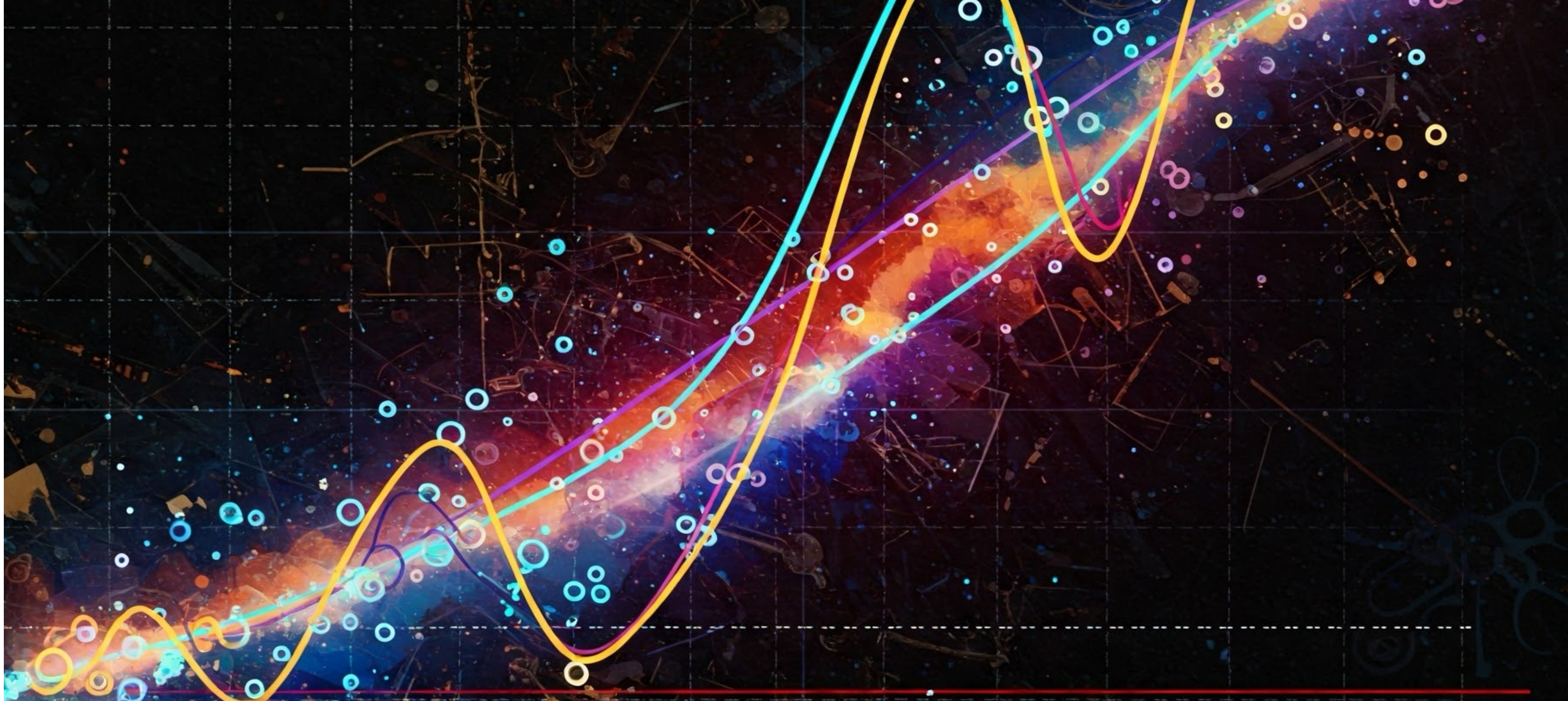
Deep Heteroscedastic Regression

Methods and Applications

deep-regression.github.io

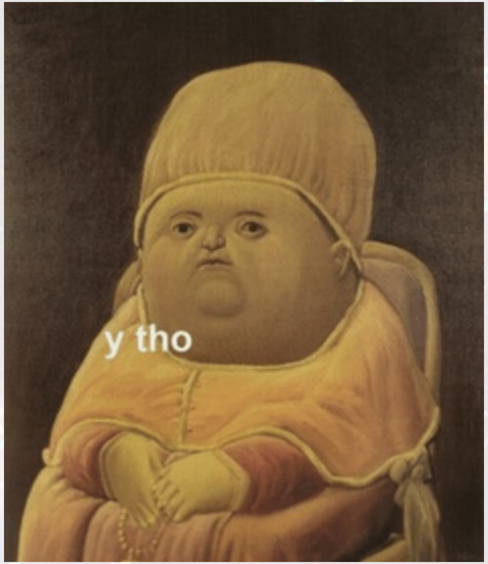


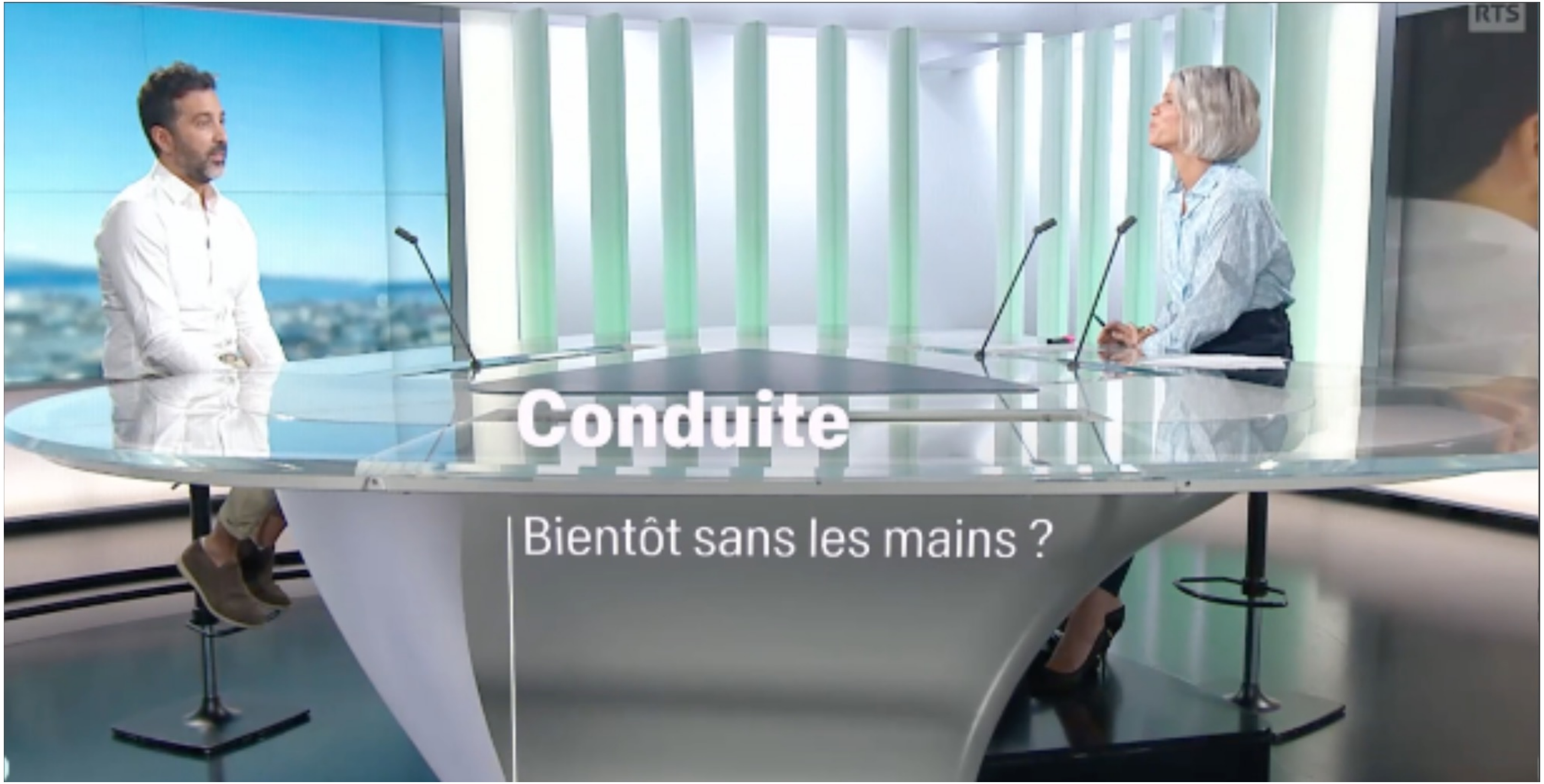
Megh Shukla

February 2025

Deep Heteroscedastic Regression

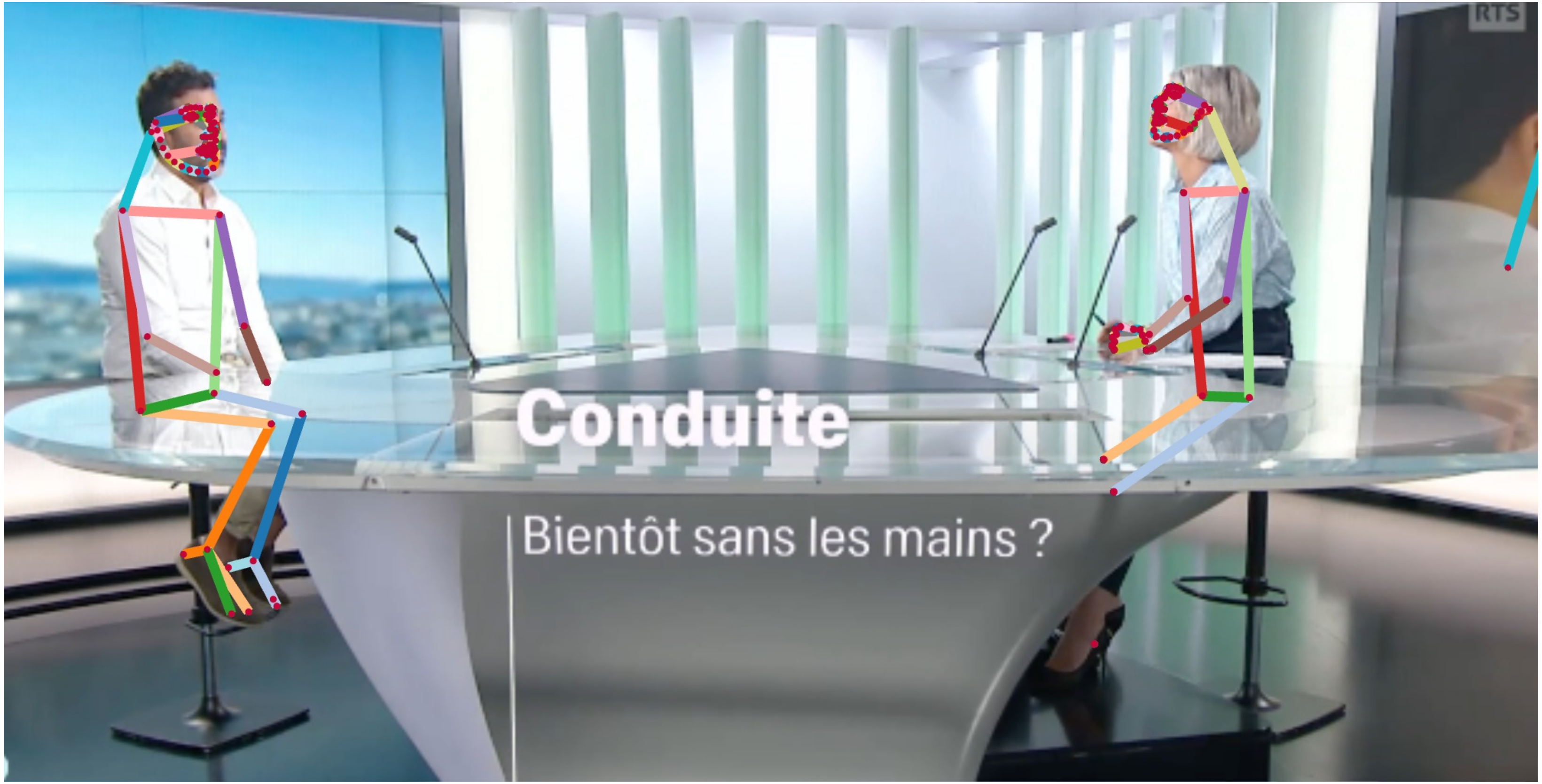
but y tho





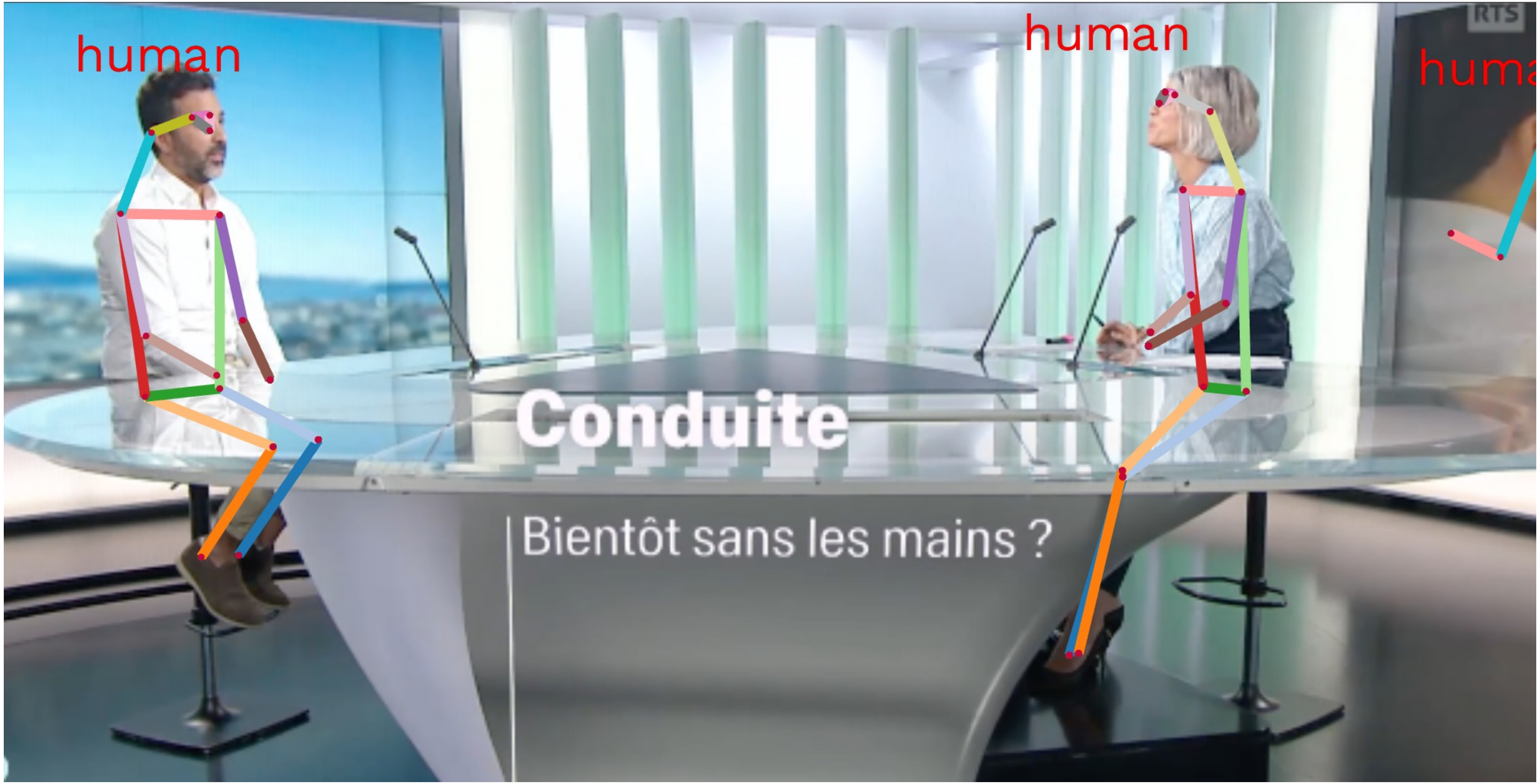
https://www.rts.ch/play/tv/12h45/video/voitures-sans-pilote-et-securite-routiere--entretien-avec-alexandre-alahi-professeur-a-lepfl-en-intelligence-visuelle?urn=urn:rts:video:15163181

