Comparison of Ordinal and Metric Gaussian Process Regression as Surrogate Models for CMA Evolution Strategy

Zbyněk Pitra[1,2,3], Lukáš Bajer[1,4], Jakub Repický[1,4], Martin Holeňa[1]

[1]Institute of Computer Science, Czech Academy of Sciences
[2]Faculty of Nuclear Sciences and Physical Engineering
[3]National Institute of Mental Health
[4]Faculty of Mathematics and Physics, Charles University

Prague, Czech Republic

GECCO 2017

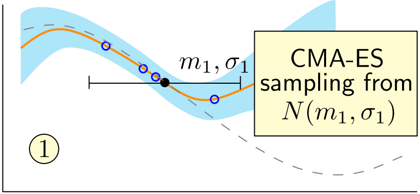
Contents

DTS-CMA-ES

Initialize: standard CMA-ES initialization with population doubled

while not terminate

1. CMA-ES sampling of population $\mathbf{x}_i \sim \mathcal{N}(\mathbf{m}, \sigma^2\mathbf{C})$, for $i = 1, \ldots, \lambda$

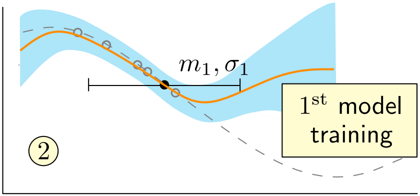


DTS-CMA-ES

Initialize: standard CMA-ES initialization with population doubled

while not terminate

1. CMA-ES sampling of population $\mathbf{x}_i \sim \mathcal{N}(\mathbf{m}, \sigma^2\mathbf{C})$, for $i = 1, \ldots, \lambda$
2. train the first model $f_{\mathcal{M}1}$ on the so-far original-evaluated points

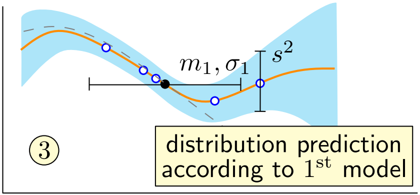


DTS-CMA-ES

Initialize: standard CMA-ES initialization with population doubled

while not terminate

1. CMA-ES sampling of population $\mathbf{x}_i \sim \mathcal{N}(\mathbf{m}, \sigma^2\mathbf{C})$, for $i = 1, \ldots, \lambda$
2. train the first model $f_{\mathcal{M}1}$ on the so-far original-evaluated points
3. get mean $\hat{\mu}_i$ and variance $\hat{s}_i^2$ of all $\mathbf{x}_i$ with the model $f_{\mathcal{M}1}$

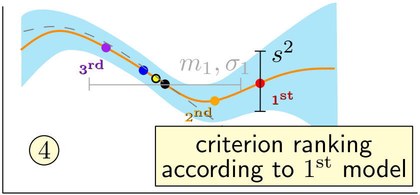


DTS-CMA-ES

Initialize: standard CMA-ES initialization with population doubled

while not terminate

1. CMA-ES sampling of population $\mathbf{x}_i \sim \mathcal{N}(\mathbf{m}, \sigma^2\mathbf{C})$, for $i = 1, \ldots, \lambda$
2. train the first model $f_{\mathcal{M}1}$ on the so-far original-evaluated points
3. get mean $\hat{\mu}_i$ and variance $\hat{s}_i^2$ of all $\mathbf{x}_i$ with the model $f_{\mathcal{M}1}$
4. select the most promising $\lceil \alpha\lambda \rceil$ points accord. to the model $f_{\mathcal{M}1}$

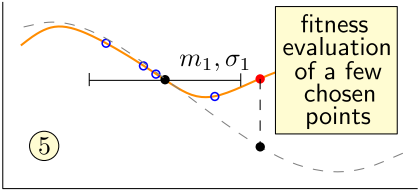


DTS-CMA-ES

Initialize: standard CMA-ES initialization with population doubled

while not terminate

1. CMA-ES sampling of population $\mathbf{x}_i \sim \mathcal{N}(\mathbf{m}, \sigma^2\mathbf{C})$, for $i = 1, \ldots, \lambda$
2. train the first model $f_{\mathcal{M}1}$ on the so-far original-evaluated points
3. get mean $\hat{\mu}_i$ and variance $\hat{s}_i^2$ of all $\mathbf{x}_i$ with the model $f_{\mathcal{M}1}$
4. select the most promising $\lceil \alpha\lambda \rceil$ points accord. to the model $f_{\mathcal{M}1}$
5. evaluate the chosen points with the original fitness f

