Learning preferences with Gaussian processes: an application to bicycle route selection

Dario Azzimonti

Istituto Dalle Molle di studi sull'intelligenza artificiale
USI – SUPSI

9th June 2026

Who

The works presented here are the results of collaborations with

- Alessio Benavoli, Trinity College, Dublin, Ireland.

- **Thaisa De Torre**, SUPSI, Lugano, Switzerland.

The following paper provides an extensive tutorial on the subject:

Benavoli, A., & Azzimonti, D. 2026. A tutorial on learning from preferences and choices with Gaussian processes. *Foundations and Trends in Machine Learning*, Vol. 19 No. 1 pp. 1–120.

All algorithms are implemented in Python and available at

prefGP: A Python package for preference and choice learning with Gaussian processes.

Example: a bicycle route recommender

Task: building a route recommendation system for cyclists in Paris.

We have these objectives:

1. learn cyclists' preferences from their route choices
2. use this knowledge to suggest better routes in the future.

How can we model this problem? What types of data do we have?

We are going to use **preference data**.

Types of preference data

Each bicycle route is described by route-level features:

$$\mathbf{x} = \Bigg(\underbrace{\texttt{pct\_bike\_paths}}_{\in[0,1]}, \ \underbrace{\texttt{avg\_slope}}_{\in\mathbb{R}}, \ \underbrace{\texttt{total\_distance}}_{\in\mathbb{R}_+}, \ \underbrace{\texttt{pct\_lit}}_{\in[0,1]}, \ \ldots \Bigg) \in \mathcal{X}.$$

Three types of behavioural data:

1) **Object preferences:** cyclist picks the preferred route in an A/B comparison

2) **Label preferences:** per-cyclist ranking of route *profiles* {Ifpen, Safe, Shortest}

3) **Choice data:** cyclist marks *acceptable* routes from a set of alternatives

Goal: predict future route choices from whichever data are available.

What is a preference?

Given a finite set $\mathcal{X}$, a (strict) preference is a *binary relation* $\succ$ which is

- *asymmetric:* $\forall x, y \in \mathcal{X}$ if $x \succ y$ then not $y \succ x$
- *negatively transitive:* if $x \not\succ z$ and $z \not\succ y$ then $x \not\succ y$

Representation via utility functions

For any set $\mathcal{X}$ and preference relation $\succ$ on $\mathcal{X}$, the *utility* function $u : \mathcal{X} \to \mathbb{R}$ represents $\succ$ if

$$x \succ y \quad \textit{iff} \quad u(x) > u(y)$$

This representation is *not unique*: for any increasing function g

$$x \succ y \;\; \textit{iff} \;\; u(x) > u(y) \;\; \textit{iff} \;\; g(u(x)) > g(u(y)).$$

Back to the bicycle routing example

Each bicycle route is described by route-level features:

$$\mathbf{x} = \left(\underbrace{\texttt{pct\_bike\_paths}}_{\in[0,1]}, \ \underbrace{\texttt{avg\_slope}}_{\in\mathbb{R}}, \ \underbrace{\texttt{total\_distance}}_{\in\mathbb{R}_+}, \ \ldots \right) \in \mathcal{X}.$$

Object preferences: cyclist picks the preferred route: $\mathbf{x}_i$ vs $\mathbf{x}_j$.

$$\mathbf{x}_i \succ \mathbf{x}_j$$

Latent function: $u(\mathbf{x})$, the cyclist's utility for a route with features $\mathbf{x}$.

Goal: Infer $u(\mathbf{x})$ from the data and use it to predict future route choices.

How to do inference on $u(x)$?

We need a model that:

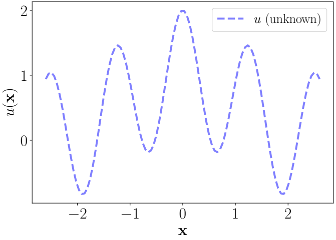
- can fit an unknown function u;
- can work with limited non-standard data (e.g., pairwise comparisons);
- provides a measure of uncertainty on the function values;

We will use Gaussian processes (GPs).

What is a Gaussian process model?