OpenPifPaf with 133 keypoints



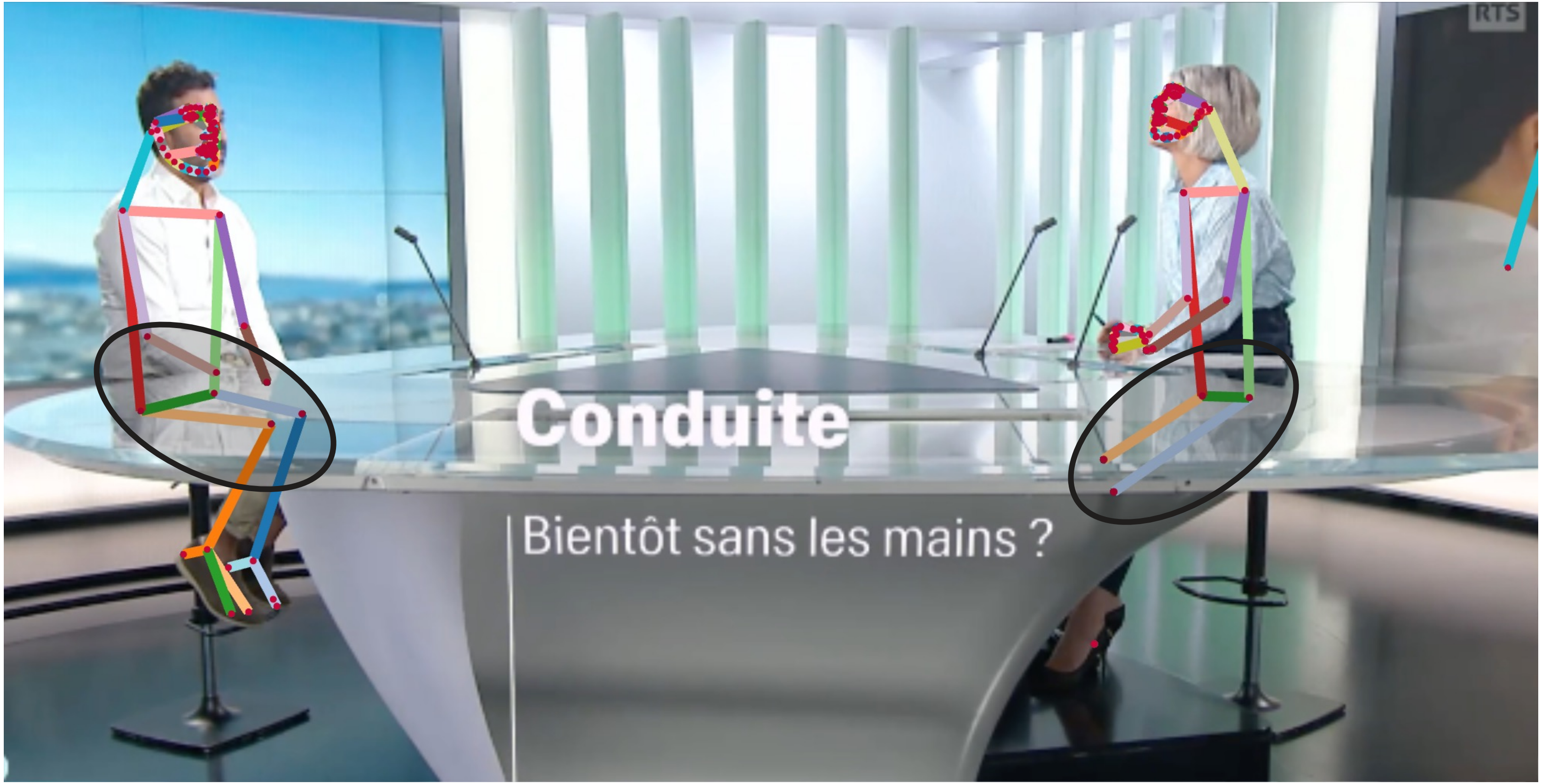
Sven Kreiss et al.. "Openpifpaf: Composite fields for semantic keypoint detection and spatio-temporal association." *IEEE Transactions on Intelligent Transportation Systems* 23, no. 8 (2021)

OpenPifPaf with 17 keypoints

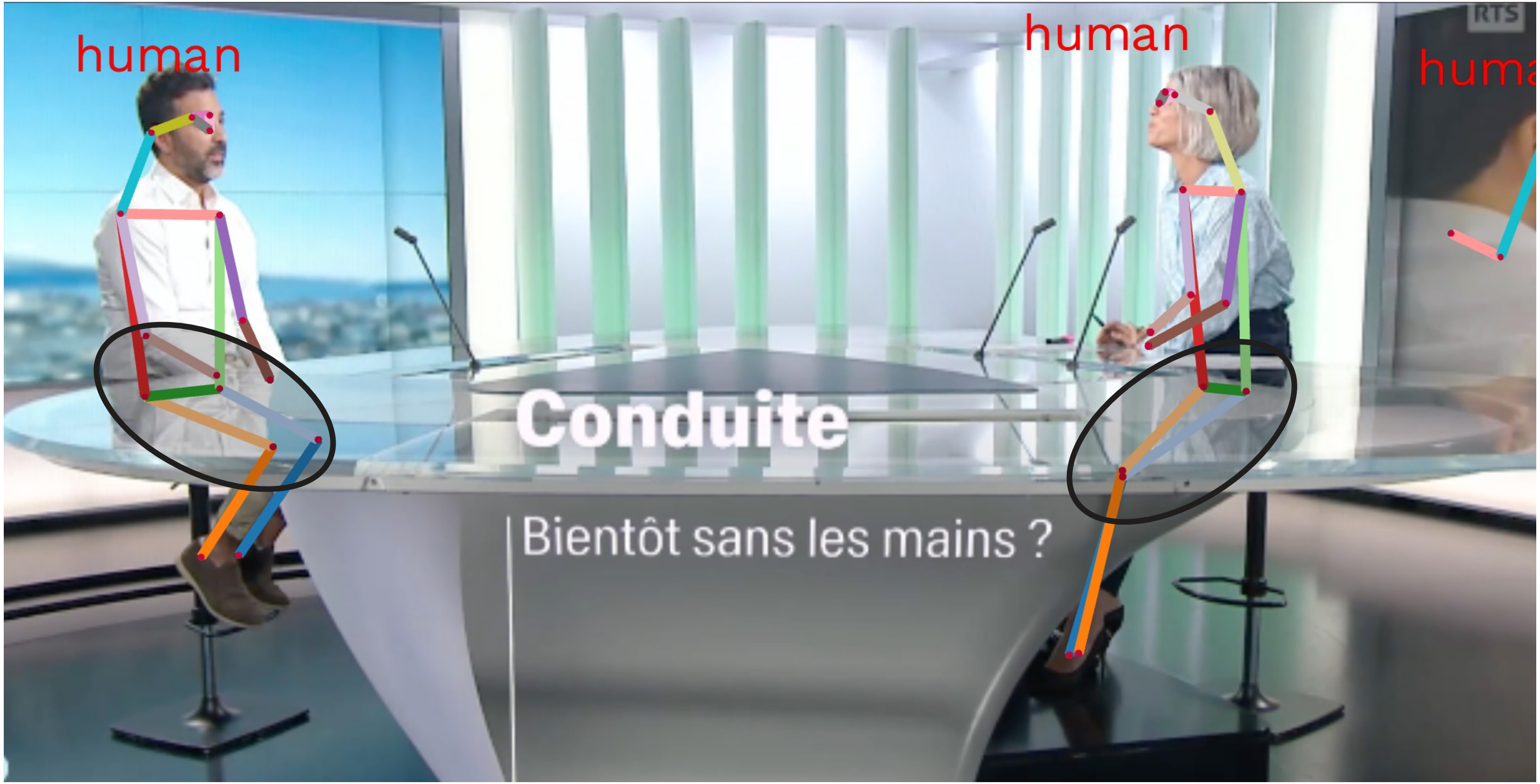


Sven Kreiss et al.. "Openpifpaf: Composite fields for semantic keypoint detection and spatio-temporal association." *IEEE Transactions on Intelligent Transportation Systems* 23, no. 8 (2021)

Deep Heteroscedastic Regression



Deep Heteroscedastic Regression



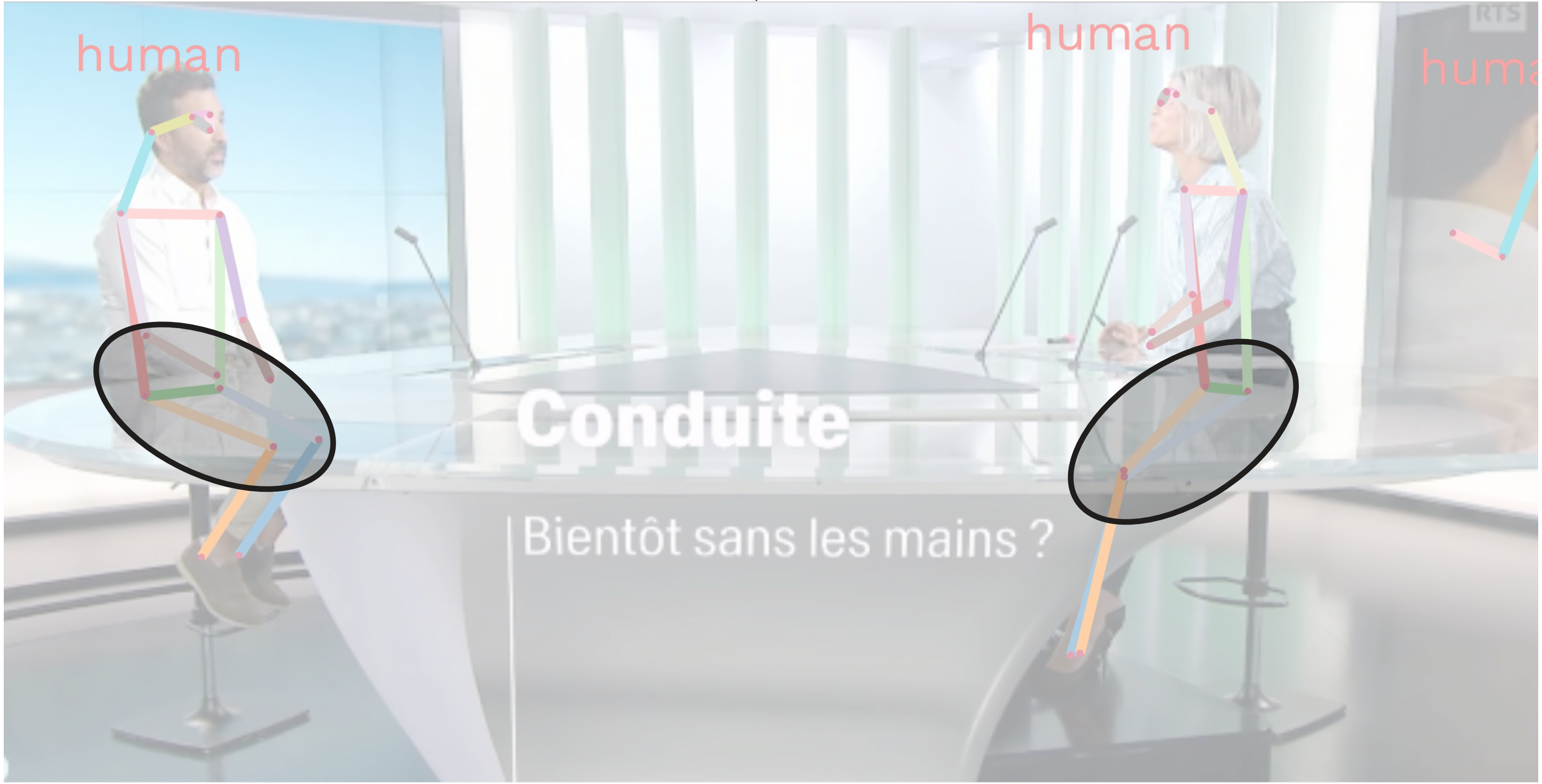
Notice that the occluded joint (knee) is where there is more ambiguity

Predicting distributions >> point predictions!

Deep Heteroscedastic Regression

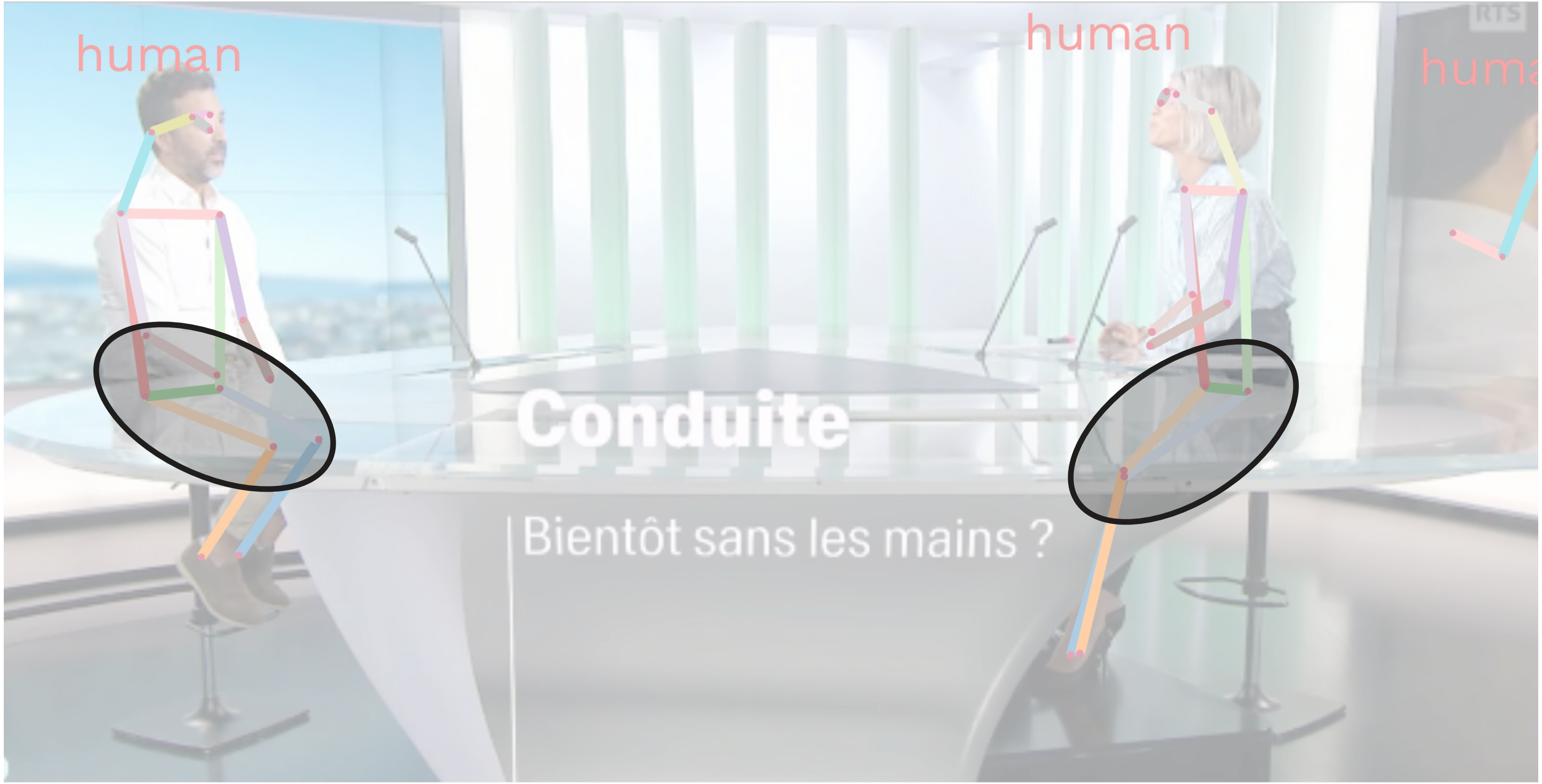
"Heteroscedastic regression: because one-size-fits-all variance just doesn't fit reality!"

