

Gaussian Process Introduction

Lukáš Bajer^{1,2}

¹Faculty of Mathematics and Physics, Charles University,

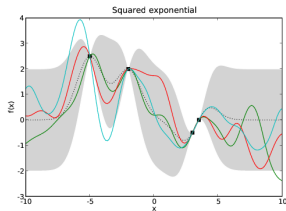
²Institute of Computer Science, Czech Academy of Sciences, and

Prague, Czech Republic

May 2014

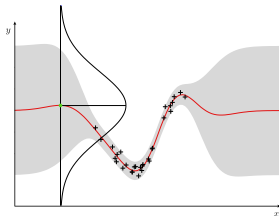
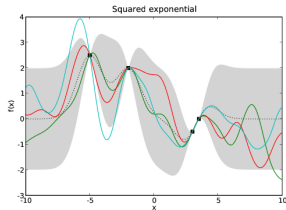
Gaussian Process

GP is a stochastic approximation method based on Gaussian distributions



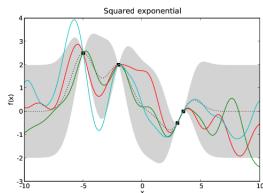
Gaussian Process

GP is a stochastic approximation method based on Gaussian distributions



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

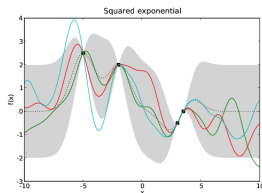
Gaussian Process



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

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

Gaussian Process



- given a set of N training points $\mathbf{X}_N = (\mathbf{x}_1 \dots \mathbf{x}_N)^\top$, $\mathbf{x}_i \in \mathbb{R}^d$, and 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

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

Gaussian Process

$\mathbf{X} = (\mathbf{x}_1 \dots \mathbf{x}_N)^\top$, $\mathbf{x}_i \in \mathbb{R}^d$ N training data points
 $\mathbf{y} = (y_1, \dots, y_N)^\top$, $y_i = f(\mathbf{x}_i)$ measured function values

$\mathbf{y} \in \mathbb{R}^N$ considered to be a realisation of a N -dimensional **Gaussian distribution** with a covariance matrix \mathbf{C}_N and zero mean

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

Gaussian Process

$\mathbf{X} = (\mathbf{x}_1 \dots \mathbf{x}_N)^\top$, $\mathbf{x}_i \in \mathbb{R}^d$ N training data points
 $\mathbf{y} = (y_1, \dots, y_N)^\top$, $y_i = f(\mathbf{x}_i)$ measured function values

$\mathbf{y} \in \mathbb{R}^N$ considered to be a realisation of a N -dimensional
Gaussian distribution with a covariance matrix \mathbf{C}_N and zero
 mean

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

Covariance \mathbf{C}_N is determined by

- covariance function $cov(\mathbf{x}_i, \mathbf{x}_j)$ and its hyperparameters
- training data points \mathbf{X}_N

forming the density of the Gaussian $p(\mathbf{y}|\mathbf{X}_N)$

Gaussian Process covariance

covariance \mathbf{C}_N is given by

$$\mathbf{C}_N = \mathbf{K} + \sigma^2 \mathbf{I}$$

where \mathbf{K} is a matrix of **covariance function** values and σ^2 is the **signal noise**.

Gaussian Process covariance

covariance \mathbf{C}_N is given by

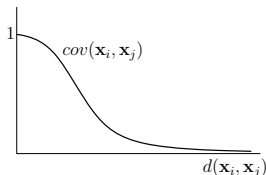
$$\mathbf{C}_N = \mathbf{K} + \sigma^2 \mathbf{I}$$

where \mathbf{K} is a matrix of **covariance function** values and σ^2 is the **signal noise**.

