outputs into a single Φ_t by averaging.

IV. CAUSAL TRANSFORMERS (CT)
B. CT: Architecture & Loss Objectives

More nuanced requirement for representations:

- Want Φ_t to predict outcome Y_{t+1}.
- Do not want Φ_t to predict a_t.
- Still want Φ_t to be predict a_t **for diagnostic purposes**.
- **Goldilocks zone: the representation should be able to forecast what happens next, but unable to predict doctor's decision.**

B. CT: Architecture & Loss Objectives

- **Outcome prediction network**, G_y:
  - Given Φ_t, a_t, predict $\hat{Y}_{t+1}$.
  - Loss (want to minimize): standard MSE $\mathcal{L}_y = \|Y_{t+1} - G_y(\Phi_t, a_t)\|^2$.
- **Treatment classifier network**, G_a:
  - Given Φ_t, predict the distribution over next treatment.
  - Loss #1: want G_a to be able to predict a_t:

$$\mathcal{L}_{G_a} = -\sum_{\text{treatment } j} \mathbf{1}[a_j = a] \log G_a(\Phi_t) \tag{10}$$

  - Loss #2 (adversarial): **C**ounterfactual **D**omain **C**onfusion: want only G_a, *not* Φ_t, to be predictive:

$$\mathcal{L}_{\text{conf}} = -\sum_{j=1}^{k} \frac{1}{k} \log G_a(\Phi_t). \tag{11}$$

IV. CAUSAL TRANSFORMERS (CT)

B. CT: Loss Objectives

Let $\theta_Y, \theta_A, \theta_R$ be the parameters for G_y, G_a, and parameters for generating Φ_t. Iteratively compute

$$(\hat{\theta}_Y, \hat{\theta}_R) = \underset{\theta_Y, \theta_R}{\operatorname{argmin}} \mathcal{L}_y(\theta_Y, \theta_R) + \lambda \mathcal{L}_{\text{conf}}(\hat{\theta}_A, \theta_R)$$

$$\hat{\theta}_A = \underset{\theta_A}{\operatorname{argmin}} \lambda \mathcal{L}_{G_a}(\theta_A, \hat{\theta}_R).$$

Intuitions:

- Bottom equation: optimizes the classifier *using a nice representation.*
- Top equation: updates the representation adversarially to balance outcome prediction and domain confusion, *using a nice treatment classifier (doctor).*

V. G-TRANSFORMER

A. Motivation

- Prior methods estimate treatment effects over time under *static* regimes (predetemrined treatments).
- Real-world treatment decisions are *dynamic* and *time-varying* (evolving over time).

G-transformers fill this gap.

B. G-Transformer: Architecture

- Transformer-based encoder-only model.
- Two transformer encoders: one for continuous covariats, one for discrete.
- **G-computation**: dynamically simulates counterfactual trajectories under specified policies.
- MC simulation for multi-step counterfactual prediction.

V. G-TRANSFORMER

B. G-Transformer: Architecture

Split time-dependent covariants L_t at time t into categorical/disjoint L_t^d and continuous L_t^c.

- **Categorical encoder**: temporal patterns, conditional class distributions.
- **Continuous encoder**: dynamics of continuous covariates (e.g. vitals).
- Goal: given info by time t, predict next-step L_{t+1}^c, L_{t+1}^d.
- Teacher-forcing training, autoregressive simulation.

V. G-TRANSFORMER

B. G-Transformer: Architecture — g-computation

TL;DR: simulate full counterfactual outcome trajectory if they were to follow a certain policy g.

Example: *"give drug A if blood pressure* < 90 *for past* 3 *hours."*

- Start with observed history H_m (simulation starts here).
- At each future time step $t \geqslant m$:
  - Simulate treatment $a_t \leftarrow g(H_t)$
  - Sampling: categorical sampling uses softmax logits; contiuous sampling uses point estimation.

V. G-TRANSFORMER

B. G-Transformer: Loss Objectives

- Categorical covariates: cross-entropy. For patient i at time j, let $p_{i,t,j}$ be the probability that the model correctly assigns the desired categorical value to variable j:

$$\mathcal{L}_{\mathrm{CE}} = -\sum_{i} \sum_{t \geqslant m} \sum_{j} \log p_{i,t,j}.$$

- Continuous covariates: MSE. Let $\hat{L}_{i,t,j}$ be the predicted value and $L_{i,t,j}$ the ground truth:

$$\mathcal{L}_{\mathrm{MSE}} = \sum_{i} \sum_{t \geqslant m} \sum_{j} (L_{i,t,j} - \hat{L}_{i,t,j})^2.$$

- Final loss is $\mathcal{L}_{\mathrm{total}} = \mathcal{L}_{\mathrm{CE}} + \mathcal{L}_{\mathrm{MSE}}$, up to some weighting factors.

V. REFERENCES

Template submitted by D. Backhouse