Thanks, ChatGPT!

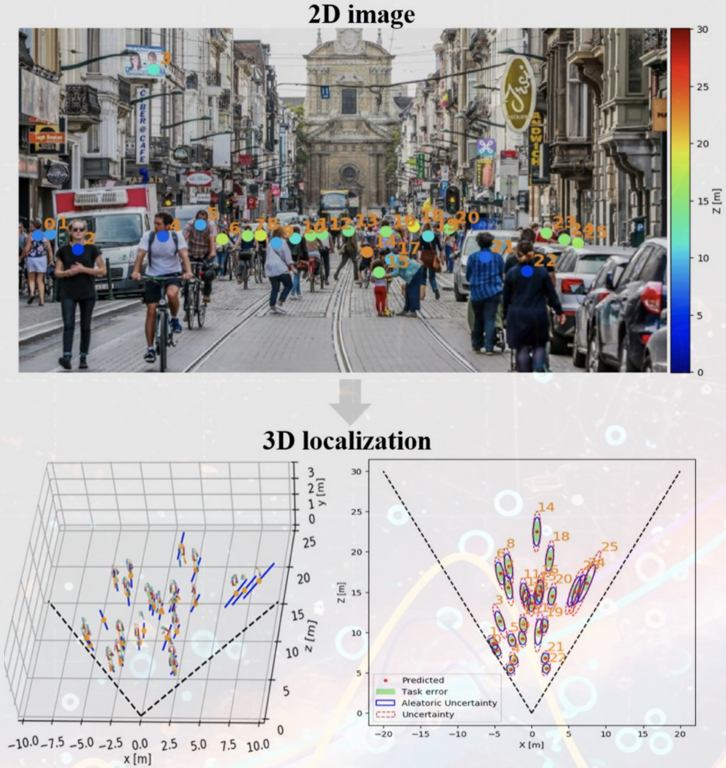


Deep Heteroscedastic Regression

"Heteroscedastic regression: because one-size-fits-all variance just doesn't fit **reality**!"

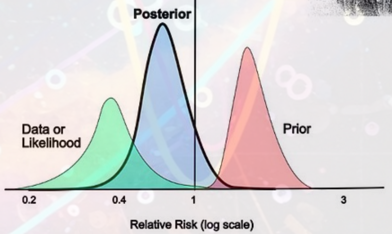


”But where is this **reality**!?"

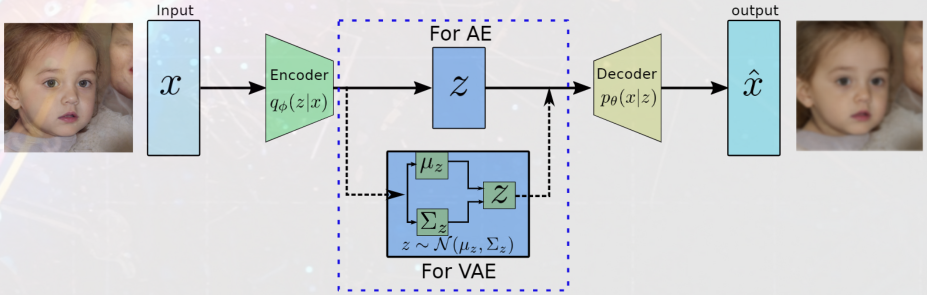


Uncertainty Estimation

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$



Bayesian Frameworks



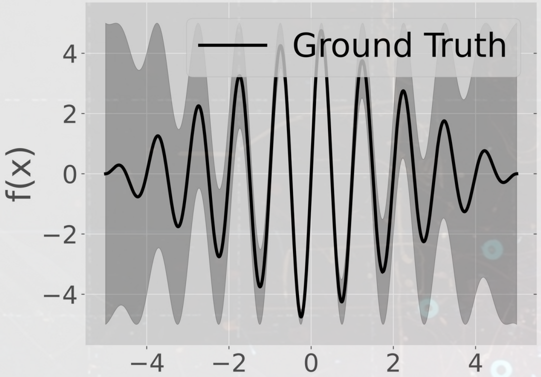
Variational Inference

Moral of the story

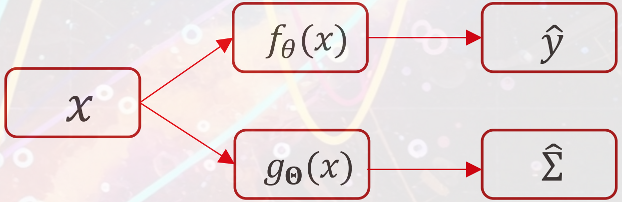
Deep Heteroscedastic Regression

La Science

Deep Heteroscedastic Regression



Regression



Deep

$$-\log p = \frac{1}{2}\log(2\pi\sigma_\Theta^2(x)) + \frac{1}{2}\left(\frac{y-\hat{y}_\theta(x)}{\sigma_\Theta(x)}\right)^2$$

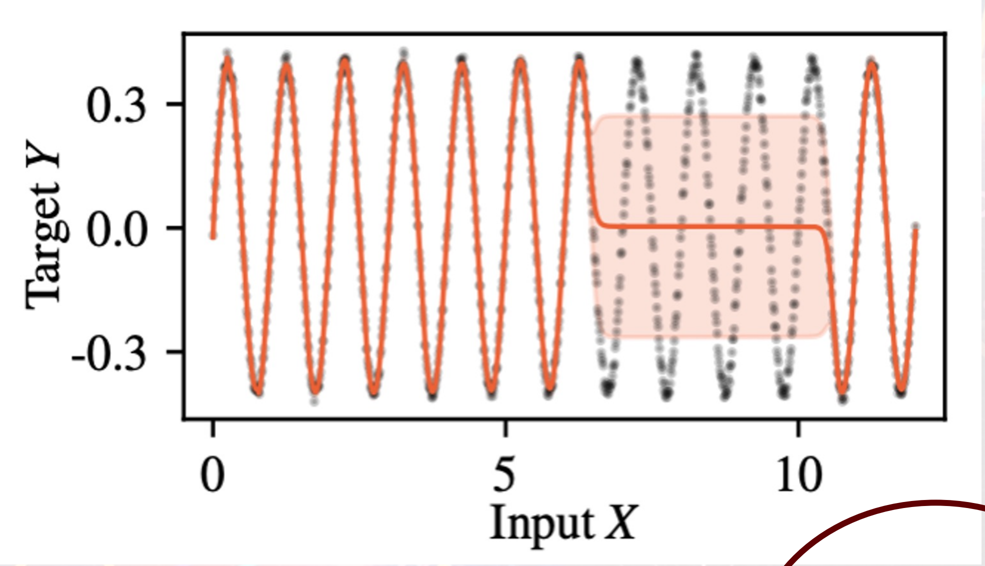
Heteroscedasticity

Variance depends on the input!

EPFL

Deep Heteroscedastic Regression

The Challenge



Without labels, the (co-)variance is estimated only from the residuals.

Loop: Large variance -> mean estimator does not converge -> large variance
Therefore, suboptimal convergence

$$-\log p = \frac{1}{2}\log(2\pi\sigma^2_\Theta(x)) + \frac{1}{2}\left(\frac{y-\hat{y}_\theta(x)}{\sigma_\Theta(x)}\right)^2$$

Seitzer et al., "*On The Pitfalls Of Heteroscedastic Uncertainty Estimation With Probabilistic Neural Networks*". ICLR 2022

EPFL

ON THE PITFALLS OF HETEROSCEDASTIC UNCERTAINTY ESTIMATION WITH PROBABILISTIC NEURAL NETWORKS

Maximilian Seitzer[1], Arash Tavakoli[1], Dimitrije Antić[2], Georg Martius[1]
[1] Max Planck Institute for Intelligent Systems, Tübingen, Germany
[2] University of Tübingen, Tübingen, Germany
maximilian.seitzer@tuebingen.mpg.de

Effective Bayesian Heteroscedastic Regression with Deep Neural Networks

Alexander Immer*,[1,2] Emanuele Palumbo*,[1,3] Alexander Marx†[1,3] Julia E. Vogt†[1]
[1]Department of Computer Science, ETH Zurich, Switzerland
[2]Max Planck Institute for Intelligent Systems, Tübingen, Germany
[3]AI Center, ETH Zurich, Switzerland

Deep Inference for Covariance Estimation: Learning Gaussian Noise Models for State Estimation

Katherine Liu*, Kyel Ok*, William Vega-Brown, and Nicholas Roy

Few-shot Keypoint De

Uncertain

Rebecca L. Russell

Aleatoric Uncertainty in Human Pose

certainty Learning for Unseen Species

Piotr Koniusz*,§,†
§Data61/CSIRO

Reliable training and estimation of variance networks

Nicki S. Detlefsen* † nsde@dtu.dk
Martin Jørgensen* † marjor@dtu.dk
Søren Hauberg † sohau@dtu.dk

Learning Structured Gaussians

Ivor J.A. Simpson
University of Sussex, UK
i.simpson@sussex.ac.uk
Niantic,
svicente@nianti

Faithful Heteroscedastic Regression with Neural Networks

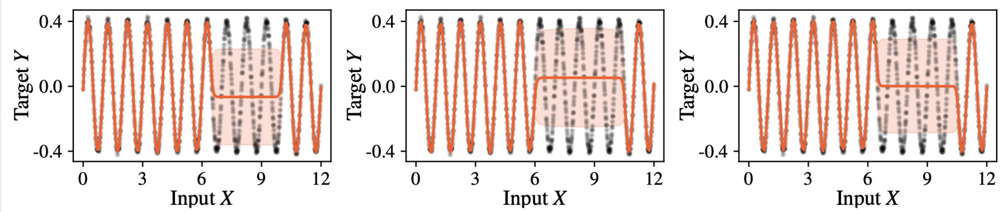
Andrew Stirn, Columbia University (CU)
Hans-Hermann Wessels, New York Genome Center (NYGC)
Megan Schertzer, CU & NYGC
Laura Pereira, NYGC
Neville E. Sanjana, New York University & NYGC
David A. Knowles, CU & NYGC

tainty Prediction Networks

Neill D.F. Campbell[1] Ivor Simpson[2]
[3]University College London
Garoe Dorta
[1]University of Bath, UK
n.campbell@bath.ac.uk

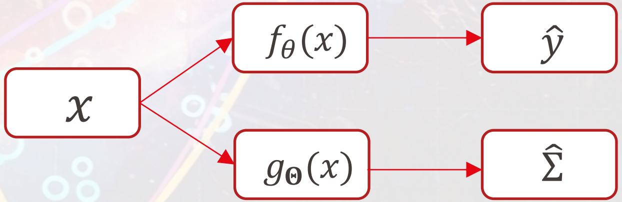
The Problem: Sub-Optimal Convergence

Existing Work: Negative Loglikehood bad ☹



Us: Negative Loglikehood NOT bad ☹

Do we predict the randomness of the predicted mean?




RESEARCH

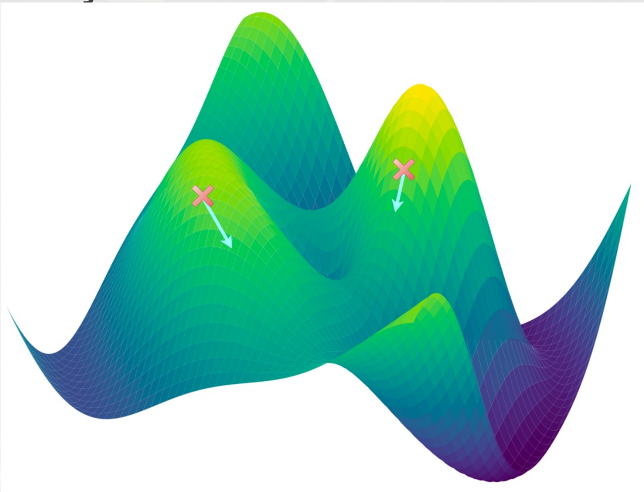
Deep Heteroscedastic Regression

La Méthodologie

TIC-TAC

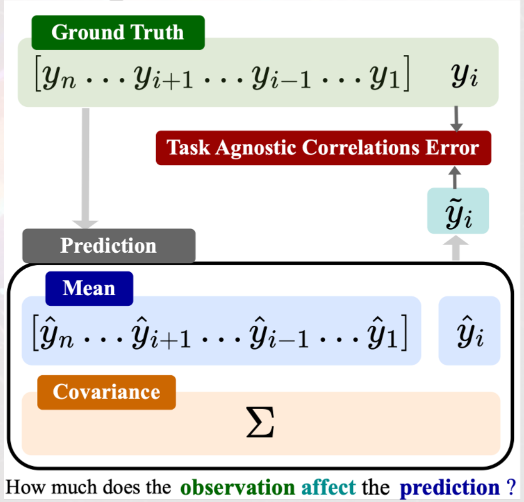
1. Do we predict the randomness of the predicted mean?