A Bayesian, non-parametric model to fit unknown functions.

Problem: fit unknown function.



Start: Prior on functions $p(f)$.

Observe the function at few values
$\mathcal{D} = (\mathbf{x}_i, y_i)_{i=1}^n, \quad \mathbf{x}_i \in D \subset \mathbb{R}^d$

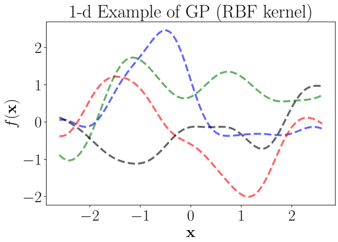
Likelihood: $p(\mathcal{D}|f) = \prod_{i=1}^n p(y_i|f(\mathbf{x}_i))$
$p(y_i|f(\mathbf{x}_i)) = N(f(\mathbf{x}_i), \sigma_n^2)$

What is a Gaussian process model?

A Bayesian, non-parametric model to fit unknown functions.

Problem: fit unknown function.

Start: Prior on functions $p(f)$.



Observe the function at few values
$\mathcal{D} = (\mathbf{x}_i, y_i)_{i=1}^n, \quad \mathbf{x}_i \in D \subset \mathbb{R}^d$

Likelihood: $p(\mathcal{D}|f) = \prod_{i=1}^n p(y_i|f(\mathbf{x}_i))$
$p(y_i|f(\mathbf{x}_i)) = N(f(\mathbf{x}_i), \sigma_n^2)$

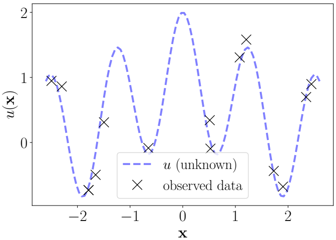
What is a Gaussian process model?

A Bayesian, non-parametric model to fit unknown functions.

Problem: fit unknown function.

Start: Prior on functions $p(f)$.

Observe the function at few values
$\mathcal{D} = (\mathbf{x}_i, y_i)_{i=1}^n, \quad \mathbf{x}_i \in D \subset \mathbb{R}^d$



Likelihood: $p(\mathcal{D}|f) = \prod_{i=1}^n p(y_i|f(\mathbf{x}_i))$
$p(y_i|f(\mathbf{x}_i)) = N(f(\mathbf{x}_i), \sigma_n^2)$

What is a Gaussian process model?

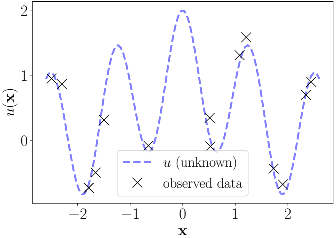
A Bayesian, non-parametric model to fit unknown functions.

Problem: fit unknown function.

Start: Prior on functions $p(f)$.

Observe the function at few values
$\mathcal{D} = (\mathbf{x}_i, y_i)_{i=1}^n, \quad \mathbf{x}_i \in D \subset \mathbb{R}^d$

Likelihood: $p(\mathcal{D}|f) = \prod_{i=1}^n p(y_i|f(\mathbf{x}_i))$
$p(y_i|f(\mathbf{x}_i)) = N(f(\mathbf{x}_i), \sigma_n^2)$



What is a Gaussian process model?

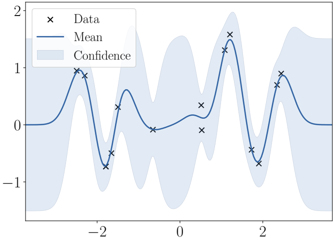
A Bayesian, non-parametric model to fit unknown functions.

Problem: fit unknown function.

Start: Prior on functions $p(f)$.

Observe the function at few values
$\mathcal{D} = (\mathbf{x}_i, y_i)_{i=1}^n, \quad \mathbf{x}_i \in D \subset \mathbb{R}^d$

Likelihood: $p(\mathcal{D}|f) = \prod_{i=1}^n p(y_i|f(\mathbf{x}_i))$
$p(y_i|f(\mathbf{x}_i)) = N(f(\mathbf{x}_i), \sigma_n^2)$



Update the prior — include data knowledge with Bayes theorem!