Covariance functions are defined on pairs from the input space

$$(\mathbf{K})_{ij} = \text{cov}(\mathbf{x}_i, \mathbf{x}_j), \quad \mathbf{x}_{i,j} \in \mathbb{R}^d$$

expressing the degree of correlations between two points' values; typically decreasing functions on two points distance



Gaussian Process covariance

The most frequent covariance function is *squared-exponential*

$$(\mathbf{K})_{ij} = \text{cov}^{\text{SE}}(\mathbf{x}_i, \mathbf{x}_j) = \theta \exp\left(\frac{-1}{2\ell^2} (\mathbf{x}_i - \mathbf{x}_j)^\top (\mathbf{x}_i - \mathbf{x}_j)\right)$$

with the parameters (usually fitted by MLE)

- θ – **signal variance** (scales the correlation)
- ℓ – **characteristic length scale**

Gaussian Process prediction

Making predictions

Prediction y^* in a new point \mathbf{x}^* is made by adding this new point to the matrix \mathbf{X}_N and vector \mathbf{y}_N .

Gaussian Process prediction

Making predictions

Prediction \mathbf{y}^* in a new point \mathbf{x}^* is made by adding this new point to the matrix \mathbf{X}_N and vector \mathbf{y}_N .

This gives an $(N + 1)$ -dimensional Gaussian with density

$$p(\mathbf{y}_{N+1} | \mathbf{X}_{N+1}) = \frac{1}{\sqrt{(2\pi)^{N+1} \det(\mathbf{C}_{N+1})}} \exp\left(-\frac{1}{2} \mathbf{y}_{N+1}^\top \mathbf{C}_{N+1}^{-1} \mathbf{y}_{N+1}\right)$$

Gaussian Process prediction

Making predictions

Prediction \mathbf{y}^* in a new point \mathbf{x}^* is made by adding this new point to the matrix \mathbf{X}_N and vector \mathbf{y}_N .

This gives an $(N + 1)$ -dimensional Gaussian with density

$$p(\mathbf{y}_{N+1} | \mathbf{X}_{N+1}) = \frac{1}{\sqrt{(2\pi)^{N+1} \det(\mathbf{C}_{N+1})}} \exp\left(-\frac{1}{2} \mathbf{y}_{N+1}^\top \mathbf{C}_{N+1}^{-1} \mathbf{y}_{N+1}\right)$$

where \mathbf{C}_{N+1} is the covariance matrix

$$\mathbf{C}_{N+1} = \begin{pmatrix} \mathbf{C}_N & \mathbf{k} \\ \mathbf{k}^\top & \kappa + \sigma \end{pmatrix}$$

which is \mathbf{C}_N extended with

- \mathbf{k} – covariances between \mathbf{x}^* and \mathbf{X}_N
- $\kappa + \sigma$ – variance of the new point itself (with added signal noise)

Gaussian Process prediction

Making predictions

Because \mathbf{y}_N is known and the inverse \mathbf{C}_{N+1}^{-1} can be expressed using inverse of the training covariance \mathbf{C}_N^{-1} ,

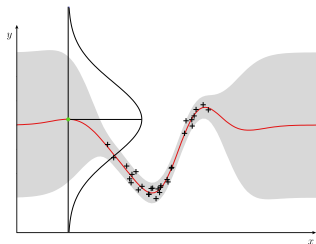
Gaussian Process prediction

Making predictions

Because \mathbf{y}_N is known and the inverse \mathbf{C}_{N+1}^{-1} can be expressed using inverse of the training covariance \mathbf{C}_N^{-1} ,

the density in a new point marginalize to **1D Gaussian** density

$$p(y^* | \mathbf{X}_{N+1}, \mathbf{y}_N) \propto \exp\left(-\frac{1}{2} \frac{(y^* - \hat{y}_{N+1})^2}{s_{y_{N+1}}^2}\right)$$



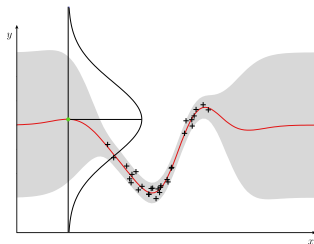
Gaussian Process prediction

Making predictions

Because \mathbf{y}_N is known and the inverse \mathbf{C}_{N+1}^{-1} can be expressed using inverse of the training covariance \mathbf{C}_N^{-1} ,

the density in a new point marginalize to **1D Gaussian** density

$$p(\mathbf{y}^* | \mathbf{X}_{N+1}, \mathbf{y}_N) \propto \exp\left(-\frac{1}{2} \frac{(\mathbf{y}^* - \hat{\mathbf{y}}_{N+1})^2}{s_{y_{N+1}}^2}\right)$$



with the mean and variance given by

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

Urychlení evoluční optimalizace pomocí gaussovských procesů

Zbyněk Pitra

Fakulta jaderná a fyzikálně inženýrská, ČVUT

2014

- Evoluční algoritmy
- Náhradní modelování
- GENACAT
- Gaussovské procesy jako náhradní model
- Testování

