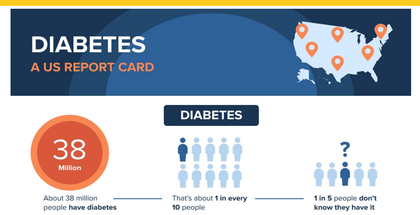
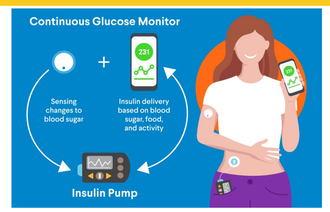
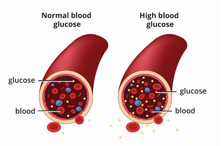
Continuous Diabetic Glucose Monitoring by Conditional Diffusion Insulin Control via Offline Reinforcement Learning

Matthew Manion, Sathvika Ayyappa Prabhu
Department of Chemical Engineering; Department of Computer Science Engineering

Introduction



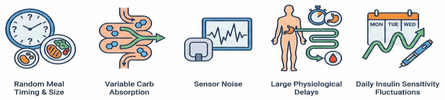




- Diabetes management requires continuous glucose monitoring with dynamic insulin release calculations
- Current controllers are limited in their ability to predict out of distribution events
- Here, we explore conditional diffusion as a potential remedy to current insulin controllers

Background & Experimental

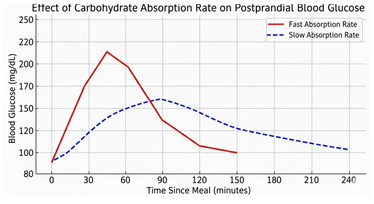
Causes of Metabolic Stochasticity



Python implementation of the FDA-approved UVA/Padova Type-1 Diabetes Simulator (2008)

Includes 30 virtual patients
10 adults, 10 adolescents, 10 children

Designed to be "reinforcement-learning-ready":
OpenAI Gym / Gymnasium API
Each step returns (observation, reward, done, info)



We use the clinical Blood Glucose Risk Index, which quantifies how dangerous a glucose value is.

Risk is split into two components:

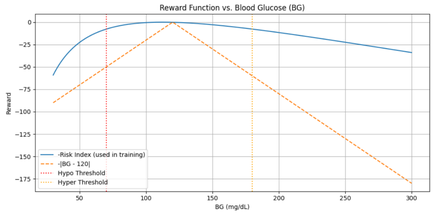
LBGI (Low BG Risk Index) severe penalty for hypoglyc-i-cemia

HBGI (High BG Risk Index) moderate penalty for hyperglycemia

Hypoglycemia carries much higher risk, so LBGI is weighted more heavily.

For each glucose value Gt, we compute a risk score:

$$Risk_t = \lambda \cdot LBGI_t + (1-\lambda) \cdot HBGI_t \quad , \lambda > 0.5$$



Reward is: $r_t = -Risk_t$

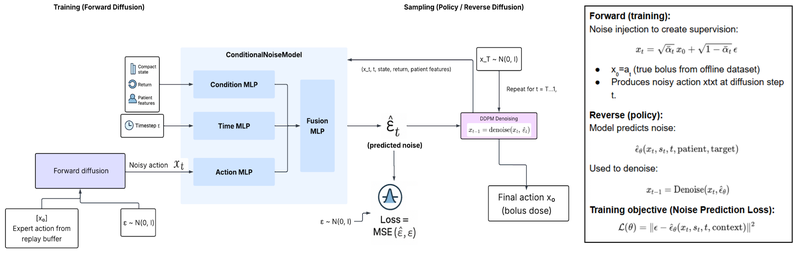
We treat an insulin bolus a_t as a sample generated from noise

Forward process: add Gaussian noise to dataset actions → produce noisy samples

Reverse process: learn to denoise step-by-step conditioned on:
- current state s
- patient physiology
- target return
- diffusion timestep t

At test time:
Start from pure noise → Iteratively denoise → final action = insulin bolus

Diffusion Model Architecture



Forward (training):
Noise injection to create supervision:

$$x_t = \sqrt{\bar{\alpha}_t}\, x_0 + \sqrt{1-\bar{\alpha}_t}\, \epsilon$$

- $x_0 = a_t$ (true bolus from offline dataset)
- Produces noisy action xtxt at diffusion step t.

Reverse (policy):
Model predicts noise:

$$\hat{\epsilon}_\theta(x_t, s_t, t, \text{patient}, \text{target})$$

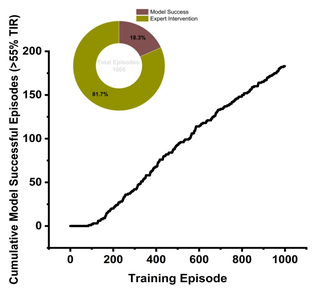
Used to denoise:

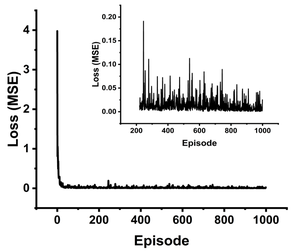
$$x_{t-1} = \text{Denoise}(x_t, \hat{\epsilon}_\theta)$$