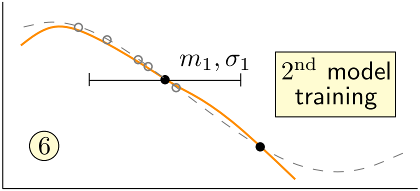


DTS-CMA-ES

Initialize: standard CMA-ES initialization with population doubled

while not terminate

1. CMA-ES sampling of population $\mathbf{x}_i \sim \mathcal{N}(\mathbf{m}, \sigma^2\mathbf{C})$, for $i = 1, \ldots, \lambda$
2. train the first model $f_{\mathcal{M}1}$ on the so-far original-evaluated points
3. get mean $\hat{\mu}_i$ and variance $\hat{s}_i^2$ of all $\mathbf{x}_i$ with the model $f_{\mathcal{M}1}$
4. select the most promising $\lceil \alpha\lambda \rceil$ points accord. to the model $f_{\mathcal{M}1}$
5. evaluate the chosen points with the original fitness f
6. re-train the second model $f_{\mathcal{M}2}$ with these new points

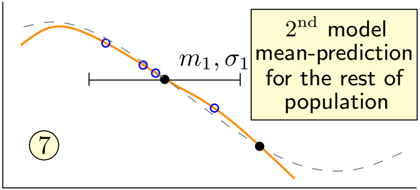


DTS-CMA-ES

Initialize: standard CMA-ES initialization with population doubled

while not terminate

1. CMA-ES sampling of population $\mathbf{x}_i \sim \mathcal{N}(\mathbf{m}, \sigma^2\mathbf{C})$, for $i = 1, \ldots, \lambda$
2. train the first model $f_{\mathcal{M}1}$ on the so-far original-evaluated points
3. get mean $\hat{\mu}_i$ and variance $\hat{s}_i^2$ of all $\mathbf{x}_i$ with the model $f_{\mathcal{M}1}$
4. select the most promising $\lceil \alpha\lambda \rceil$ points accord. to the model $f_{\mathcal{M}1}$
5. evaluate the chosen points with the original fitness f
6. re-train the second model $f_{\mathcal{M}2}$ with these new points
7. predict the fitness for the non-original-evaluated points with $f_{\mathcal{M}2}$

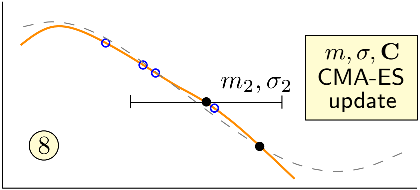


DTS-CMA-ES

Initialize: standard CMA-ES initialization with population doubled

while not terminate

1. CMA-ES sampling of population $\mathbf{x}_i \sim \mathcal{N}(\mathbf{m}, \sigma^2\mathbf{C})$, for $i = 1, \ldots, \lambda$
2. train the first model $f_{\mathcal{M}1}$ on the so-far original-evaluated points
3. get mean $\hat{\mu}_i$ and variance $\hat{s}_i^2$ of all $\mathbf{x}_i$ with the model $f_{\mathcal{M}1}$
4. select the most promising $\lceil \alpha\lambda \rceil$ points accord. to the model $f_{\mathcal{M}1}$
5. evaluate the chosen points with the original fitness f
6. re-train the second model $f_{\mathcal{M}2}$ with these new points
7. predict the fitness for the non-original-evaluated points with $f_{\mathcal{M}2}$
8. CMA-ES update of $\mathbf{m}$, σ, $\mathbf{C}$

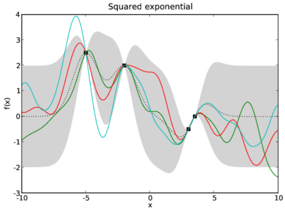


Gaussian Process

GP is a stochastic process, where any finite collection of random variables has a joint Gaussian distribution

$$f_{GP}(\mathbf{x}) \sim \mathrm{GP}(\mu(\mathbf{x}), k(\mathbf{x}_1, \mathbf{x}_2))$$

Defined by the mean function $\mu(\mathbf{x})$ (usually constant) and covariance function $k(\mathbf{x}_1, \mathbf{x}_2)$ and their (hyper)parameters