Optimalizace katalyzátorů pro specifickou reakci

- Prostory o velké dimenzi (až 50)

Optimalizace katalyzátorů pro specifickou reakci

- Prostory o velké dimenzi (až 50)
- Diskrétní a spojité proměnné

Optimalizace katalyzátorů pro specifickou reakci

- Prostory o velké dimenzi (až 50)
- Diskrétní a spojité proměnné
- Mnoho omezení

Evoluční a genetické algoritmy

- Chování organismů v umělém prostředí = kvalita řešení → fitness funkce
- Jedinec \sim přípustné řešení
- Populace = množina jedinců
- Výběr populace a následný vývoj
- Typy: Genetické algoritmy, Evoluční programování, ...

- Nultá populace
- Výběr několika jedinců
- Vznik další generace: křížení, mutace, reprodukce
- Ohodnocení nových jedinců
- Kontrola ukončovací podmínky
- Konec algoritmu

- Výhody:

- Výhody:
 - Odolnost vůči uvíznutí v lokálním optimu

- Výhody:
 - Odolnost vůči uvíznutí v lokálním optimu
 - Dobré výsledky u problémů s rozsáhlými množinami přípustných řešení

- Výhody:
 - Odolnost vůči uvíznutí v lokálním optimu
 - Dobré výsledky u problémů s rozsáhlými množinami přípustných řešení
- Nevýhody:

- Výhody:
 - Odolnost vůči uvíznutí v lokálním optimu
 - Dobré výsledky u problémů s rozsáhlými množinami přípustných řešení
- Nevýhody:
 - Problém s nalezením přesného optima

- Výhody:
 - Odolnost vůči uvíznutí v lokálním optimu
 - Dobré výsledky u problémů s rozsáhlými množinami přípustných řešení
- Nevýhody:
 - Problém s nalezením přesného optima
 - Velké množství vyhodnocování cílové funkce

Náhradní modelování

- Metoda využívající model místo skutečných dat
- Aproximace neznámé funkce $f : \Omega \mapsto \mathbb{C}^q$ z omezeného počtu dat $\{\mathbf{x}_1, \dots, \mathbf{x}_k\} \subset \Omega$, kde $\Omega \subset \mathbb{R}^d$
- Gaussové procesy, neuronové sítě, náhodné lesy, atd.

Náhradní modelování

Hledání nejlepší aproximace

$$\tilde{f}^* = \arg \min_{t \in T} \arg \min_{\theta \in \Theta} \Lambda(\varepsilon, \tilde{f}_{t,\theta}, D)$$

$\tilde{f}_{t,\theta}$ — aproximační funkce s parametrizací θ a typem modelu t

Λ — odhad kvality modelu

ε — chybová funkce

$D = \{(\mathbf{x}_1, f(\mathbf{x}_1)), \dots, (\mathbf{x}_k, f(\mathbf{x}_k))\}$

Θ – parametrický prostor \tilde{f}

T – množina typů modelů

Náhradní modelování

Genetické algoritmy a náhradní modelování

- Není třeba vždy vyhodnocovat skutečnou fitness funkci
- Řízení evoluce založené na
 - jedincích
 - generacích

GENACAT

System pro genetickou optimalizaci

- Leibnitzův institut pro katalýzu (LIKAT) v Německu
- Automatické generování uživatelem definovaných genetických algoritmů
- Náhradní modelování pomocí neuronových sítí a gaussovských procesů

Gaussovský proces je náhodný proces, kde libovolná konečná podmnožina náhodných veličin má sdružené gaussovské rozdělení.

- Hyperparametry:

Gaussovský proces je náhodný proces, kde libovolná konečná podmnožina náhodných veličin má sdružené gaussovské rozdělení.