Taylor Induced Covariance



2. How do we assess the performance?

Task Agnostic Correlations



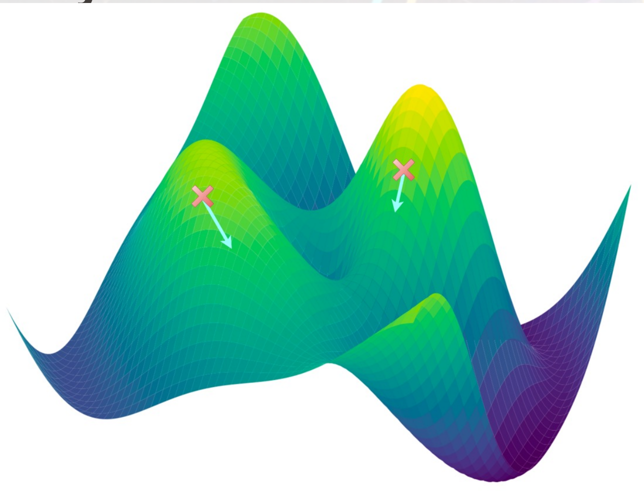
Metric which quantifies the improvement in accuracy through the covariance

Shukla et al., "*TIC-TAC: A Framework for Improved Covariance Estimation in Deep Heteroscedastic Regression*". ICML 2024

TIC-TAC

1. Do we predict the randomness of the predicted mean?

Taylor Induced Covariance

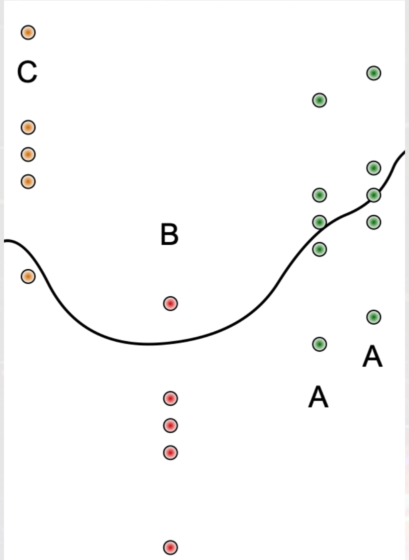


MotionMap
ToSS
TIC-TAC

TIC-TAC

Taylor Induced Covariance

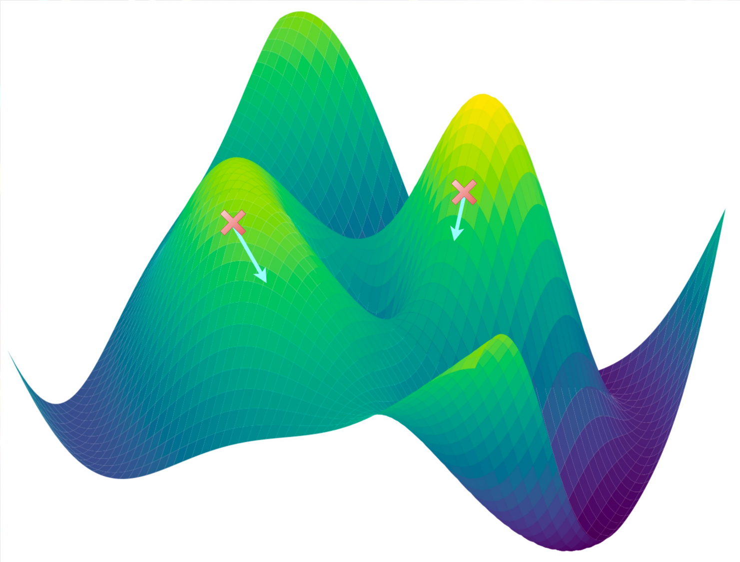
ℰ - Neighborhood



What is $Cov(\hat{Y} \mid X = x)$?
$\ldots\, p(X = x) = 0$

$$X \in [x - \delta, x + \delta]$$

ℰ - Neighborhood

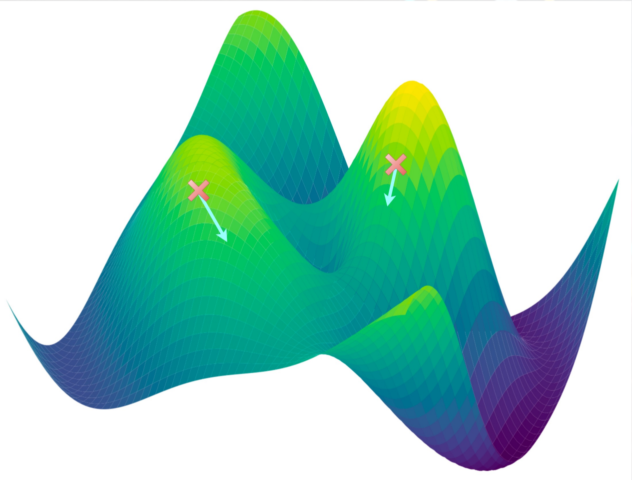


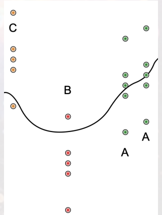
Tie the variance of the prediction to its gradient and curvature

Gradient and Curvature

TIC-TAC

Taylor Induced Covariance



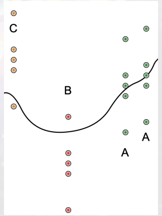


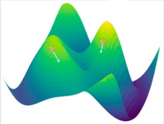
$$f_\theta(\boldsymbol{x} + \epsilon) = f_\theta(\boldsymbol{x}) + \boldsymbol{J}(\boldsymbol{x})\epsilon^T + \frac{\boldsymbol{h}}{2}$$

$$\text{where } \boldsymbol{h}_i = \epsilon \mathbf{H}_i(\boldsymbol{x})\epsilon^T \;\; \forall i \in 1 \ldots n$$

TIC-TAC

Taylor Induced Covariance





$$\mathrm{Cov}\, f_\theta(\boldsymbol{x} + \epsilon) = \mathrm{Cov}(\boldsymbol{J}(\boldsymbol{x})\epsilon^T) + \mathrm{Cov}(\frac{\boldsymbol{h}}{2}) + 2\left[\mathrm{Cov}(\boldsymbol{J}(\boldsymbol{x})\epsilon^T, \frac{\boldsymbol{h}}{2})\right]$$

EPFL

TIC-TAC

Taylor Induced Covariance

$\mathcal{E}$ - Neighborhood

Taylor Polynomial

Covariance

$$\mathrm{Cov}\, f_\theta(\boldsymbol{x}+\epsilon) = \underbrace{\mathrm{Cov}(\boldsymbol{J}(\boldsymbol{x})\epsilon^T) + \mathrm{Cov}(\frac{\boldsymbol{h}}{2})}_{\text{Known solutions!}}$$

$$\mathrm{Cov}(\boldsymbol{J}(\boldsymbol{x})\ \epsilon^T) = k_1(x)\boldsymbol{J}(\boldsymbol{x})\boldsymbol{J}(\boldsymbol{x})^T$$

$$\mathrm{Cov}(\boldsymbol{h}/2)_{i,j} = k_2(x)\ \texttt{Trace}\ (\mathbf{H}_{i,:,:}(\boldsymbol{x})\ \mathbf{H}_{j,:,:}(\boldsymbol{x}))$$

MotionMap

ToSS

TIC-TAC

TIC-TAC

Taylor Induced Covariance

$$\mathrm{Cov}\, f_\theta(\boldsymbol{x}+\epsilon) = \underbrace{\mathrm{Cov}(\boldsymbol{J}(\boldsymbol{x})\epsilon^T) + \mathrm{Cov}(\frac{\boldsymbol{h}}{2})}_{\text{Known solutions!}} + 2\left[\mathrm{Cov}(\boldsymbol{J}(\boldsymbol{x})\epsilon^T, \frac{\boldsymbol{h}}{2})\right]$$

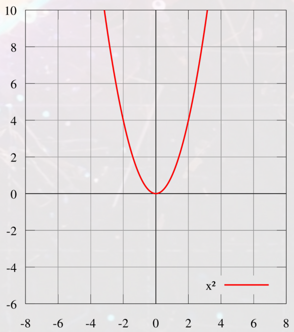
$$\mathbb{E}\big(\boldsymbol{J}_i(\boldsymbol{x})\epsilon^T \epsilon\, \mathbf{H}_k(\boldsymbol{x})\epsilon^T\big) - \mathbb{E}\big(\boldsymbol{J}_i(\boldsymbol{x})\epsilon^T\big)\mathbb{E}\big(\epsilon\, \mathbf{H}_k(\boldsymbol{x})\epsilon^T\big)$$

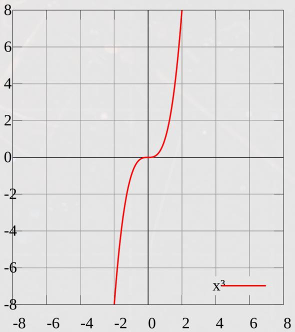
$$= 0$$

$$\mathrm{Cov}(\boldsymbol{J}(\boldsymbol{x})\ \epsilon^T) = k_1(x)\boldsymbol{J}(\boldsymbol{x})\boldsymbol{J}(\boldsymbol{x})^T$$

$$\mathrm{Cov}(\boldsymbol{h}/2)_{i,j} = k_2(x)\ \texttt{Trace}\ (\mathbf{H}_{i,:,:}(\boldsymbol{x})\ \mathbf{H}_{j,:,:}(\boldsymbol{x}))$$

Odd and even functions





$$p(\boldsymbol{\epsilon}) = \mathcal{N}(0, \sigma_\epsilon^2(\boldsymbol{x})\boldsymbol{I}_m)$$

TIC-TAC

Taylor Induced Covariance

ℰ - Neighborhood

Taylor Polynomial

Covariance

$$\mathrm{Cov}(\hat{Y}|X=x) \approx k_1(\boldsymbol{x})\boldsymbol{J}(\boldsymbol{x})\boldsymbol{J}(\boldsymbol{x})^T + \mathcal{H} + k_3(\boldsymbol{x})$$

$$\text{where } \mathcal{H}_{i,j} = k_2(\boldsymbol{x})\,\mathtt{Trace}\left(\mathbf{H}_{i,:,:}(\boldsymbol{x})\,\mathbf{H}_{j,:,:}(\boldsymbol{x})\right)$$

Tie the variance of the prediction
to its gradient and curvature

MotionMap

ToSS

TIC-TAC

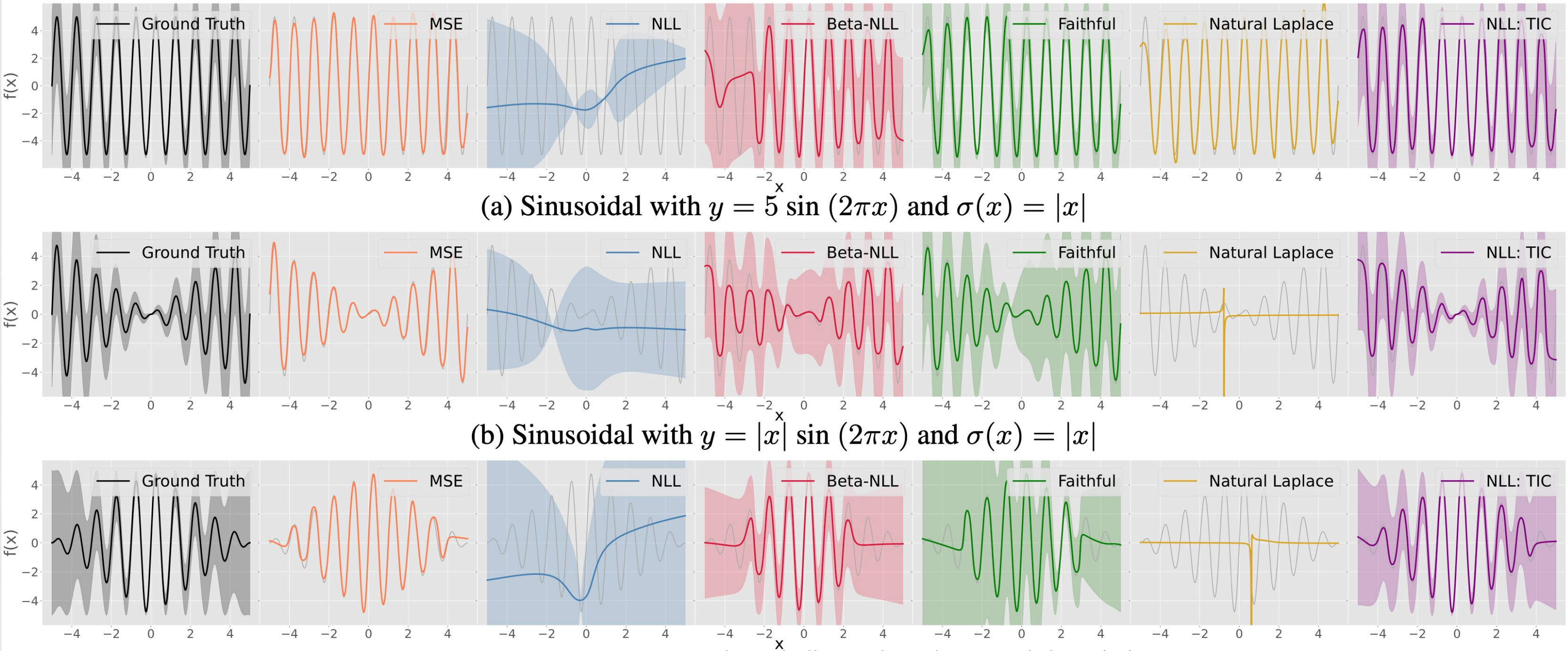
TIC-TAC

Results

EPFL

TIC-TAC

Results - Univariate



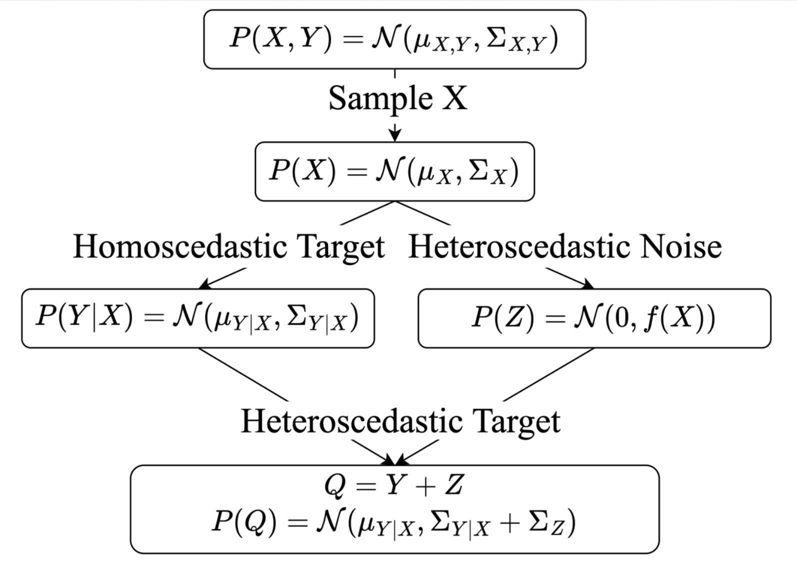
(a) Sinusoidal with $y = 5\sin(2\pi x)$ and $\sigma(x) = |x|$

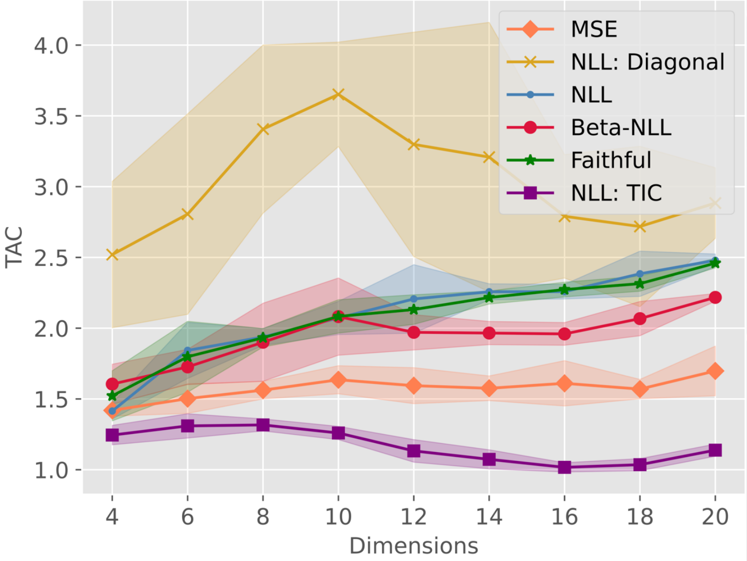
(b) Sinusoidal with $y = |x|\sin(2\pi x)$ and $\sigma(x) = |x|$

(c) Sinusoidal with $y = (5 - |x|)\sin(2\pi x)$ and $\sigma(x) = |x|$

TIC-TAC

Results - Multivariate





Method	Dim: 4	6	8	10	12	14	16	18	20
MSE	-10.1	-16.4	-23.1	-30.9	-36.1	-41.6	-49.2	-53.2	-66.6
NLL	-8.2	-14.9	-19.9	-26.6	-34.2	-42.7	-46.6	-60.9	-67.2
Faithful	-8.7	-14.7	-20.3	4.94	-32.4	-40.2	-48.6	-55.4	-69.0
NLL-TIC	**-7.6**	**-11.7**	**-15.8**	**-19.9**	**-23.3**	**-26.8**	**-30.3**	**-34.2**	**-39.7**