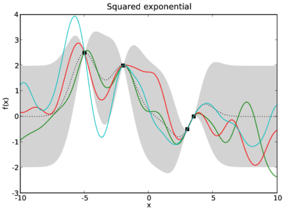


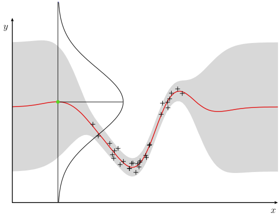
Gaussian Process

GP is a stochastic process, where any finite collection of random variables has a joint Gaussian distribution

$$f_{GP}(\mathbf{x}) \sim \mathrm{GP}(\mu(\mathbf{x}), k(\mathbf{x}_1, \mathbf{x}_2))$$

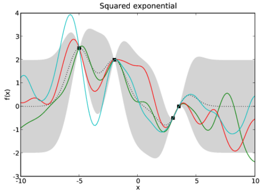
Defined by the mean function $\mu(\mathbf{x})$ (usually constant) and covariance function $k(\mathbf{x}_1, \mathbf{x}_2)$ and their (hyper)parameters





GP can express **uncertainty** of the prediction in a new point $\mathbf{x}$: it gives a probability distribution of the output value

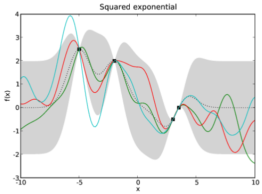
Gaussian Process



- given a set of N training points $\mathbf{X}_N = (\mathbf{x}_1 \ldots \mathbf{x}_N)$, $\mathbf{x}_i \in \mathbb{R}^d$, and corresponding measured values $\mathbf{y}_N = (y_1, \ldots, y_N)^\top$ of a function f being approximated

$$y_i = f(\mathbf{x}_i), \quad i = 1, \ldots, N$$

Gaussian Process



- given a set of N training points $\mathbf{X}_N = (\mathbf{x}_1 \dots \mathbf{x}_N)$, $\mathbf{x}_i \in \mathbb{R}^d$, and corresponding measured values $\mathbf{y}_N = (y_1, \dots, y_N)^\top$ of a function f being approximated

$$y_i = f(\mathbf{x}_i), \quad i = 1, \dots, N$$

GP considers vector of these function values as a sample from N-variate Gaussian distribution

$$\mathbf{y}_N \sim \mathcal{N}(\mathbf{0}, \mathbf{C}_N)$$

Gaussian Process prediction

When considering a new point $(\mathbf{x}^*, y^*)$, the prob. density of its f-values is 1D Gaussian

$$p(y^* \mid \mathbf{X}_N, \mathbf{x}^*, \mathbf{y}_N) \sim \mathcal{N}(\hat{\mu}_{N+1}, \hat{s}^2_{N+1})$$

Gaussian Process prediction

When considering a new point $(\mathbf{x}^*, y^*)$, the prob. density of its f-values is 1D Gaussian

$$p(y^* \,|\, \mathbf{X}_N, \mathbf{x}^*, \mathbf{y}_N) \;\sim\; \mathcal{N}(\hat{\mu}_{N+1}, \hat{s}^2{}_{N+1})$$

with the mean and variance given by

$$\begin{aligned} \hat{\mu}_{N+1} &= \mathbf{k}^\top \mathbf{C}_N{}^{-1} \mathbf{y}_N, \\ s^2{}_{N+1} &= \kappa - \mathbf{k}^\top \mathbf{C}_N{}^{-1} \mathbf{k} \end{aligned}$$

where

- $\mathbf{C}_N$ is GP covariance matrix – matrix of covariance function's values $k(\mathbf{x}_i, \mathbf{x}_j)$ for each pair $\mathbf{x}_i$, $\mathbf{x}_j$
- $\mathbf{k}$ is vector of covariance function's values $k(\mathbf{x}^*, \mathbf{x}_i)$ between the new point $\mathbf{x}^*$ and $\mathbf{x}_i \in \mathbf{X}_N$
- κ is the variance of the new point itself $k(\mathbf{x}^*, \mathbf{x}^*)$

Ordinal Gaussian Processes

Ordinal GP = Gaussian process $f_{GP}(\mathbf{x}) \sim \mathrm{GP}(\mu(\mathbf{x}), k(\mathbf{x}_1, \mathbf{x}_2))$

- trained on ordinal values $0, 1, \ldots, r$ instead of original f-values (including the following transformation)
- linearly mapped via set of additional parameters $\alpha_0, \alpha, b_1, \ldots, b_{r-1}$ onto the space of ordinal values $0, 1, \ldots, r$ as

$$f_{ORD}(\mathbf{x}) = \alpha_0 - \alpha f_{GP}(\mathbf{x})$$

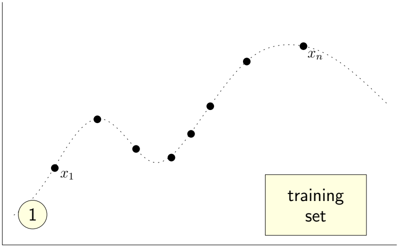
where $-\infty = b_0 < b_1 < \cdots < b_{r-1} < b_r = \infty$.