- Hyperparametry:
 - Volné parametry kovarianční funkce gaussovského procesu

Gaussovský proces je náhodný proces, kde libovolná konečná podmnožina náhodných veličin má sdružené gaussovské rozdělení.

- Hyperparametry:
 - Volné parametry kovarianční funkce gaussovského procesu
 - Optimalizovány pomocí maximální věrohodnosti

Gaussovský proces je náhodný proces, kde libovolná konečná podmnožina náhodných veličin má sdružené gaussovské rozdělení.

- Hyperparametry:
 - Volné parametry kovarianční funkce gaussovského procesu
 - Optimalizovány pomocí maximální věrohodnosti
- Vlastnosti:

Gaussovský proces je náhodný proces, kde libovolná konečná podmnožina náhodných veličin má sdružené gaussovské rozdělení.

- Hyperparametry:
 - Volné parametry kovarianční funkce gaussovského procesu
 - Optimalizovány pomocí maximální věrohodnosti
- Vlastnosti:
 - Nepotřebují předdefinovanou strukturu

Gaussovský proces je náhodný proces, kde libovolná konečná podmnožina náhodných veličin má sdružené gaussovské rozdělení.

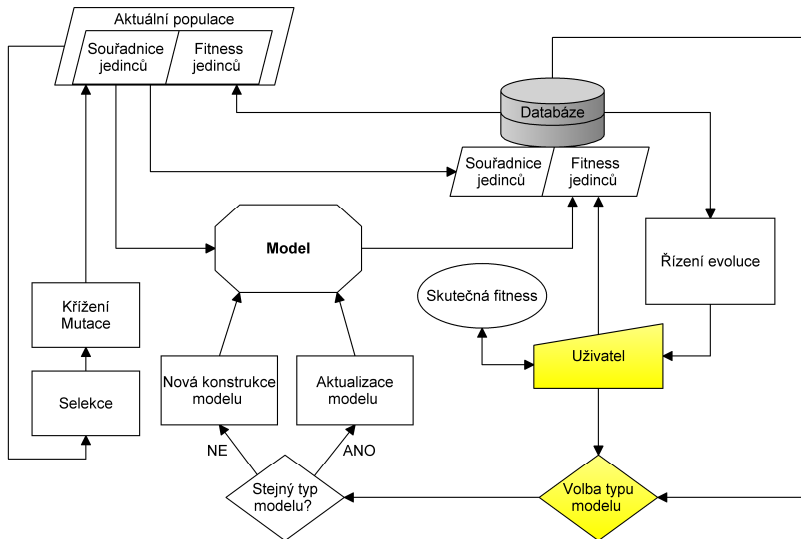
- Hyperparametry:
 - Volné parametry kovarianční funkce gaussovského procesu
 - Optimalizovány pomocí maximální věrohodnosti
- Vlastnosti:
 - Nepotřebují předdefinovanou strukturu
 - Aproximace libovolných funkcí

Gaussovský proces je náhodný proces, kde libovolná konečná podmnožina náhodných veličin má sdružené gaussovské rozdělení.

- Hyperparametry:
 - Volné parametry kovarianční funkce gaussovského procesu
 - Optimalizovány pomocí maximální věrohodnosti
- Vlastnosti:
 - Nepotřebují předdefinovanou strukturu
 - Aproximace libovolných funkcí
 - Nastavitelné kovarianční funkce s jejich hyperparametry

GENACAT

Hlavní schéma



Náhradní model s gaussovskými procesy

Genetický algoritmus s gaussovskými procesy

```
repeat until (UKONČOVACÍ PODMÍNKA je splněna) do  
  ActualPop := poslední generace jedinců z databáze;  
  Empirical := všichni jedinci ohodnocení skutečnou fitness;  
  Model := gplearnmodel(Empirical);  
if řízení založené na jedincích  
  then ZVYŠ počet jedinců pro přehodnocení modelem;  
for Gen := 1 to požadovaný počet generací  
  VYPOČÍTEJ počty jedinců ActualPop v přípustných mnohostěnech;  
  ChosenPol := mnohostěny vybrané diskrétní optimalizací;  
  NewPop := gpsimrun(ChosenPol);  
  ZAPIŠ NewPop do database;  
  ActualPop := NewPop;  
end  
  PŘEHODNOŤ některé jedince z poslední NewPop skutečnou fitness;  
end
```