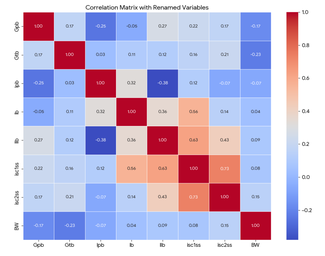
Training objective (Noise Prediction Loss):

$$\mathcal{L}(\theta) = \|\epsilon - \hat{\epsilon}_\theta(x_t, s_t, t, \text{context})\|^2$$

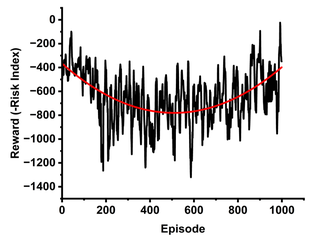
Multi-Parameter Model Training Metrics





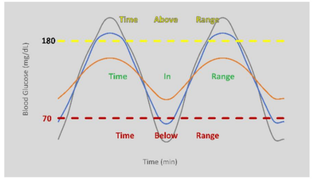


Metabolic Input Parameter Correlations



Results

Metric	Definition
TIR (%) Time in Range	% of time BG is **70–180 mg/dL**
TBR (%) Time Below Range	% of time BG is **< 70 mg/dL**(hypoglycemia)
TAR (%) Time Above Range	% of time BG is **> 180 mg/dL**(hyperglycemia)
CV (%) Coefficient of Variation	Glucose variability (**SD ÷ mean × 100**)



Diffusion Trained Only on Body Weight

Method	TIR (%)	TBR (%)	CV (%)
PID Controller	68.2	6.1	32.1
Tabular Q-Learning	70.1	5.2	30.3
TD3-BC	71.8	4.5	28.4
Diffusion Policy (ours)*	**100.0**	**0.0**	**13.35**
Clinical Controller*	91.3	0.0	8.83

Danger

*Averaged Over 3 Adult Evaluations

Diffusion Trained on Multi Params

Method	TIR (%)	TBR (%)	CV (%)
Diffusion Policy (ours)*	**58.9**	**0.0**	**16.2**
Clinical Controller*	94.2	0.0	14.8

*Averaged Over 3 Adult Evaluations

Diffusion Trained Only on Body Weight (3-day Traiectories)

Method	TIR (%)	TBR (%)	CV (%)
Tabular Q-Learning	60.2	11.2	35.9
Diffusion Policy (ours)*	**72.7**	**9.3**	**24.1**
Clinical Controller*	90.5	1.8	17.3

*Averaged Over 3 Adult Evaluations

Diffusion Trained Only on Body Weight (7-day Trajectories)

Method	TIR (%)	TBR (%)	CV (%)
Tabular Q-Learning	68.3	7.8	33.1
Diffusion Policy (ours)*	**89.1**	**5.7**	**19.8**
Clinical Controller*	92.8	0.2	11.2

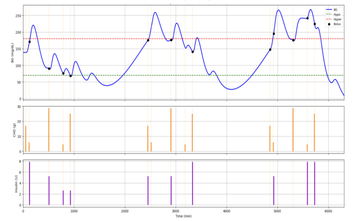
*Averaged Over 3 Adult Evaluations

Results cont.

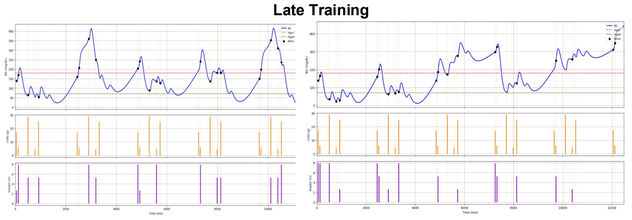
Body Weight Model Generalization to Other Groups

Method	TIR (%)	TBR (%)
Diffusion Policy (ours) (Adolescent 1)	79.5	8.3
Clinical Controller (Adolescent 1)	71.2	0.0
Diffusion Policy (ours) (Child 1)	49.0	51.0
Clinical Controller (Child 1)	76.0	12.8

Early Training (Hypoglygemic Death)



Late Training



Conclusions & Next Steps

- Diffusion policies are effective for stochastic offline RL in blood glucose control.
- Performs reliably on adult virtual patients, where physiological dynamics are more stable.
- Generalization to adolescents/children remains challenging due to higher variability and noisier glucose–insulin dynamics.
- Training diffusion on multiple patient parameters requires significant compute, especially under stochastic transitions.

Next Steps:
- Scale up compute
- Change Reward Scheme: Emphasize Ideal Glucose Region
- Expand Time Horizon

References

[1] Anurag Ajay, Yilun Du, Abhi Gupta, Joshua Tenenbaum, Tommi Jaakkola, and Pulkit Agrawal. 2023. Is Conditional Generative Modeling All You Need for Decision-Making?. In International Conference on Learning Representations(ICLR)

[2] Harry Emerson, Matthew Guy, and Ryan McConville. 2023. Offline reinforcement learning for safer blood glucose control in people with type 1 diabetes. Journal of Biomedical Informatics 142 (2023).

[3] Michael Janner, Qiyang Li, and Sergey Levine. 2021. Offline reinforcement learning as one big sequence modeling problem. In Advances in Neural Information Processing Systems, Vol. 34.