Boosting heterogeneous VAEs via multi-objective optimization

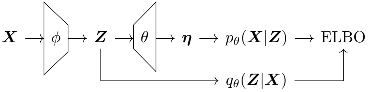
NeurIPS 2021 Workshop "Your Model is Wrong: Robustness and misspecification in probabilistic modeling"

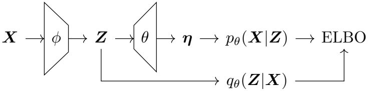
Adrián Javaloy, Maryam Meghdadi, Isabel Valera

University of Saarland
MPI Software Systems

December 3, 2021

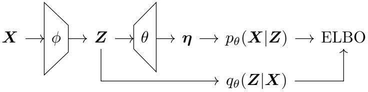
VAEs and heterogeneous data





$$p_\theta(\boldsymbol{X}|\mathbf{z}) = \prod_{d=1}^{D} p_d(\mathbf{x}_d|\mathbf{z})$$

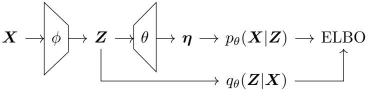
VAEs and heterogeneous data



$$p_\theta(\boldsymbol{X}|\mathbf{z}) = \prod_{d=1}^{D} p_d(\mathbf{x}_d|\mathbf{z})$$

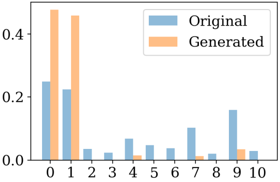
- **Implicit assumption.** We want to learn <u>all features</u> equally well.

VAEs and heterogeneous data

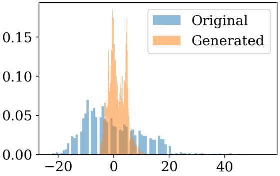


$$p_\theta(\boldsymbol{X}|\mathbf{z}) = \prod_{d=1}^{D} p_d(\mathbf{x}_d|\mathbf{z})$$

- **Implicit assumption.** We want to learn all features equally well.
- **Problem.** During training, VAEs prioritize some features over others.



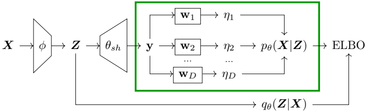
Discrete data.



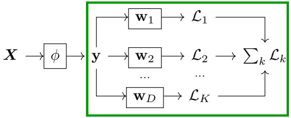
Continuous data.

Multiobjective architectures

Heterogeneous VAEs

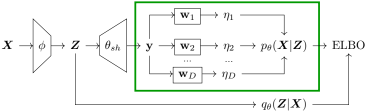


Multitask Learning

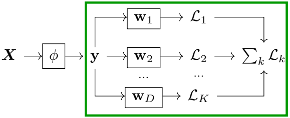


Multiobjective architectures

Heterogeneous VAEs



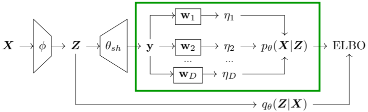
Multitask Learning



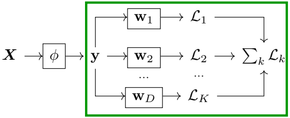
	Heterogeneous VAEs	Multitask Learning
Want:	Model all features equally well.	Learn all tasks equally well.
Problem:	Feature overlooking.	Negative transfer.

Multiobjective architectures

Heterogeneous VAEs

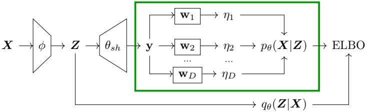
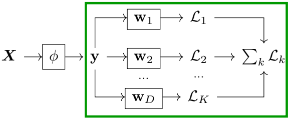


Multitask Learning



Want:	Model all features equally well.	Learn all tasks equally well.
Problem:	Feature overlooking.	Negative transfer.
Shared:	ϕ and θ_{sh}.	ϕ.
Exclusive:	$\mathbf{w_1}, \mathbf{w_2}, \ldots, \mathbf{w_K}$.	$\mathbf{w_1}, \mathbf{w_2}, \ldots, \mathbf{w_K}$.