TIC-TAC

Results - UCI

(a) Task Agnostic Correlations (TAC) Metric

Method	Abalone	Air	Appliances	Concrete	Electrical	Energy	Turbine	Naval	Parkinson	Power	Red Wine	White Wine
MSE	2.54	4.31	1.79	6.15	7.91	4.40	4.74	0.56	2.32	6.01	5.97	6.32
NLL-Diagonal	5.49	8.03	11.71	7.86	10.06	7.12	7.07	5.01	8.56	8.16	7.96	8.44
NLL	3.28	3.42	2.41	4.16	7.14	5.10	3.40	0.25	1.86	6.22	5.81	7.26
β-NLL	2.85	5.67	4.89	7.21	8.41	6.17	5.03	1.06	5.48	6.73	6.96	7.08
Faithful	2.96	3.27	1.79	3.93	7.36	2.90	3.29	**0.20**	**1.68**	5.81	5.74	6.89
NLL-TIC	**1.83**	**2.27**	**1.39**	**2.82**	**4.89**	**2.34**	**2.40**	0.28	2.54	**3.87**	**4.05**	**4.60**

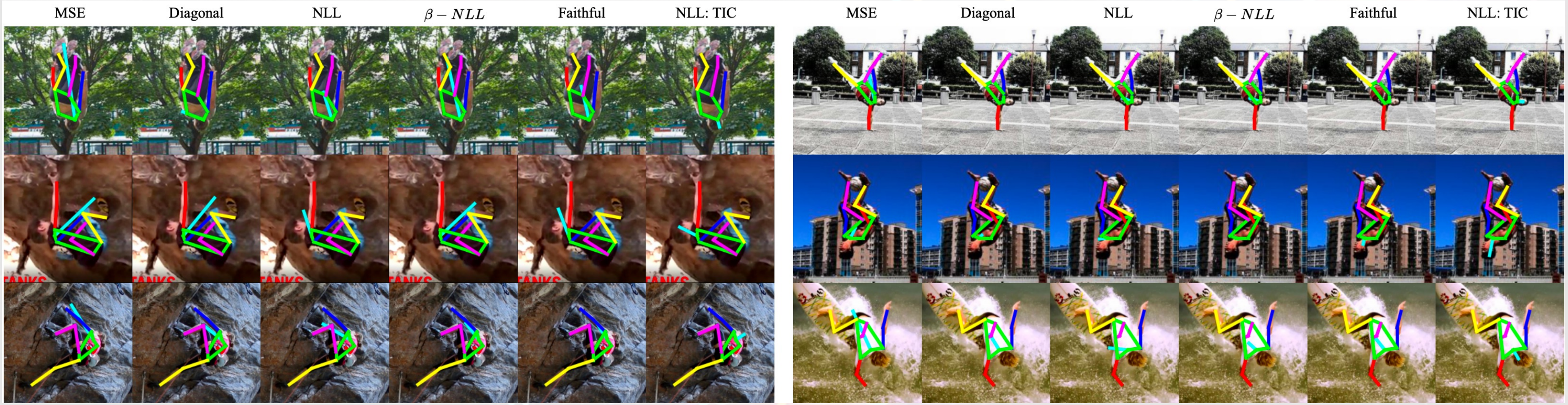
(b) Log Likelihood Metric. We skip NLL-Diagonal and β-NLL which have very low likelihoods since the methods assume diagonal covariance. We remind the reader that the datasets are adapted for covariance estimation

Method	Abalone	Air	Appliances	Concrete	Electrical	Energy	Turbine	Naval	Parkinson	Power	Red Wine	White Wine
MSE	-60.7	-231.5	-99.6	-238.3	-494.6	-169.6	-230.8	-20.9	-154.0	-295.6	-305.8	-338.15
NLL	-8.5×10^3	-53.32	-84.5	-83.6	-57.9	-55.8	-27.1	4.1	-1.5×10^3	-34.2	-236.0	-206.0
Faithful	-9.4×10^3	-52.1	-55.4	-80.6	-57.3	-30.8	-26.1	**7.5**	-1.2×10^3	-33.9	-434.4	-250.9
NLL-TIC	**-13.4**	**-29.3**	**-42.45**	**-22.2**	**-35.8**	**-19.1**	**-22.9**	-10.3	**-63.0**	**-27.1**	**-30.63**	**-30.1**

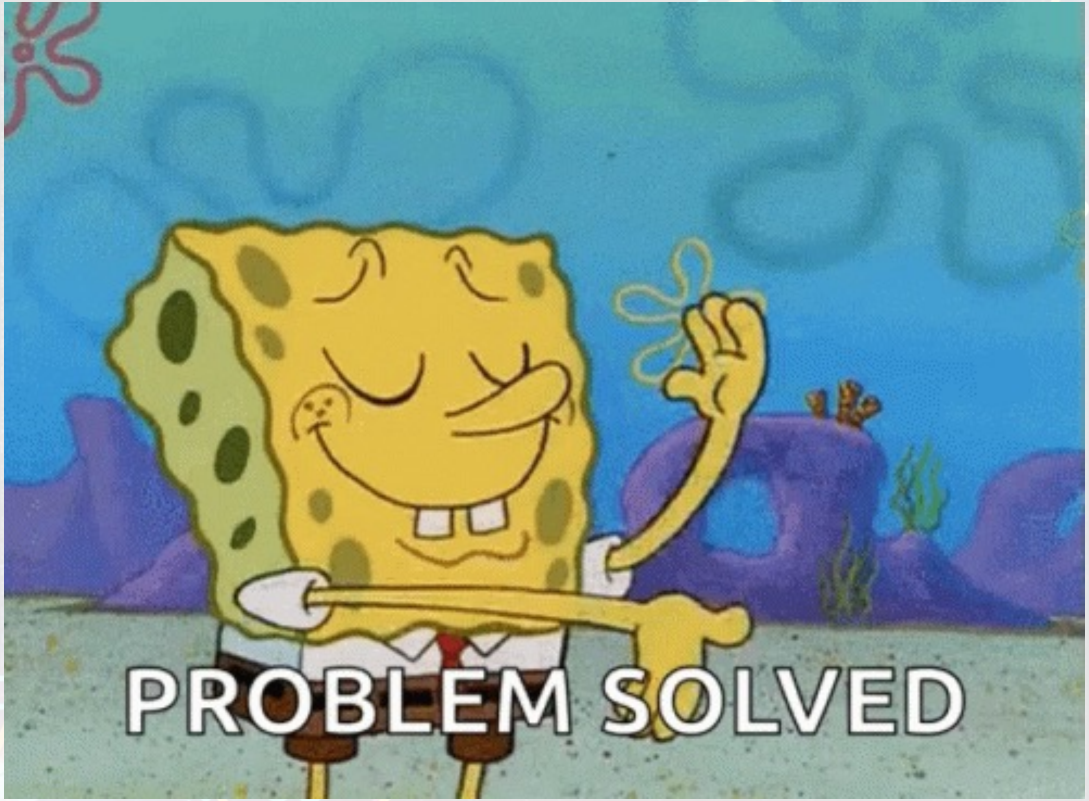
MotionMap
ToSS
TIC-TAC

TIC-TAC

Results – Human Pose

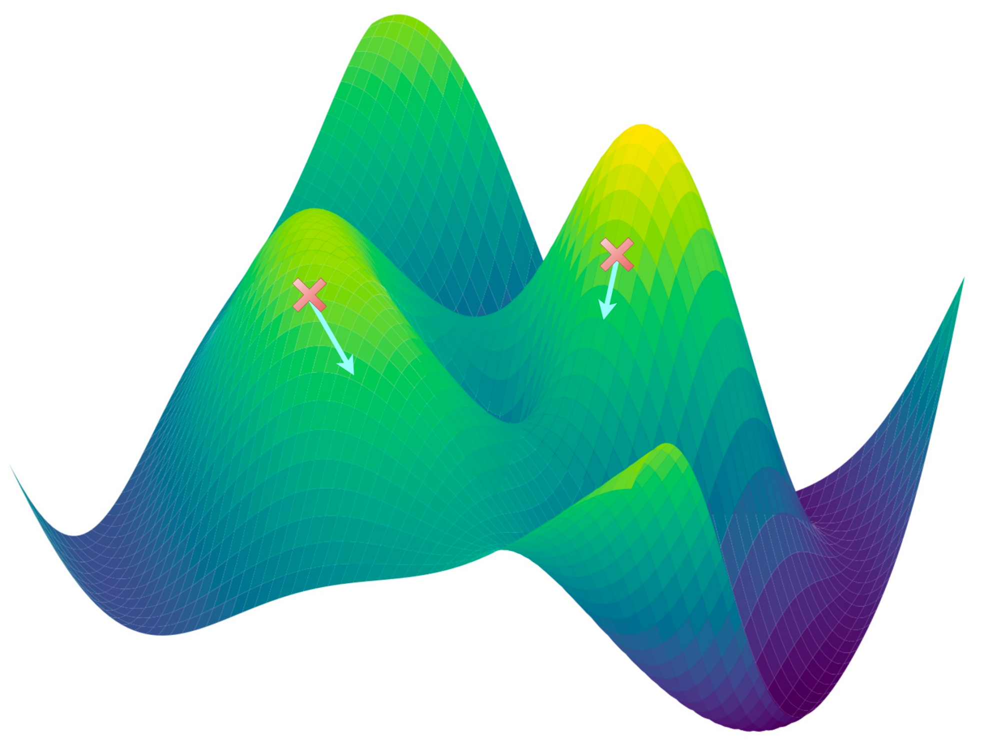


Method	head	neck	lsho	lelb	lwri	rsho	relb	rwri	lhip	lknee	lankl	rhip	rknee	rankl	Avg: TAC	Avg: LL
MSE	6.14	7.12	7.05	8.60	10.56	6.78	8.33	10.35	7.67	7.90	9.69	7.40	7.82	9.72	8.22 ± 0.05	-973.7 ± 8.6
NLL-Diagonal	14.88	12.33	12.38	12.25	13.87	11.36	11.39	13.54	10.42	11.49	17.84	9.84	11.46	18.28	12.95 ± 1.36	-204.5 ± 177.0
NLL	4.97	5.76	4.86	4.58	6.62	4.48	4.36	6.59	5.97	5.80	7.88	5.78	5.68	7.81	5.80 ± 0.07	-91.61 ± 1.26
β-NLL	12.63	11.22	11.63	12.06	13.95	10.45	11.21	13.63	10.84	11.45	16.23	10.02	11.09	15.97	12.31 ± 0.31	-4.2e3 ± 1.6e3
Faithful	5.25	5.86	4.97	4.68	6.77	4.60	4.45	6.75	6.10	5.98	7.90	5.94	5.82	7.89	5.93 ± 0.03	-91.77 ± 0.11
NLL-TIC	**3.97**	**5.38**	**4.47**	**4.29**	**6.06**	**4.12**	**4.08**	**5.89**	**5.45**	**5.24**	**7.03**	**5.25**	**5.09**	**6.97**	**5.23 ± 0.03**	**-80.31 ± 0.39**


PROBLEM SOLVED

MotionMap
ToSS
TIC-TAC

EPFL



The Taylor Induced Covariance is computationally expensive in memory and time

Towards Self-Supervised Covariance Estimation

(Q1) How can we supervise the learning of the covariance assuming annotations are available?

(Q2) How can we obtain pseudo-labels for the covariance when annotations are not available?

Shukla et al., "*Towards Self-Supervised Covariance Estimation in Deep Heteroscedastic Regression*". ICLR 2025

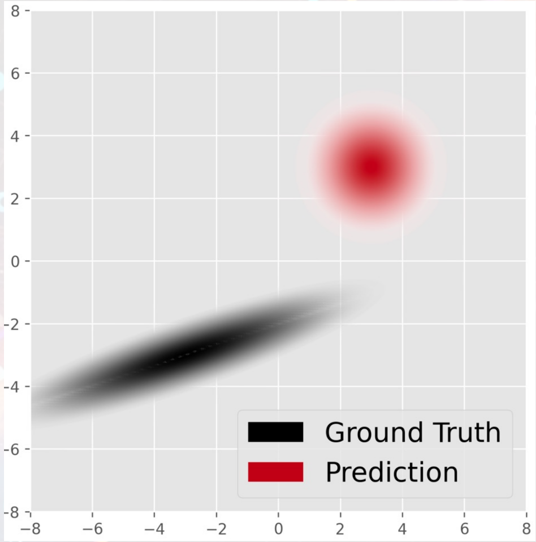
Towards Self-Supervised Covariance Estimation

(Q1) How can we supervise the learning of the covariance assuming annotations are available?

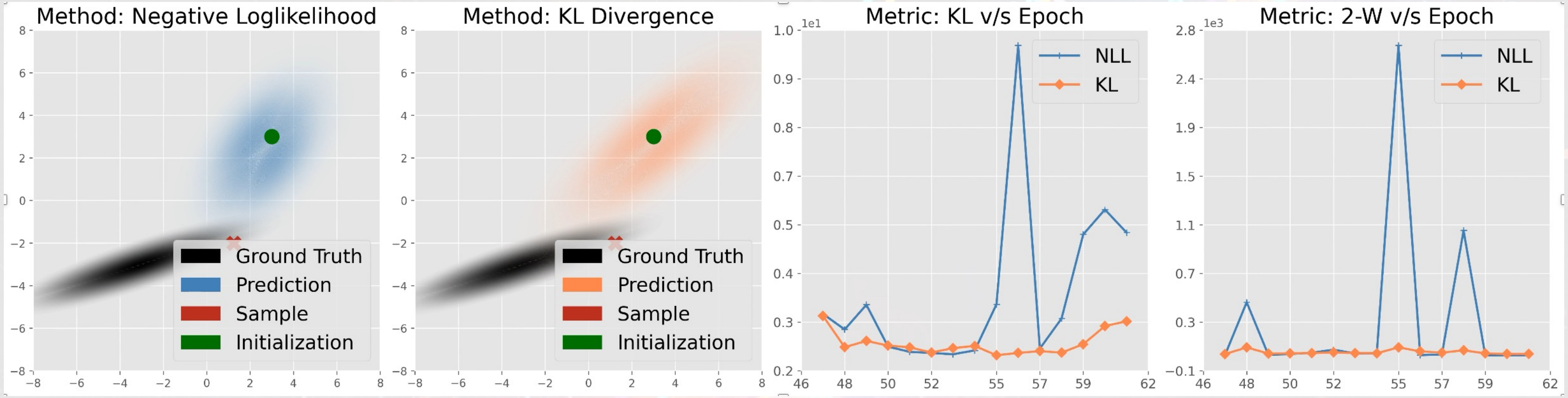
KL Divergence

2-Wasserstein Distance

Analysis Setting



What happens when using the KL Divergence?



Observations:

1. NLL's predicted covariance is significantly chaotic **(why?)**
2. Emergence of the residual covariance affects optimisation for both NLL, KL **(how?)**

What happens when using the KL Divergence?

Observations:

1. **NLL's predicted covariance is significantly chaotic (why?)**
2. Emergence of the residual covariance affects optimisation for both NLL, KL **(how?)**

Lemma 1 (Calibration). *Let* $\mathbf{S} = \{\mathbf{x}, \mathbf{y_i}\}_{\mathbf{i=1}}^{\mathbf{N}}$ *be a set of samples drawn from the unknown target* $P(Y|X) = \mathcal{N}(\mu_Y(X), \Sigma_Y(X))$ *for a given* $\boldsymbol{x}$*. We write each label* $\boldsymbol{y}_i$ *as a distribution* $\mathcal{N}(\boldsymbol{y}_i, \Sigma_Y^{(prior)}(X))$*. Then, the optimal solution using the KL Divergence for the predicted covariance over the set* $\mathbf{S}$ *is* $\widehat{\Sigma}_Y(X) \approx \Sigma_Y(X) + \Sigma_Y^{(prior)}(X)$*. Consequently, if the target covariance is known and set as the prior, we have* $\widehat{\Sigma}_Y(X) \approx 2\,\Sigma_Y(X)$.