Ordinal Gaussian Processes

Training

1. $(\mathbf{x}_i, y_i)_{i=1}^{N} \leftarrow \mathcal{A}$ *{load data from archive}*

$\mathcal{A}$ – original data archive

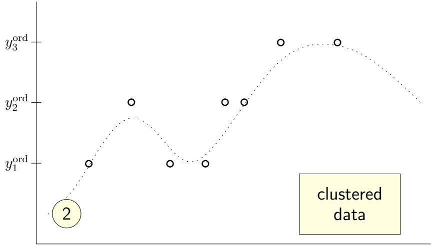


Ordinal Gaussian Processes

Training

1. $(\mathbf{x}_i, y_i)_{i=1}^{N} \leftarrow \mathcal{A}$ *{load data from archive}*
2. $\{y_i^{\text{ord}}\}_{i=1}^{N} \leftarrow \text{cluster}(\{y_i\}_{i=1}^{N}, r)$

$\mathcal{A}$ – original data archive
r – number of cluster levels

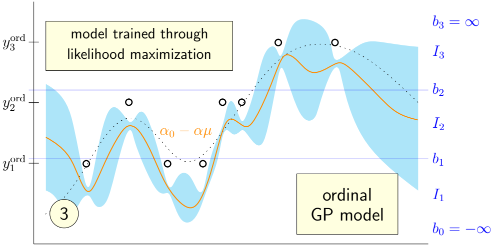


Ordinal Gaussian Processes

Training

1. $(\mathbf{x}_i, y_i)_{i=1}^{N} \leftarrow \mathcal{A}$ *{load data from archive}*
2. $\{y_i^{\text{ord}}\}_{i=1}^{N} \leftarrow \text{cluster}(\{y_i\}_{i=1}^{N}, r)$
3. $(\alpha, \{\beta_j\}_{j=1}^{r-1}, \boldsymbol{\theta})^* \leftarrow \underset{\alpha, \{\beta_j\}_{j=1}^{r-1}, \boldsymbol{\theta}}{\arg\max} \log \hat{\mathcal{L}}(\{y_i^{\text{ord}}\}_{i=1}^{N} | \{\mathbf{x}_i\}_{i=1}^{N}, \alpha, \{\beta_j\}_{j=1}^{r-1}, \boldsymbol{\theta})$

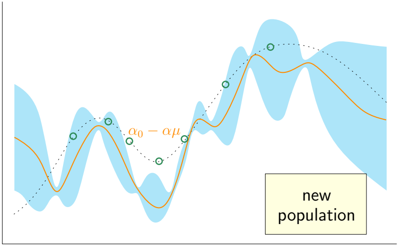
$\mathcal{A}$ – original data archive
r – number of cluster levels
α, α_0 – linear mapping parameters
$\beta_i = \alpha_0 + b_i$
$\boldsymbol{\theta}$ – latent GP hyperparameters



Ordinal Gaussian Processes

Prediction

$\{\mathbf{x}_i\}_{i=1}^{\lambda}$ – population to predict

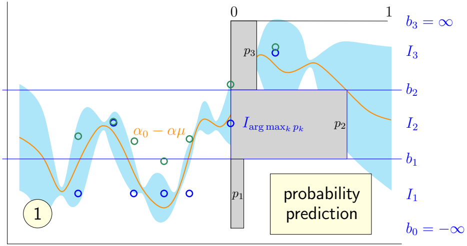


Ordinal Gaussian Processes

Prediction

1. $p_{i,k} \leftarrow P(f(\mathbf{x}_i) \in I_k | \mathbf{x}_i, \alpha, \{\beta_j\}_{j=1}^{r-1}, \boldsymbol{\theta}) \qquad \forall k = 1, \ldots, r, \forall i = 1, \ldots, \lambda$

$\{\mathbf{x}_i\}_{i=1}^{\lambda}$ – population to predict
r – number of cluster levels
α, α_0 – linear mapping parameters
$\beta_i = \alpha_0 + b_i$
$\boldsymbol{\theta}$ – latent GP hyperparameters