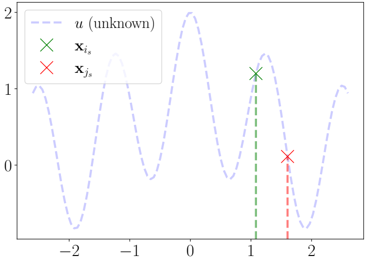
$$p(f|\mathbf{y}) = \frac{p(\mathbf{y}|f)p(f)}{p(\mathbf{y})}$$

A model for object preferences

Preference: $\mathbf{x}_{i_s} \succ \mathbf{x}_{j_s}$ for two items $\mathbf{x}_{i_s} \neq \mathbf{x}_{j_s} \in \mathcal{X}$;

Assumption: underlying hidden function u such that

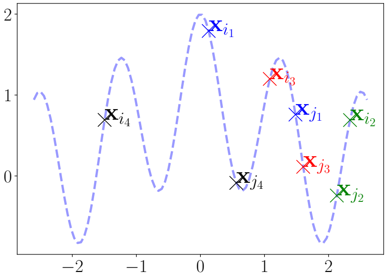
$$u(\mathbf{x}_{i_s}) \geq u(\mathbf{x}_{j_s}) \quad \text{if} \quad \mathbf{x}_{i_s} \succ \mathbf{x}_{j_s}$$



Consistent object preferences: GP model

Data: $m = 50$ pairwise preferences

$$\mathcal{D}_m = \{\mathbf{x}_{i_s} \succ \mathbf{x}_{j_s} : s = 1, \ldots, m\}$$



Prior: $u \sim GP(0, k)$;

Likelihood:

$$p(\mathcal{D}_m | u(X)) = \prod_{s=1}^{m} I_{\{u(\mathbf{x}_{i_s}) - u(\mathbf{x}_{j_s}) > 0\}}(u(X)) = I_{\{Wu(X) > 0\}}(u(X)).$$

Consistent object preferences: GP model

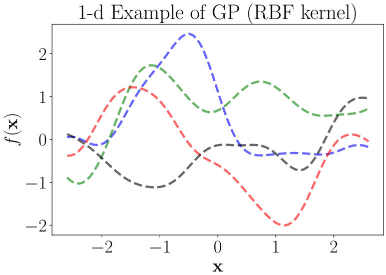
Data: $m = 50$ pairwise preferences

$$\mathcal{D}_m = \{\mathbf{x}_{i_s} \succ \mathbf{x}_{j_s} : s = 1, \ldots, m\}$$

Prior: $u \sim GP(0, k)$;

Likelihood:

$$p(\mathcal{D}_m | u(X)) = \prod_{s=1}^{m} I_{\{u(\mathbf{x}_{i_s}) - u(\mathbf{x}_{j_s}) > 0\}}(u(X)) = I_{\{Wu(X) > 0\}}(u(X)).$$



Consistent object preferences: GP model

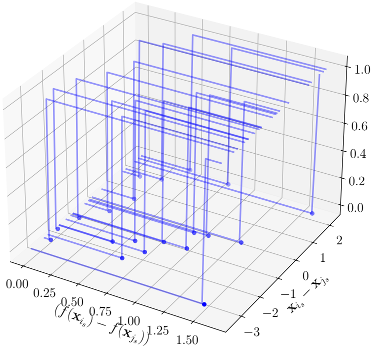
Data: $m = 50$ pairwise preferences

$$\mathcal{D}_m = \{\mathbf{x}_{i_s} \succ \mathbf{x}_{j_s} : s = 1, \ldots, m\}$$

Prior: $u \sim GP(0, k)$;

Likelihood:

$$p(\mathcal{D}_m | u(X)) = \prod_{s=1}^{m} I_{\{u(\mathbf{x}_{i_s}) - u(\mathbf{x}_{j_s}) > 0\}}(u(X)) = I_{\{Wu(X) > 0\}}(u(X)).$$