Náhradní model s gaussovskými procesy

Spojité optimalizace - gpsimrun

Vstup: *ChosenPol* (vybrané mnohostěny), *Model* (natrénovaný model), *Ker* (zadaná jádra)

for mnohostěny *M* z *ChosenPol*

Ker(*M*):= VYHLEDEJ jádro z *Model* pro *M*;

if *ActualPop*(*M*) je neprázdná

then GENERUJ novou populaci *NewPop*(*M*) genetickými operacemi z *ActualPop*(*M*);

else GENERUJ *NewPop*(*M*) náhodně;

end

Výstup: nová populace *NewPop*

Náhradní model s gaussovskými procesy

Trénovací procedura - gplearnmodel

Vstup: *Empirical* (Empirical data), p (minimální velikost shluku), *Ker* (zadaná jádra)

$n := p * x / (x - 1);$

ROZDĚL *Empirical* do k shluků cl , aby velikost $(cl) \geq n$

for $i := 1$ **to** k

$Err(i, Ker) := \text{kernelError}(cl(i), Ker);$

$MainKer(i) := Ker(i)$, kde $Err(i, Ker)$ je minimální;

$Hyp(MainKer, i) := \text{argmax}_{\theta} (\text{věrohodnost}(cl(i)));$

end

Výstup: *MainKer* a *Hyp(MainKer)* pro každý shluk cl

Náhradní model s gaussovskými procesy

Procedura počítající chybu

Vstup: cl (shluk), x (násobnost křížové validace), Ker (zadaná jádra)

for Ker

PROVEĎ x -násobnou křížovou validaci cl , kde v každém foldu:

$Hyp(Ker) :=$ odhad hyperparametrů θ ;

$Hyp(Ker) := \operatorname{argmax}_{\theta}$ (věrohodnost(cl));

$Err(Ker) :=$ průměrná chyba z křížové validace;

end

Výstup: Err chyby pro shluk cl a všechna jádra z Ker

GENACAT

Nastavení gaussovských procesů

The screenshot shows a window titled "gpsetlearnmodel" with the following content:

Gaussian process model parameters setting:

Uniform setting for all clusters

Individual setting for each cluster

Cluster: 7

Cluster 7

Kernels

Available Kernels	Chosen Kernels
covLINard	covLIN
covSEard	covLINone
covPoly,1	covConst
covPoly,5	covSEisoU
covMaterniso,5	covPoly,2
covPPiso,2	covPoly,3
covPPiso,3	covPoly,4
covNoise	covMaterniso,1
covSEiso	covMaterniso,3
	covPPiso,0
	covPPiso,1

Buttons: Add, Remove, Add all, Remove all

Clusters size: 29

X-fold cross validation: 15

Number of hyperparameters: 10

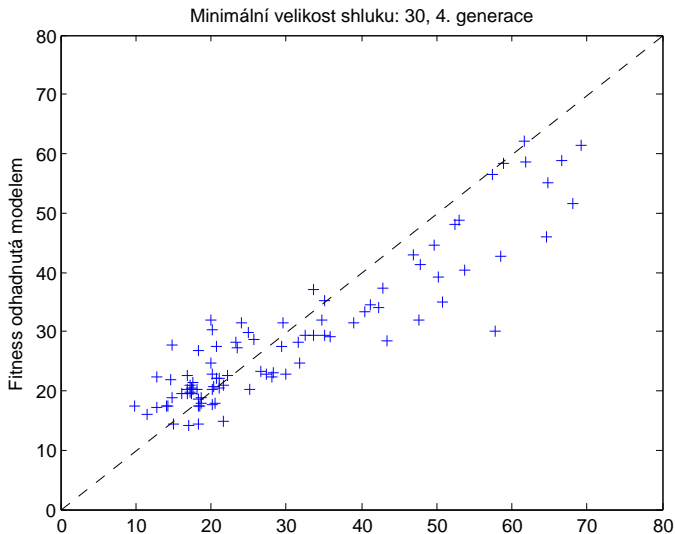
Show ten best kernels

Buttons: OK, Default, Quit

- Nejvhodnější katalyzátor při výrobě HCN
- 0. – 7. generace
- Zlepšování modelu v závislosti na rostoucí trénovací množině
- 3 modely - minimální počty jedinců ve shluku
 - 30 jedinců
 - 100 jedinců
 - bez shlukování