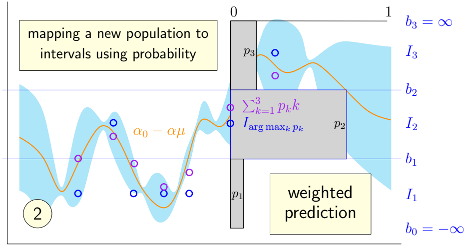


Ordinal Gaussian Processes

Prediction

1. $p_{i,k} \leftarrow P(f(\mathbf{x}_i) \in I_k | \mathbf{x}_i, \alpha, \{\beta_j\}_{j=1}^{r-1}, \boldsymbol{\theta})$ $\quad \forall k = 1, \ldots, r, \forall i = 1, \ldots, \lambda$
2. $q_i \leftarrow \sum_{k=1}^{r} p_{i,k} k$ $\quad \forall i = 1, \ldots, \lambda$

$\{\mathbf{x}_i\}_{i=1}^{\lambda}$ – population to predict
r – number of cluster levels
α, α_0 – linear mapping parameters
$\beta_i = \alpha_0 + b_i$
$\boldsymbol{\theta}$ – latent GP hyperparameters

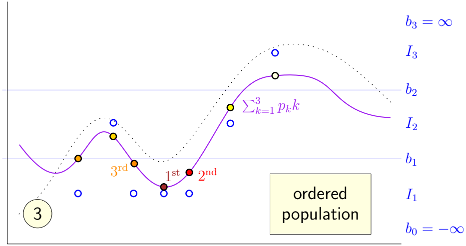


Ordinal Gaussian Processes

Prediction

1. $p_{i,k} \leftarrow P(f(\mathbf{x}_i) \in I_k | \mathbf{x}_i, \alpha, \{\beta_j\}_{j=1}^{r-1}, \boldsymbol{\theta})$ $\quad \forall k = 1, \ldots, r, \forall i = 1, \ldots, \lambda$
2. $q_i \leftarrow \sum_{k=1}^{r} p_{i,k} k$ $\quad \forall i = 1, \ldots, \lambda$
3. $\{\mathbf{x}_{i:\lambda}\}_{i=1}^{\lambda} \leftarrow$ order $\{\mathbf{x}_i\}_{i=1}^{\lambda}$ according to $q_{1:\lambda} \leq q_{2:\lambda} \leq \cdots \leq q_{\lambda:\lambda}$

$\{\mathbf{x}_i\}_{i=1}^{\lambda}$ – population to predict
r – number of cluster levels
α, α_0 – linear mapping parameters
$\beta_i = \alpha_0 + b_i$
$\boldsymbol{\theta}$ – latent GP hyperparameters



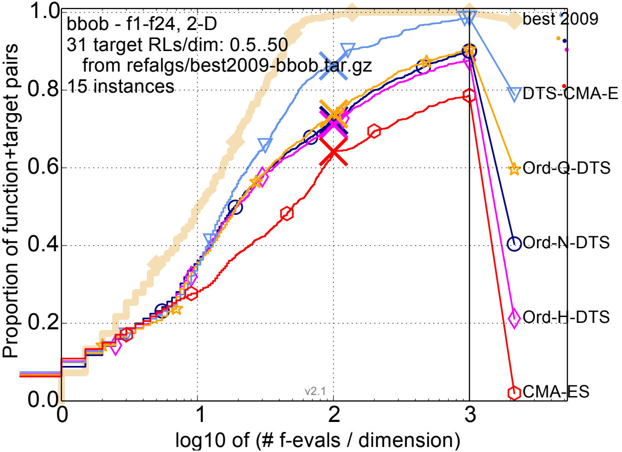
Experimental settings

- Noiseless part of the BBOB
- 100 FE/D budget
- Algorithms
  - CMA-ES
  - DTS-CMA-ES
  - Ord-N-DTS – **no** clustering
  - Ord-Q-DTS – **quantile**-based clustering
  - Ord-H-DTS – **aglomerative hierarchical** clustering