Consistent object preferences: GP model

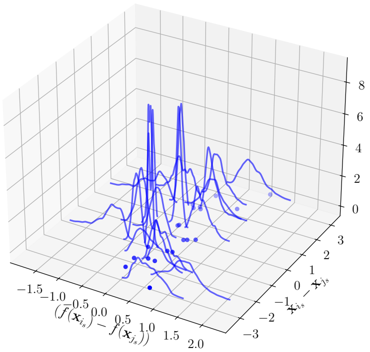
Data: $m = 50$ pairwise preferences

$$\mathcal{D}_m = \{\mathbf{x}_{i_s} \succ \mathbf{x}_{j_s} : s = 1, \ldots, m\}$$

Prior: $u \sim GP(0, k)$;

Likelihood:

$$p(\mathcal{D}_m | u(X)) = \prod_{s=1}^{m} I_{\{u(\mathbf{x}_{i_s}) - u(\mathbf{x}_{j_s}) > 0\}}(u(X)) = I_{\{Wu(X) > 0\}}(u(X)).$$



Posterior: $p(u(X)|\mathcal{D}_m)$ is analytical but inference requires sampling.

What if preferences contain errors?

Consistent object preferences: underlying hidden function u such that

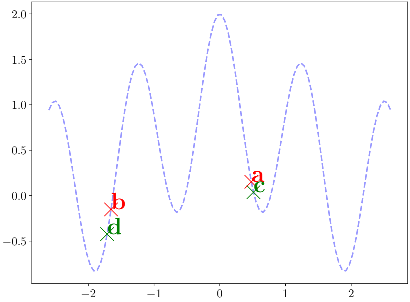
$$u(\mathbf{x}_{i_s}) \geq u(\mathbf{x}_{j_s}) \quad \text{if} \quad \mathbf{x}_{i_s} \succ \mathbf{x}_{j_s}$$

What if we have this data?

$$\mathbf{a} \succ \mathbf{b}$$

$$\mathbf{d} \succ \mathbf{c}$$

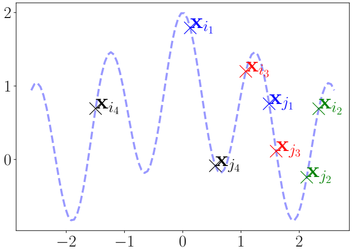
Contradictory preferences!



Erroneous preferences: GP model

Data: m pairwise preferences

$$\mathcal{D}_m = \{\mathbf{x}_{i_s} \succ \mathbf{x}_{j_s} : s = 1, \ldots, m\}$$



Prior: $u \sim GP(0, k)$;

Likelihood: $p(\mathcal{D}_m | u(X)) =$

$$\prod_{s=1}^{m} \Phi\left(u(\mathbf{x}_{i_s}) - u(\mathbf{x}_{j_s})\right) =$$

$$\Phi_m(Wu(X)).$$

Posterior: $p(u(X)|\mathcal{D}_m) =$ **?**

Chu, W. and Ghahramani, Z. 2005. Preference Learning with Gaussian Processes. In *Proceedings of the 22nd International Conference on Machine Learning* (ICML '05).

Erroneous preferences: GP model

Data: m pairwise preferences

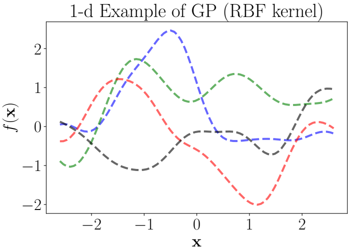
$$\mathcal{D}_m = \{\mathbf{x}_{i_s} \succ \mathbf{x}_{j_s} : s = 1, \ldots, m\}$$

Prior: $u \sim GP(0, k)$;

Likelihood: $p(\mathcal{D}_m | u(X)) =$

$$\prod_{s=1}^{m} \Phi\left(u(\mathbf{x}_{i_s}) - u(\mathbf{x}_{j_s})\right) =$$

$$\Phi_m(Wu(X)).$$



Posterior: $p(u(X)|\mathcal{D}_m) = \textbf{?}$

Chu, W. and Ghahramani, Z. 2005. Preference Learning with Gaussian Processes. In *Proceedings of the 22nd International Conference on Machine Learning* (ICML '05).