Discussion. The optimal solution for $\widehat{\Sigma}_Y(X)$ is twice the target covariance. This can be addressed by a simple calibration of the KL Divergence (Eq. 2);

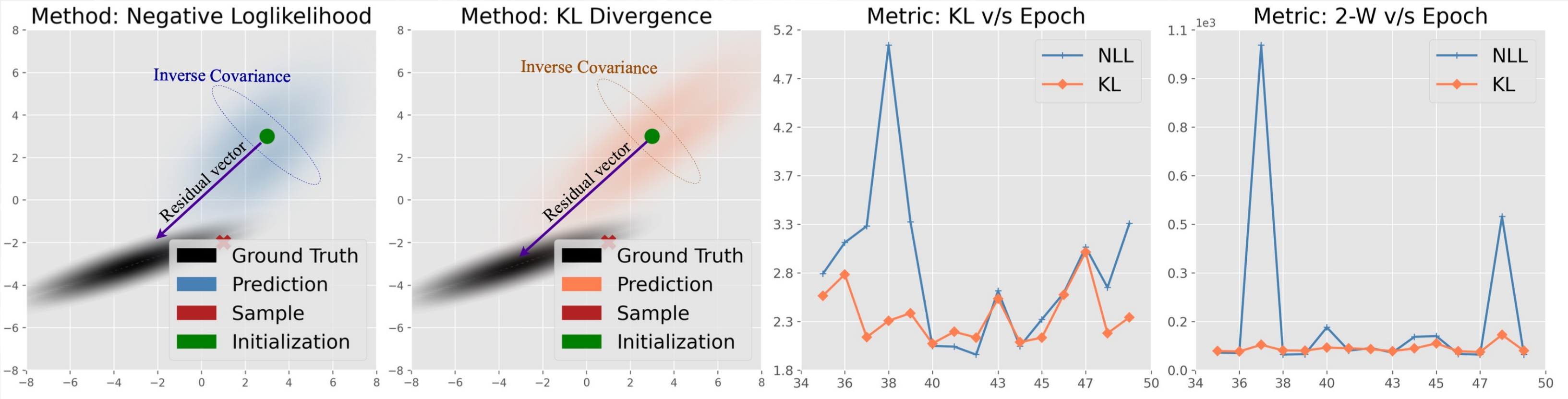
$$\frac{1}{2}\left[\frac{\mathrm{Tr}(\widehat{\Sigma}_Y^{-1}\Sigma_Y^{(\text{prior})}) + (\widehat{\mu}_Y - \boldsymbol{y})^\top \widehat{\Sigma}_Y^{-1}(\widehat{\mu}_Y - \boldsymbol{y})}{2} - k + \ln\left(\frac{\det \widehat{\Sigma}_Y}{\det \Sigma_Y^{(\text{prior})}}\right)\right].$$

What happens when using the KL Divergence?

Observations:

1. NLL's predicted covariance is significantly chaotic (why?)
2. **Emergence of the residual covariance affects optimisation for both NLL, KL (how?)**

$$\widehat{\Sigma}_Y(X) = \Sigma_Y^{(\text{prior})}(X) + \frac{1}{N}\sum_{i=1}^{N}(\widehat{\mu}_Y(X) - \boldsymbol{y}_i)(\widehat{\mu}_Y(X) - \boldsymbol{y}_i)^T$$



What happens when using the 2-Wasserstein Distance?

$$||\mu_1 - \mu_2||^2 + \mathrm{Tr}\left[\Sigma_1 + \Sigma_2 - 2(\Sigma_2^{1/2}\Sigma_1\Sigma_2^{1/2})^{1/2}\right]$$

Observation: **Eigendecomposition may be unstable through gradient descent!**

If Σ_1 and Σ_2 are commutative

$$\mathcal{W}_2(\mathcal{N}_1, \mathcal{N}_2) = ||\mu_1 - \mu_2||^2 + ||\Sigma_1^{1/2} - \Sigma_2^{1/2}||_F^2$$

What happens when using the 2-Wasserstein Distance?

$$||\mu_1 - \mu_2||^2 + \mathrm{Tr}\left[\Sigma_1 + \Sigma_2 - \underbrace{2(\Sigma_2^{1/2}\Sigma_1\Sigma_2^{1/2})^{1/2}}\right]$$

Observation: **Eigendecomposition may be unstable through gradient descent!**

If Σ_1 and Σ_2 are commutative $\quad \mathcal{W}_2(\mathcal{N}_1, \mathcal{N}_2) = ||\mu_1 - \mu_2||^2 + ||\Sigma_1^{1/2} - \Sigma_2^{1/2}||_F^2$

But what about the general non-commutative case?

Theorem 1 (2-Wasserstein bound for non-commutative covariances). *Let $\mathcal{N}_1(\mu_1, \Sigma_1)$, $\mathcal{N}_2(\mu_2, \Sigma_2)$ be two multivariate normal distributions, where Σ_1 and Σ_2 are non-commutative matrices. Then, the 2-Wasserstein distance between the two distributions has an upper bound of*

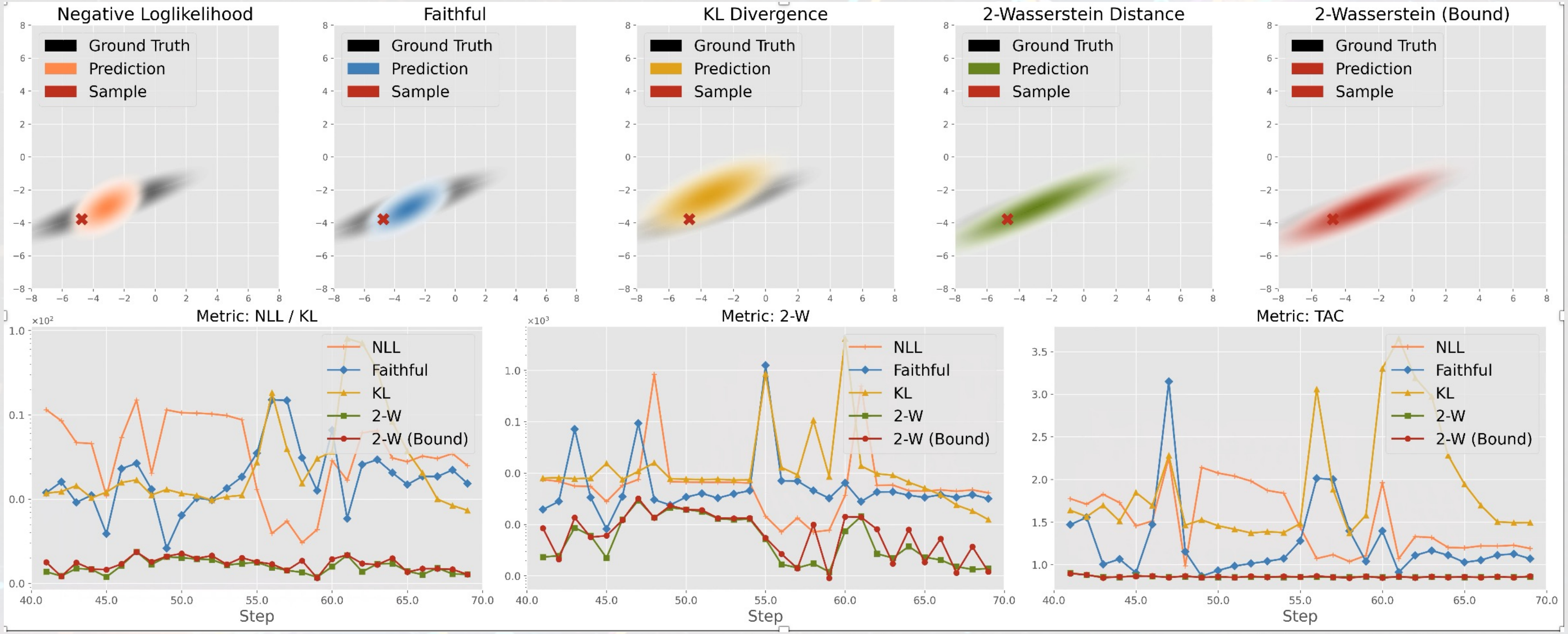
$$\mathcal{W}_2(\mathcal{N}_1, \mathcal{N}_2) \leq ||\mu_1 - \mu_2||_2^2 + ||\Sigma_1^{1/2} - \Sigma_2^{1/2}||_F^2 \, ,$$

where $||(.)||_F$ represents the Frobenius norm of a matrix.

EPFL

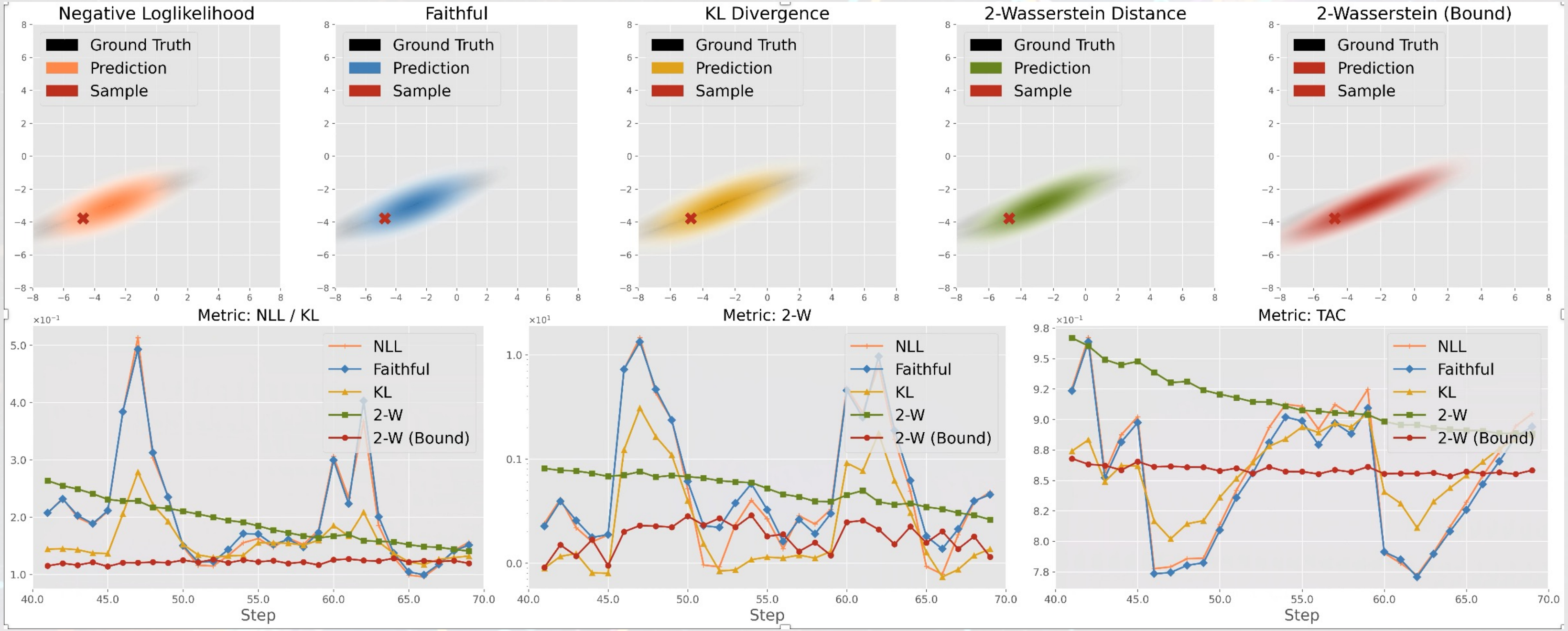
(Q1) How can we supervise the learning of the covariance assuming annotations are available?

What happens when using the 2-Wasserstein Distance?

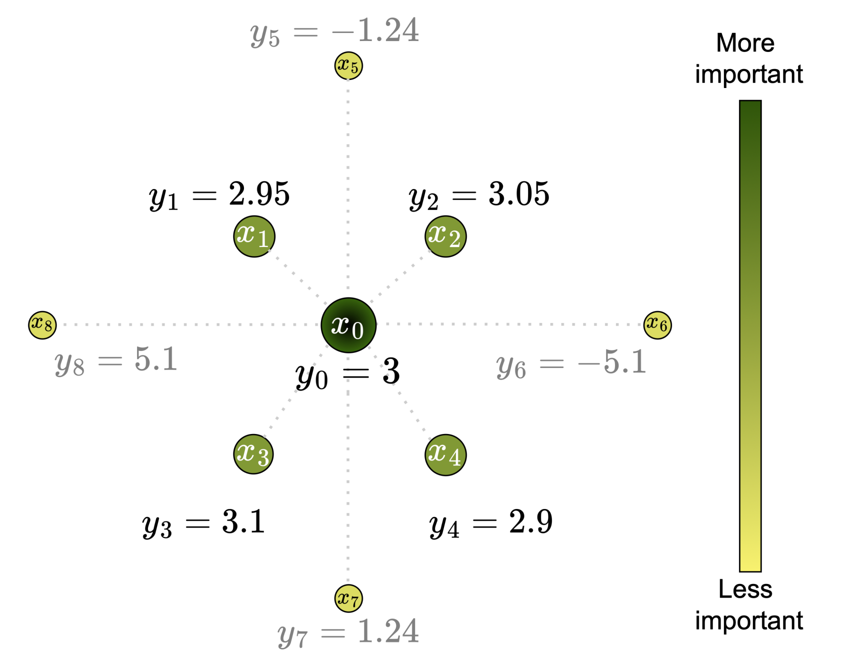


(Q1) How can we supervise the learning of the covariance assuming annotations are available?

What happens when using the 2-Wasserstein Distance?



(Q2) How can we obtain pseudo-labels for the covariance when annotations are not available?



MotionMap
ToSS
TIC-TAC

The key ideas

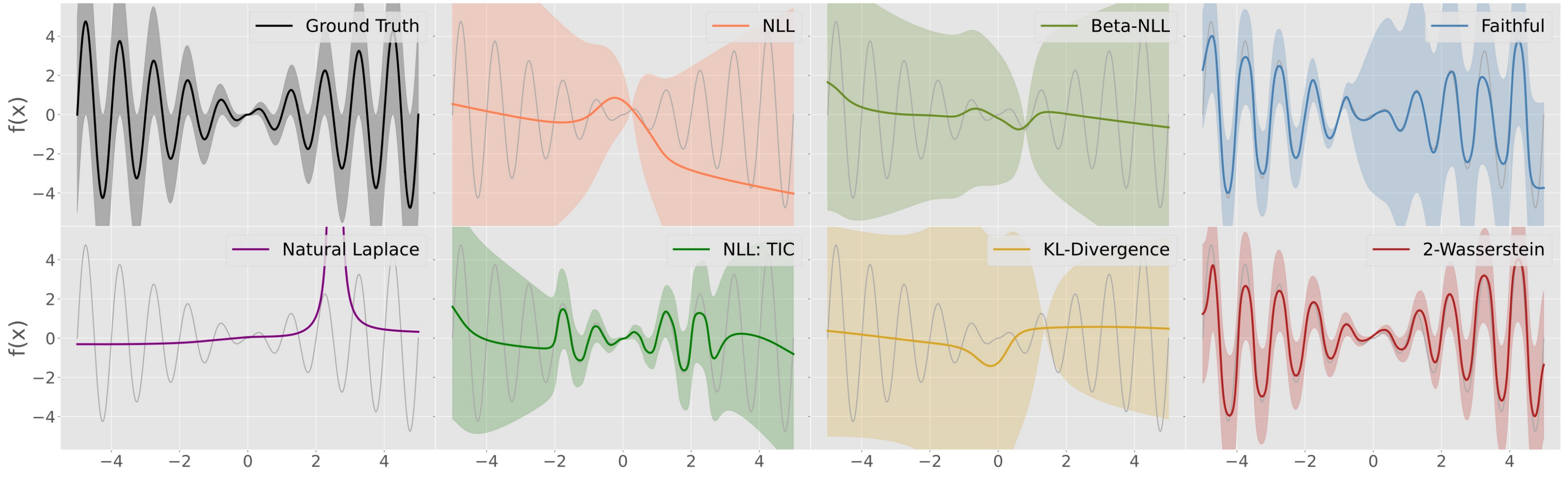
1. The target $\boldsymbol{y}$ has a high (co-)variance if it exhibits large variations in a small vicinity of $\boldsymbol{x}$.
2. The closer $\boldsymbol{x}_j$ is to $\boldsymbol{x}_i$, the likelier it is that $\boldsymbol{y}_j$ is a potential label for $\boldsymbol{x}_i$.