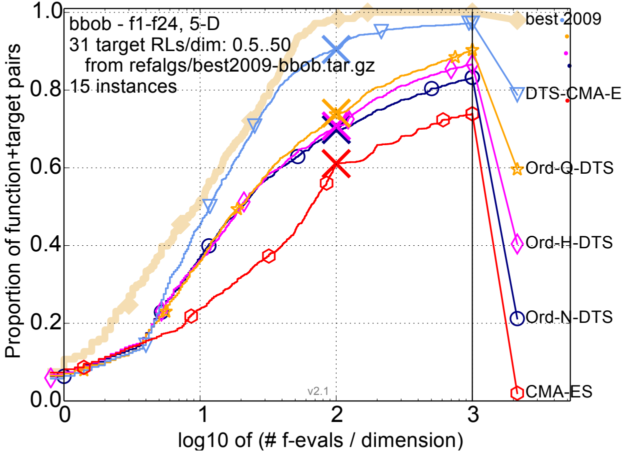
Experimental settings

- Noiseless part of the BBOB
- 100 FE/D budget
- Algorithms
  - CMA-ES
  - DTS-CMA-ES
  - Ord-N-DTS – **no** clustering
  - Ord-Q-DTS – **quantile**-based clustering
  - Ord-H-DTS – **aglomerative hierarchical** clustering
- Ordinal settings
  - λ ordinal levels
  - Matérn GP kernel

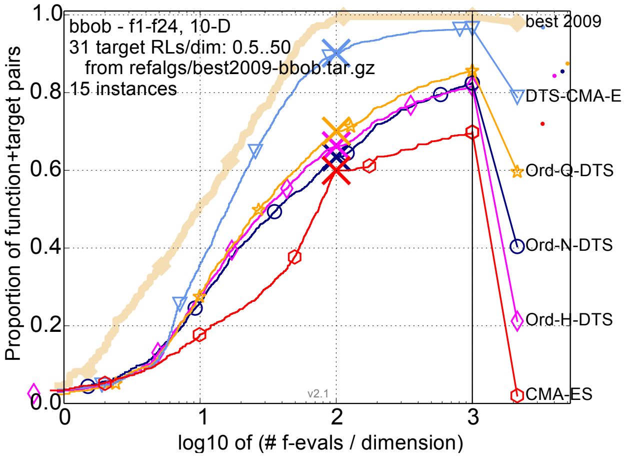
Experimental results on BBOB (2 D)



Experimental results on BBOB (5 D)

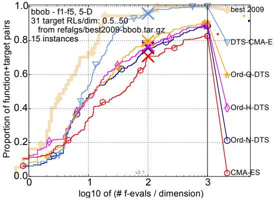


Experimental results on BBOB (10 D)

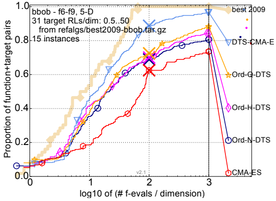


ECDF results on the whole BBOB (5 D)

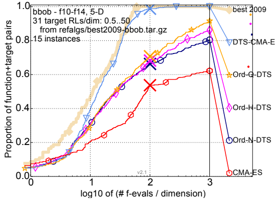
separable



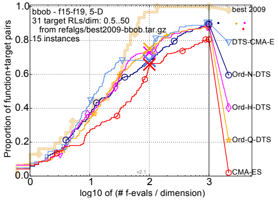
moderate



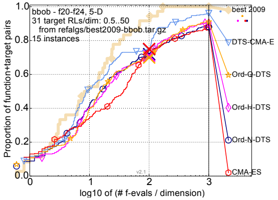
ill-conditional



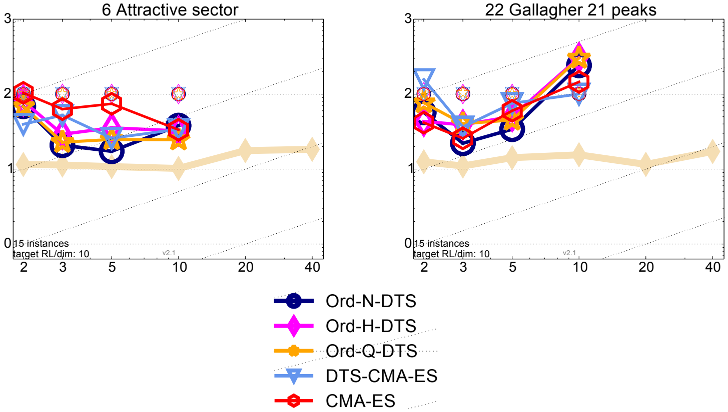
multi-modal



weakly structured multi-modal



Results on f6 and f22



Conclusions

- Effect of different clustering methods not crucial
- Performance of the ordinal GP models is considerably lower than the standard GP models with few exceptions (e. g., *attractive sector* f_6)
- Further investigation:
  - Adaptive switch between metric and ordinal models

Thank you!

z.pitra@gmail.com
bajeluk@gmail.com
j.repicky@gmail.com
martin@cs.cas.cz