Erroneous preferences: GP model

Data: m pairwise preferences

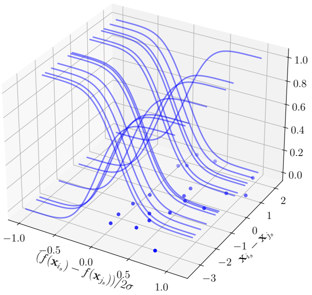
$$\mathcal{D}_m = \{\mathbf{x}_{i_s} \succ \mathbf{x}_{j_s} : s = 1, \ldots, m\}$$

Prior: $u \sim GP(0, k)$;

Likelihood: $p(\mathcal{D}_m | u(X)) =$

$$\prod_{s=1}^{m} \Phi\left(u(\mathbf{x}_{i_s}) - u(\mathbf{x}_{j_s})\right) =$$

$$\Phi_m(Wu(X)).$$



Posterior: $p(u(X)|\mathcal{D}_m) = ?$

Chu, W. and Ghahramani, Z. 2005. Preference Learning with Gaussian Processes. In *Proceedings of the 22nd International Conference on Machine Learning* (ICML '05).

Erroneous preferences: GP model

Data: m pairwise preferences

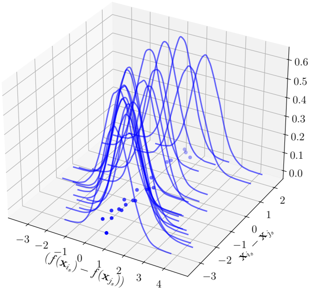
$$\mathcal{D}_m = \{\mathbf{x}_{i_s} \succ \mathbf{x}_{j_s} : s = 1, \ldots, m\}$$

Prior: $u \sim GP(0, k)$;

Likelihood: $p(\mathcal{D}_m | u(X)) =$

$$\prod_{s=1}^{m} \Phi\left(u(\mathbf{x}_{i_s}) - u(\mathbf{x}_{j_s})\right) =$$

$$\Phi_m(Wu(X)).$$



Posterior: $p(u(X)|\mathcal{D}_m)$ = **?**

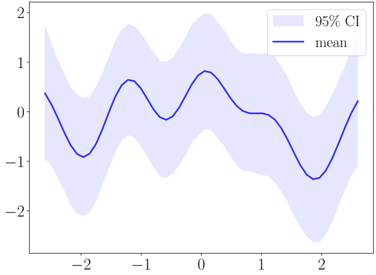
Chu, W. and Ghahramani, Z. 2005. Preference Learning with Gaussian Processes. In *Proceedings of the 22nd International Conference on Machine Learning* (ICML '05).

Erroneous preferences: analytical posterior

Data: $m = 50$ pairwise preferences

Prior: $u \sim GP(0, k)$;

Erroneous preference likelihood: $p(\mathcal{D}_m | u(X)) = \Phi_m(Wu(X))$



Posterior: $p(u(X)|\mathcal{D}_m) = SkewGP(\tilde{\xi}, \tilde{\Omega}, \tilde{\Delta}, \tilde{\gamma}, \tilde{\Gamma})$
A skew-Gaussian distribution with **closed-form parameters**.

Benavoli, A., D. A., Piga, D. 2021. A unified framework for closed-form nonparametric regression, classification, preference and mixed problems with Skew Gaussian Processes. *Machine Learning* 110, 3095–3133.

Object preferences — bicycle routes

Data: cyclists shown two routes for the same trip; they click the one they prefer.

Pair	Trip	Preferred route	Less preferred
1	Saint-François → Beaumarchais	Ifpen	Safe
2	Saint-François → Beaumarchais	Ifpen	Shortest
3	République → Notre-Dame	Safe	Shortest

Erroneous-preference likelihood:

$$P(\mathbf{x}_i \succ \mathbf{x}_j \mid u) = \Phi\left(\frac{u(\mathbf{x}_i) - u(\mathbf{x}_j)}{\sqrt{2}\,\sigma}\right)$$

Posterior: $p(u(\mathbf{x}) \mid \mathcal{D}_m) = SkewGP(\cdots)$

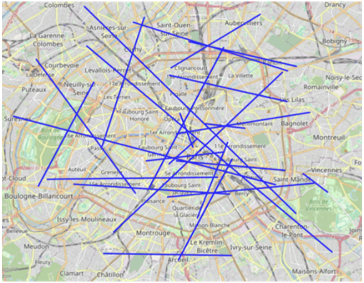
Inference: we can use the posterior to predict the utility of new routes.