Towards Self-Supervised Covariance Estimation

Results

Towards Self-Supervised Covariance Estimation

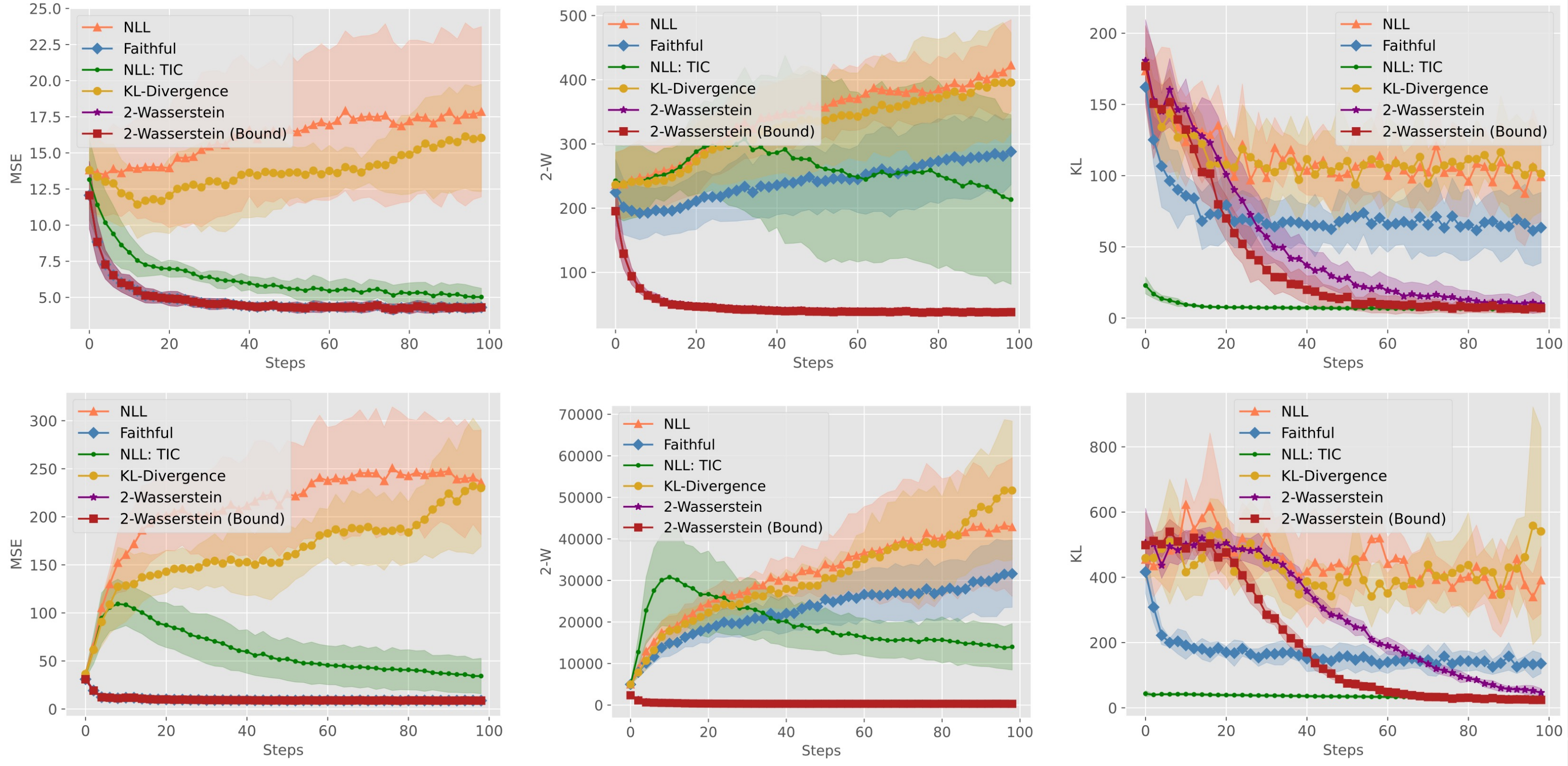
Results - Univariate



EPFL

Towards Self-Supervised Covariance Estimation

Results - Multivariate



MotionMap

ToSS

TIC-TAC

Megh Shukla

Towards Self-Supervised Covariance Estimation

Results - Multivariate

(a) Compute time (in milliseconds)

Dimensions →	4	8	12	16	20	24	28	32
Beta-NLL, Diagonal	2.88	3.15	2.17	2.06	1.83	1.74	2.00	2.04
Faithful, NLL	4.56	4.74	3.94	3.69	3.76	3.66	4.08	4.85
NLL: TIC	56.60	56.81	59.28	95.54	197.58	448.58	943.79	1961.08
KL-Divergence	4.79	5.06	4.05	4.05	4.10	3.94	4.81	5.24
2-Wasserstein	5.10	5.43	4.47	4.38	4.31	4.14	4.88	5.20
2-Wasserstein (Bound)	4.59	4.79	3.72	3.73	3.64	3.56	3.91	4.72

(b) Compute memory (in megabytes)

Dimensions →	4	8	12	16	20	24	28	32
NLL: TIC	11.68	120.84	523.22	1543.77	3625.51	7333.84	13313.31	22398.00
2-Wasserstein (Bound) +6 other methods	3.45	8.24	17.10	29.51	54.51	111.73	201.51	339.55

Towards Self-Supervised Covariance Estimation

Results - UCI

(a) Mean Square Error (MSE)

Method	Abalone	Air	Appliances	Concrete	Electrical	Energy	Gas	Naval	Parkinson	Power	Red Wine	White Wine
NLL	3.74	17.92	53.49	4.57	9.28	4.20	10.98	10.34	54.51	9.09	8.94	9.40
KL-Divergence	1.90	14.70	90.90	3.84	15.57	4.20	10.16	12.39	59.39	9.97	7.26	8.17
Beta-NLL	0.35	1.58	3.69	2.02	3.62	1.87	1.50	0.72	8.11	3.06	2.15	3.43
NLL: Diagonal	1.32	8.90	37.91	4.28	6.58	3.99	5.73	9.60	27.35	6.52	5.75	6.01
Faithful	**0.16**	**0.33**	**0.20**	**0.72**	**0.89**	**0.41**	**0.45**	**0.06**	**0.29**	**0.61**	**0.70**	**0.78**
NLL: TIC	0.21	0.82	4.45	0.96	**0.91**	0.61	0.67	1.36	8.89	**0.66**	0.97	0.92
2-W (Bound)	**0.16**	**0.34**	**0.20**	**0.72**	**0.90**	**0.41**	**0.45**	**0.07**	**0.30**	**0.61**	**0.71**	**0.79**

(b) Negative Log-Likelihood (NLL)

Method	Abalone	Air	Appliances	Concrete	Electrical	Energy	Gas	Naval	Parkinson	Power	Red Wine	White Wine
NLL	35.89	56.98	245.99	28.50	63.15	29.85	41.03	38.18	262.59	49.37	46.58	58.06
KL-Divergence	18.27	83.18	413.96	38.23	73.87	28.50	34.38	49.24	257.36	45.49	61.96	48.66
Beta-NLL	9.80	29.38	60.30	20.45	35.44	20.15	20.26	20.81	59.98	27.64	34.05	30.95
NLL: Diagonal	18.61	80.86	369.67	46.82	65.73	36.09	47.06	77.47	238.71	51.60	77.98	67.21
Faithful	11.86	33.31	65.15	17.42	34.73	19.41	22.47	27.70	57.04	24.08	24.34	26.01
NLL: TIC	**4.71**	16.46	30.41	11.36	**14.97**	12.06	**9.96**	14.99	42.52	**9.31**	14.66	**12.33**
2-W (Bound)	6.32	**13.58**	**22.72**	**8.96**	**15.57**	**8.85**	**10.49**	**11.44**	**21.48**	11.31	**11.65**	**12.12**

(c) Compute time (in milliseconds)

Method	Abalone	Air	Appliances	Concrete	Electrical	Energy	Gas	Naval	Parkinson	Power	Red Wine	White Wine
NLL	5.28	5.45	5.85	5.70	7.53	5.84	5.23	6.44	7.31	5.53	5.95	5.88
Beta-NLL	4.71	4.58	4.98	4.35	5.20	4.41	4.62	5.81	6.70	4.81	6.83	4.41
Faithful	5.28	5.35	5.62	5.73	6.02	5.03	5.03	7.02	6.47	5.81	7.40	5.05
NLL: Diagonal	4.50	4.49	4.84	4.67	5.13	4.32	4.38	5.84	4.98	4.61	5.77	4.36
NLL: TIC	45.61	53.22	68.55	47.46	49.37	47.57	45.09	56.04	59.25	49.74	65.25	45.23
KL-Divergence	5.30	6.65	6.08	5.47	8.09	5.16	5.14	6.85	9.15	5.36	6.93	5.23
2-W (Bound)	4.51	5.83	5.17	4.51	5.36	4.50	4.38	5.23	7.16	4.51	5.28	4.48

(d) Compute memory (in megabytes)

Method	Abalone	Air	Appliances	Concrete	Electrical	Energy	Gas	Naval	Parkinson	Power	Red Wine	White Wine
NLL: TIC	11.22	90.20	820.85	1.09	71.35	24.13	33.99	108.59	637.74	41.57	40.94	41.57
2-W (Bound) +6 other methods	3.10	9.02	25.30	1.09	7.75	4.63	5.56	9.02	23.05	5.56	5.56	5.56

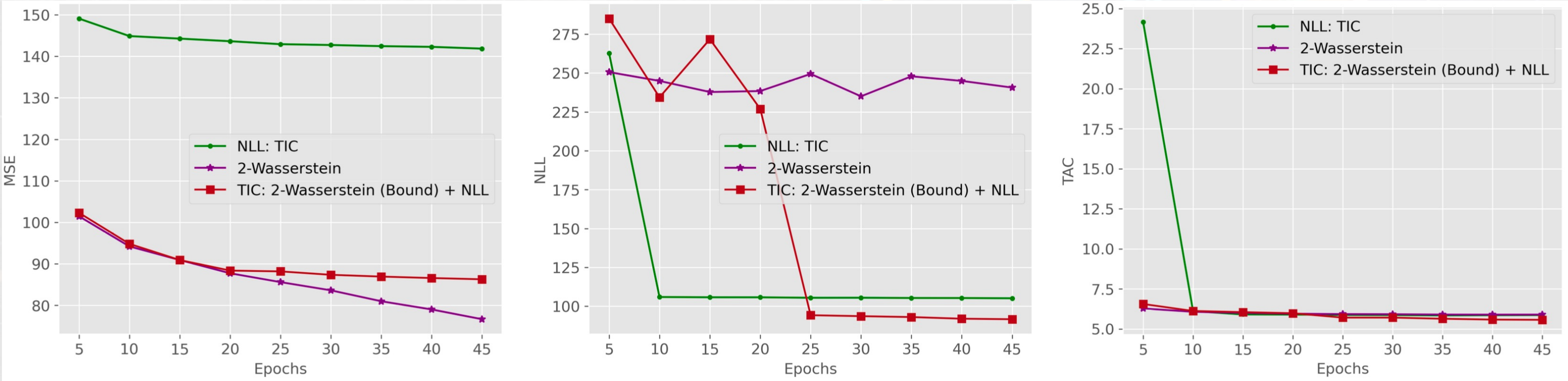
Towards Self-Supervised Covariance Estimation

Results – Human Pose

The Self-Supervised framework did not beat SoTA (finally)!

Towards Self-Supervised Covariance Estimation

Results – Human Pose

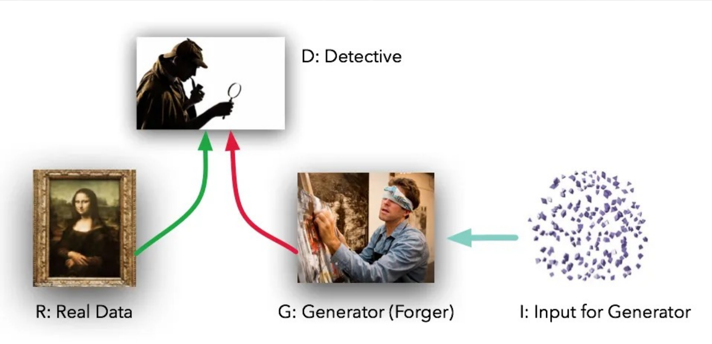


Potential for a hybrid approach: pre-train with 2-Wasserstein, train with NLL-TIC

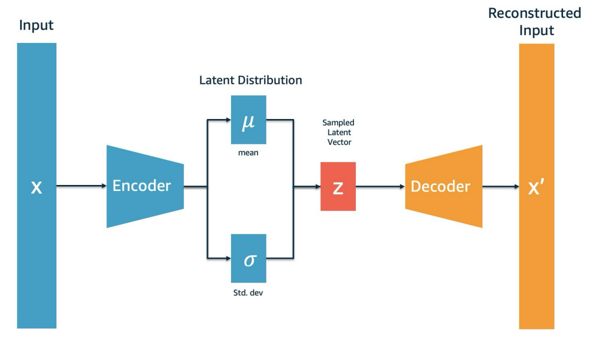
Deep Heteroscedastic Regression

La Méthodologie et L'Application

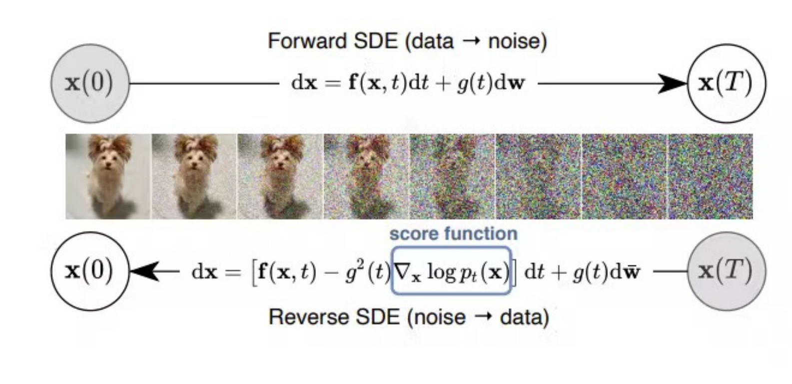
MotionMap



Generative Adversarial Networks



Variational Autoencoders

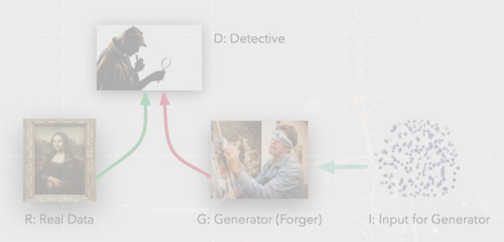


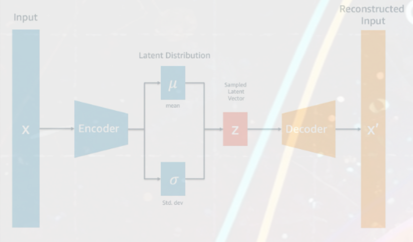
Diffusion

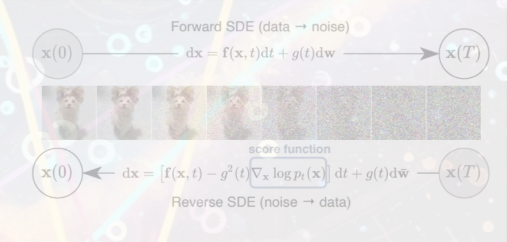
MotionMap
ToSS
TIC-TAC

Reyhaneh Hosseininejad* and Megh Shukla* et al., “*MotionMap: Representing Multimodality in Human Pose Forecasting*”. Under Review