Multiobjective architectures

	Heterogeneous VAEs	Multitask Learning
		
Want:	Model all <u>features</u> equally well.	Learn all <u>tasks</u> equally well.
Problem:	Feature overlooking.	Negative transfer.
Shared:	ϕ and θ_{sh}.	ϕ.
Exclusive:	$\mathbf{w_1}, \mathbf{w_2}, \ldots, \mathbf{w_K}$.	$\mathbf{w_1}, \mathbf{w_2}, \ldots, \mathbf{w_K}$.

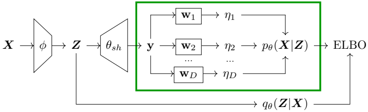
Updating ϕ:

$$\nabla_\phi p_\theta \nabla_{p_\theta} \text{ELBO} = $$
$$= \nabla_\phi \mathbf{y} \underbrace{\left(\sum_d \nabla_{\mathbf{y}} \eta_d \nabla_{\eta_d} p_\theta \right)}_{\text{feature overlooking}} \nabla_{p_\theta} \text{ELBO}$$

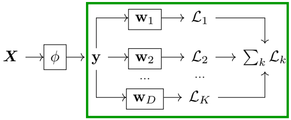
$$\nabla_\phi \mathcal{L} = \nabla_\phi \mathbf{y} \underbrace{\left(\sum_k \nabla_{\mathbf{y}} \mathcal{L}_k \right)}_{\text{negative transfer}}$$

Tackling feature overlooking

Heterogeneous VAEs



Multitask Learning

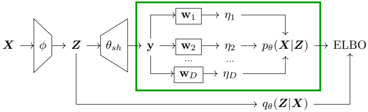


Conflicting gradients

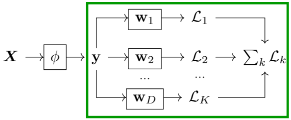
Differences between gradient among tasks/features lead to poor gradient directions, and thus shared-parameters updates.

Tackling feature overlooking

Heterogeneous VAEs



Multitask Learning

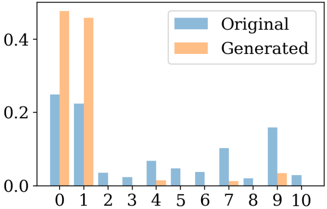


Conflicting gradients

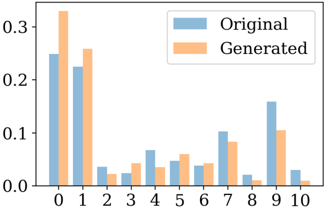
Differences between gradient among tasks/features lead to poor gradient directions, and thus shared-parameters updates.

These conflicts are restricted to the green squares. We can leverage existing MTL solutions to alleviate feature overlooking in heterogeneous VAEs.

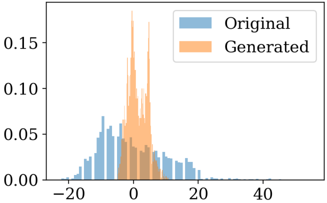
Qualitative results



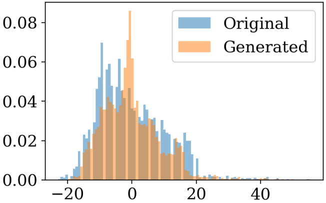
Vanilla - discrete.



MOO - discrete.



Vanilla - continuous.



MOO - continuous.

Conclusions

- Despite their principled origins, VAEs are not different from other neural models.
- Similar assumptions = similar problems. We can leverage existing solutions.
- If properly trained, VAEs can be incredibly effective at modeling heterogeneous data.

Want to stay tuned?
Follow us on !

Paper at adrian.javaloy.com/uploads/movae/ymiw-paper.pdf.