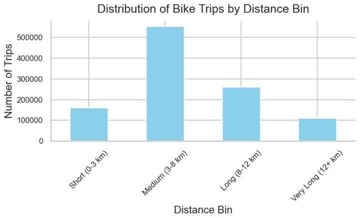
Building a simulated dataset

Focus: city of **Paris**.

Generation of realistic Origin-Destination (OD) trips

1. Velib Stations dataset: 1,473 stations in Paris;
2. Generate OD pairs by sampling from the station distribution.
3. BRouter API: compute routes for each OD pair (Ifpen, Safe, Shortest).





From routes to features

Route: described by sequence of segments, each with its own features (e.g., Distance, Elevation, bike path presence).

We need to extract route-level features to use in our model.

Feature extraction: for each route, we compute:

- the percentage of segment-level features, e.g., percentage of bike paths over whole route;
- total distance, energy, time.
- average slope.

Preferences: for each OD pair, we query the Ifpen and Safe profiles. We assume Ifpen $\succ$ Safe.

Experimental setup

Dataset: 1000 pairwise route comparisons for 1000 OD pairs in Paris.

Model: Erroneous-preference GP model with RBF kernel.

Route features: Tested different combinations, best results with

- 6 features summarizing the amenagement variable (bike and bus lanes, zones 30 and pedestrian areas)
- total distance, total time, average slope.

Evaluation: 80/20 train/test split; we compute the *accuracy* of the model in predicting the preferred route.

Results

Accuracy: 0.94 in predicting the preferred route on the test set (best model).

Feature importance from ARD of Gaussian processes:
the most important features are obtained from amenagement variable.

Contradictory preferences check:

- model tested on routes with same paths but different profiles (e.g., Ifpen and Safe);
- for those pairs, the model returns probabilities close to 0.5;
- the model correctly identifies the presence of contradictory preferences in the data.

Limitations and possible follow up for this work

Limitations:

- Simulated dataset, not real user data;
- Limited number of preference observations per OD pair.

Possible follow up:

- Generate more realistic route choice simulations (anonymized user data plus data augmentation?);
- Test the model on real user data;

Summary: GPs for object preferences

Coherent preferences model

- Requires coherent observations (no user errors);

Erroneous preferences model

- Accounts for possible user's errors;

Where to use this?

- learning rankings/preferences from user data;
- Bayesian optimization (use learned model to suggest better items).

prefGP: A Python package for preference and choice learning with Gaussian processes.

Beyond object preferences: label preference

- we have *cyclist profile* ($\mathbf{x}_k$, e.g. commuting/leisure, experience level, trip length)
- we aim to learn the ranking on a limited number of labels $\mathcal{Y} = \{\text{Ifpen}, \text{Safe}, \text{Shortest}\}$.

Cyclist $\mathbf{x}_k$	Observed pairwise label preference
Commuter, short trip, experienced	Ifpen $\succ$ Safe, Ifpen $\succ$ Shortest
Leisure, long trip, occasional	Safe $\succ$ Ifpen, Safe $\succ$ Shortest

On this problem:

- We can still use a GP model
- The posterior GP will give us the probability of each routing profile being preferred for a new cyclist profile $\mathbf{x}$.

Beyond object preferences: choice data

Setup: cyclist marks acceptable routes from a set of alternatives.

Route set A_k	Chosen subset $C(A_k)$
{Ifpen, Safe, Shortest, Custom}	{Ifpen, Safe}
{Safe, Shortest}	$\emptyset$ (cyclist declines all options)

On this problem:

- We can still use a GP model
- The empty choice is not problematic.
- The model provides uncertainty quantification.

What's next? Some open problems

- Sometimes the strength of preferences is available (e.g., how fast does a user decide), how to use it?
- How to learn from a mix of different types of data (e.g., a user decides between two items but also leaves reviews for other items)?