MotionMap

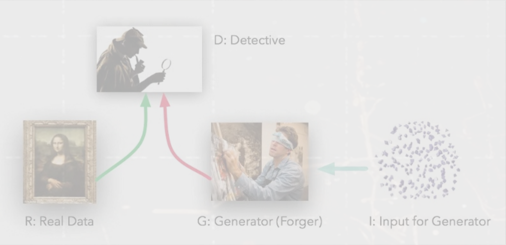


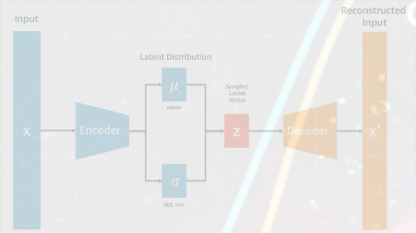


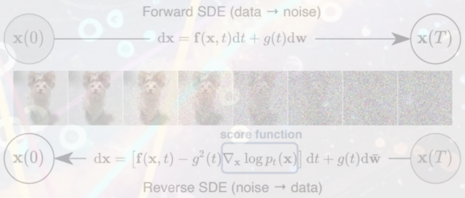


Randomness to predict multimodality

MotionMap



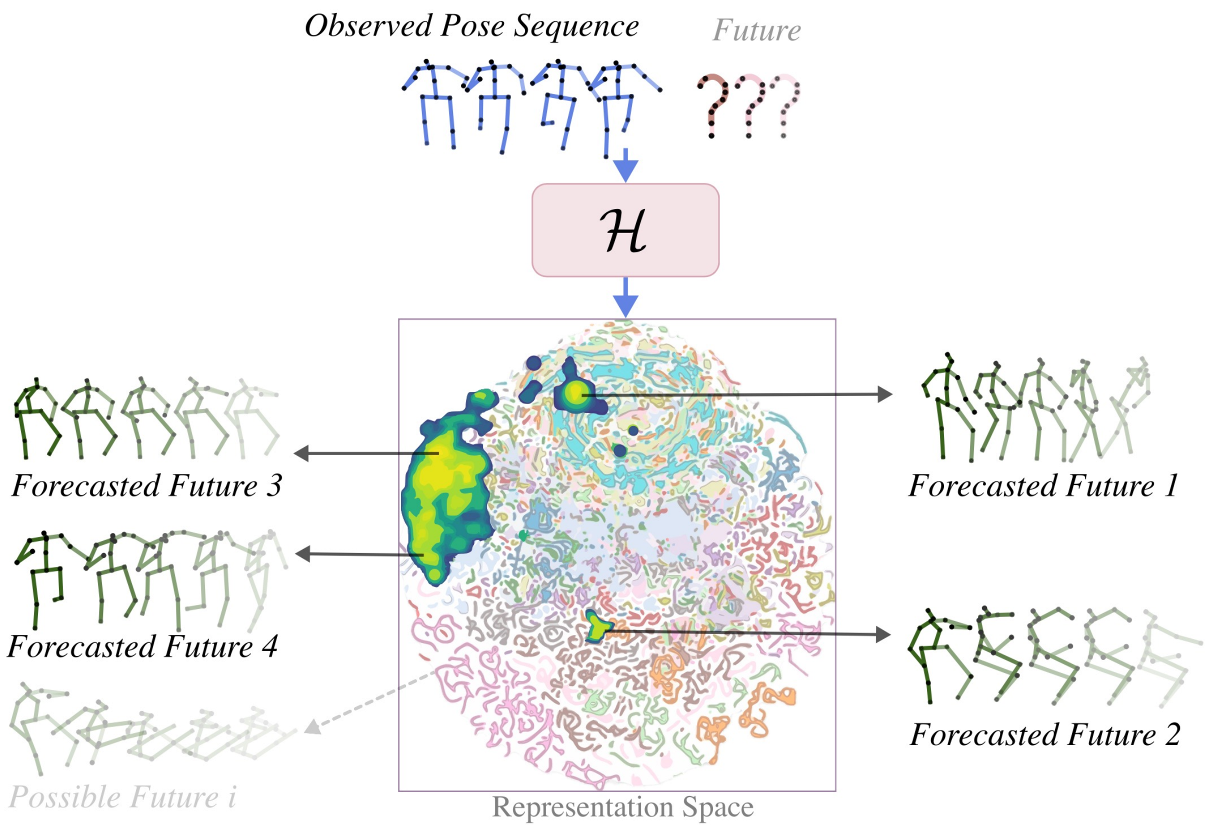




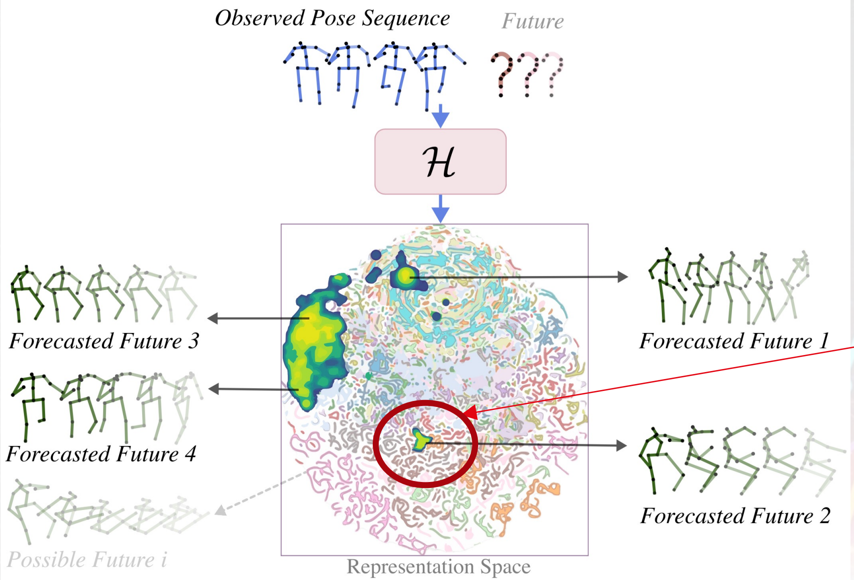
Randomness to predict multimodality

MotionMap

Multimodal Heteroscedastic Regression



MotionMap

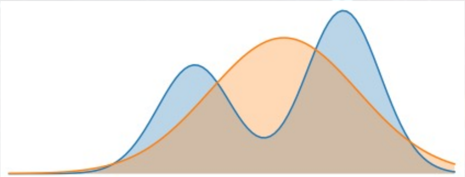


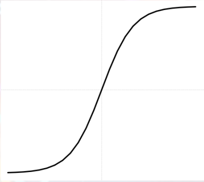
Variable number of modes
Some observations could lead to more actions

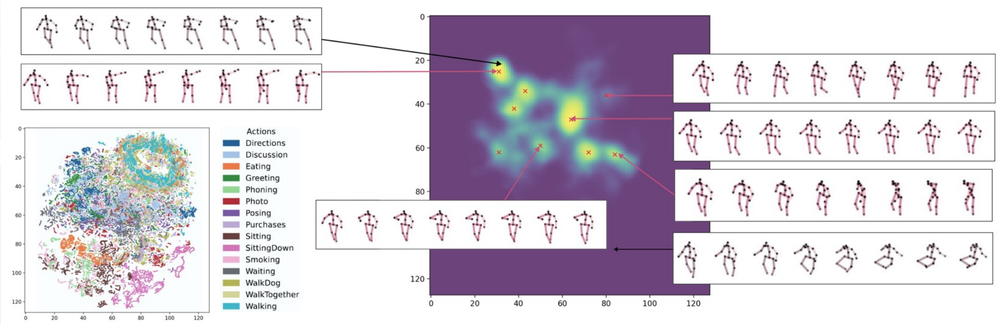
Confidence per modes
Which are the likeliest futures?

Explicitly encodes rare modes
Safety critical applications

Controllability
Which modes to generate?







Homoscedasticity v/s Heteroscedasticity
Homoscedasticity == Same uncertainty for all
Problem: Averages over all future modes

MotionMap

Quantitative

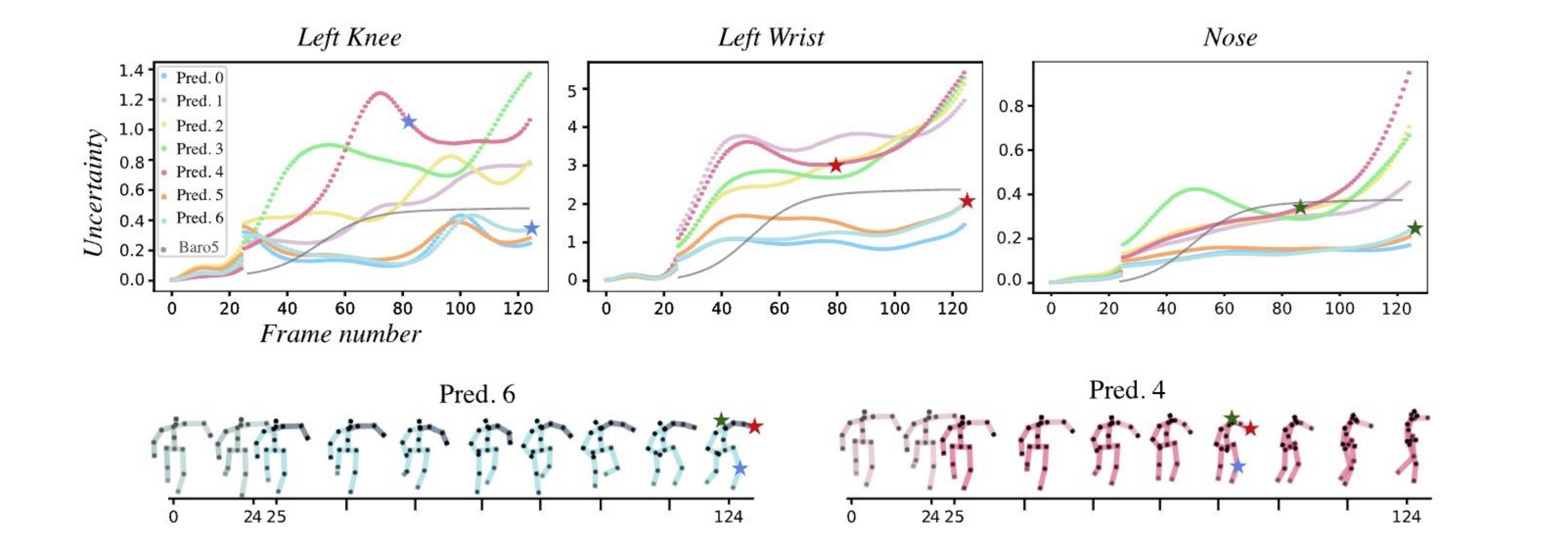
Method	Diversity (↑)	ADE (↓)	FDE (↓)	MMADE (↓)	MMFDE (↓)
Zero-Velocity	00.00	0.597	0.884	0.617	0.879
TPK [1]	06.53	0.534	0.691	0.559	0.675
DLow [2]	11.77	0.445	0.730	0.576	0.715
GSPS [3]	14.97	0.512	0.684	0.550	0.665
DivSamp [4]	15.73	0.480	0.685	0.542	0.671
BeLFusion [5]	07.11	0.441	0.597	0.491	0.586
CoMusion [6]	07.32	**0.426**	0.613	0.531	0.623
STARS [7]	**15.85**	0.508	0.697	0.551	0.686
TCD [8]	05.97	0.454	0.642	0.560	0.667
MotionMap	07.97	0.472	**0.594**	**0.464**	**0.529**

MotionMap

Qualitative - Ranking

MotionMap

Qualitative - Uncertainty

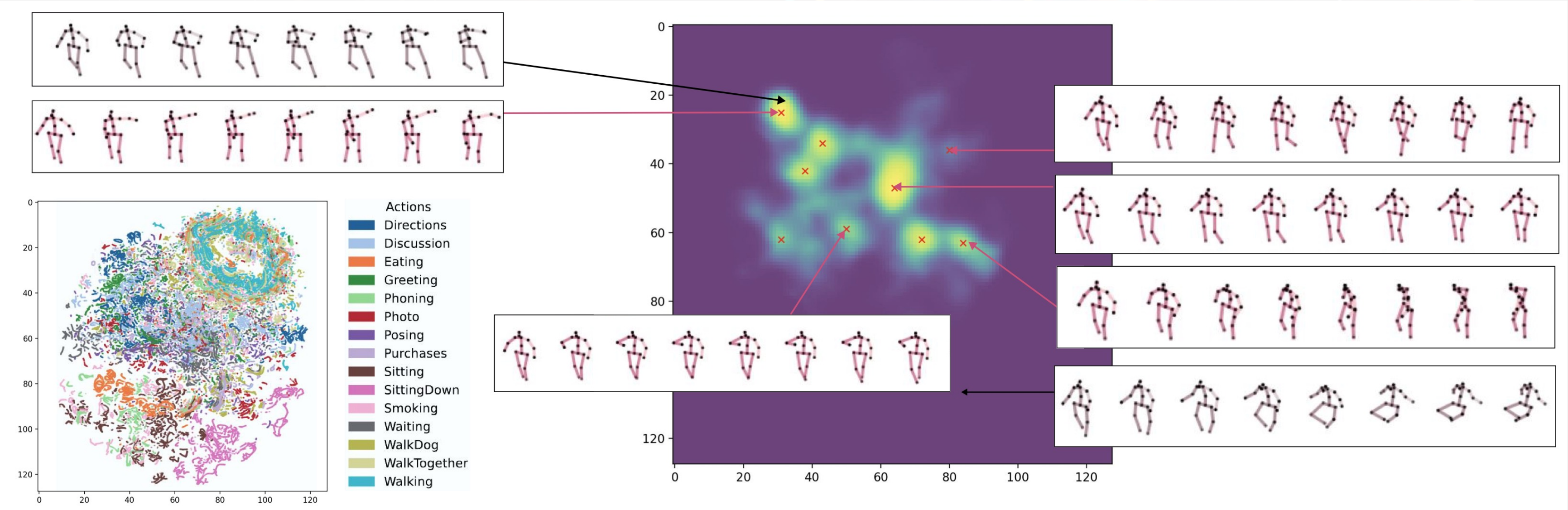


EPFL

Megh Shukla

MotionMap

Qualitative - Controllability



Deep Heteroscedastic Regression

Methods and Applications

TIC-TAC: A Framework for Improved Covariance Estimation in Deep Heteroscedastic Regression

Towards Self-Supervised Covariance Estimation in Deep Heteroscedastic Regression

MotionMap: Representing Multimodality in Human Pose Forecasting

deep-regression.github.io

ICML
International Conference On Machine Learning

ICLR
International Conference On Learning Representations

arXiv

Grazie

Gracias

謝謝

شكرًا لك

متشکرم

Thank you

Ευχαριστώ

धन्यवाद

ధన్యవాదాలు

Danke

Merci