We can also use GPs to *optimize* preferences (Bayesian optimization with preferences). Open problems:

- What is the best setup to optimize preferences?
- Can we prove convergence of these optimization algorithms?

Benavoli, A., & Azzimonti, D. 2026. A tutorial on learning from preferences and choices with Gaussian processes. *Foundations and Trends in Machine Learning*, Vol. 19 No. 1 pp. 1–120.

Thanks for your attention!

References

Benavoli, A., & Azzimonti, D. 2026. A tutorial on learning from preferences and choices with Gaussian processes. *Foundations and Trends in Machine Learning*, Vol. 19 No. 1 pp. 1–120.

Rasmussen & Williams. 2006. Gaussian processes for machine learning. *MIT Press.*

Chu, W. and Ghahramani, Z. 2005. Preference Learning with Gaussian Processes. In *Proceedings of the 22nd International Conference on Machine Learning* (ICML '05) 137-144.

De Torre, T. 2025. Application of preference modelling to bicycle routing. *Bachelor thesis, SUPSI*.

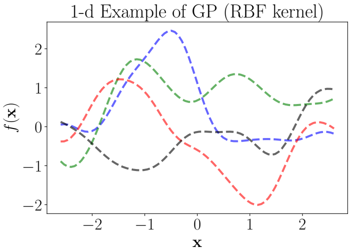
Benavoli, A., Azzimonti, D., and Piga, D. 2021. A unified framework for closed-form nonparametric regression, classification, preference and mixed problems with Skew Gaussian Processes. *Machine Learning* 110, 3095–3133.

Benavoli, A., Azzimonti, D., and Piga, D. 2020. Skew Gaussian Processes for Classification. *Machine Learning*, 109(9):1877–1902.

What is a Gaussian process?

A GP is a stochastic process $(f(\mathbf{x}))_{\mathbf{x}\in\mathcal{X}} \sim GP(\mu(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$ defined by its mean (μ often $\mu(\mathbf{x}) \equiv 0$) and kernel ($k(\mathbf{x}, \mathbf{x}')$) functions.

- For a fixed $\mathbf{x}$,
  $f(\mathbf{x})$ is a Gaussian r.v.
- For a fixed ω,
  $f(\omega, \mathbf{x})$ is a function of $\mathbf{x}$.
- k controls the properties of f.



Realization of $f \sim GP(0, k)$, RBF kernel.

We use GPs to define a prior distribution over functions.

Rasmussen & Williams. 2006. Gaussian processes for machine learning. *MIT Press*.

Transportation mode preference

Objective: predict transportation mode for a subject on a journey.

Four transportation modes: air, bus, car and train

Information about the journey:

- `dist` the distance of the trip;
- `cost` the monetary cost;
- `ivt` in-vehicle-time;
- `ovt` out-of-vehicle time;
- `freq` frequency.

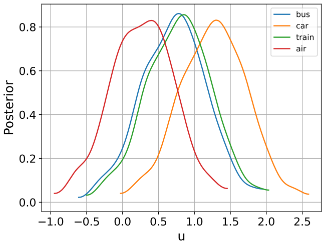
Preferences encoded as pairwise comparisons (679 cases):

$$\mathbf{x}_{car} \succ \mathbf{x}_{air}, \;\; \mathbf{x}_{car} \succ \mathbf{x}_{bus}, \;\; \mathbf{x}_{car} \succ \mathbf{x}_{train}$$

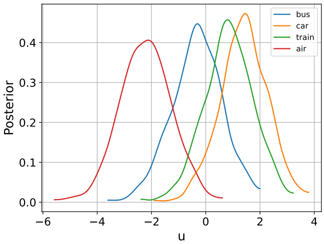
Transportation mode preference — results

alt	dist	cost	ivt	ovt	freq
bus	175	18.43	119	75	4
car	175	33.25	115	0	0
train	175	38.15	109	65	5
air	175	155.90	60	118	6

Test example: four transportation modes.



Linear kernel — accuracy 0.84



RBF kernel — accuracy 0.90