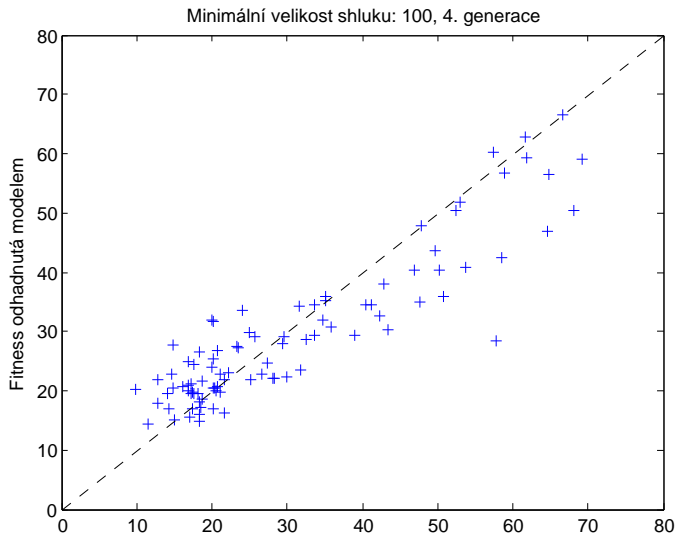
Testování na empirických datech

30 jedinců na 4. generaci



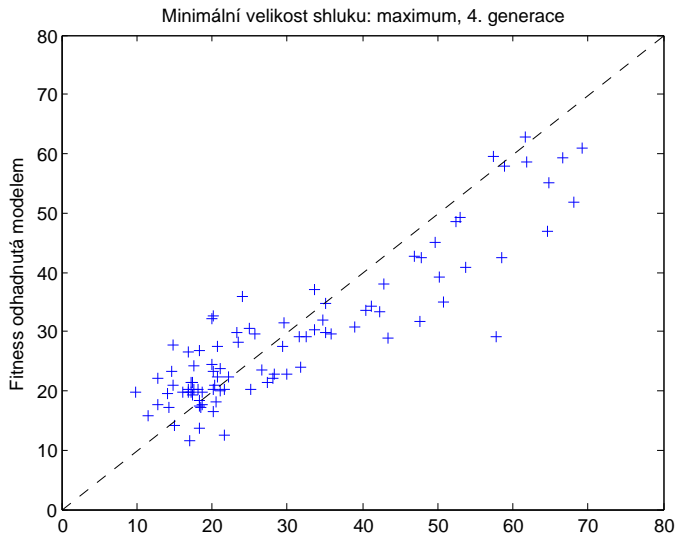
Testování na empirických datech

100 jedinců na 4. generaci



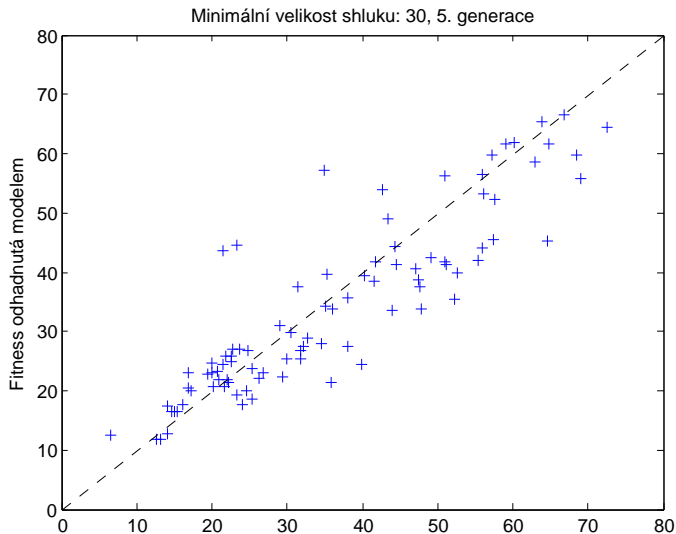
Testování na empirických datech

Bez shlukování na 4. generaci



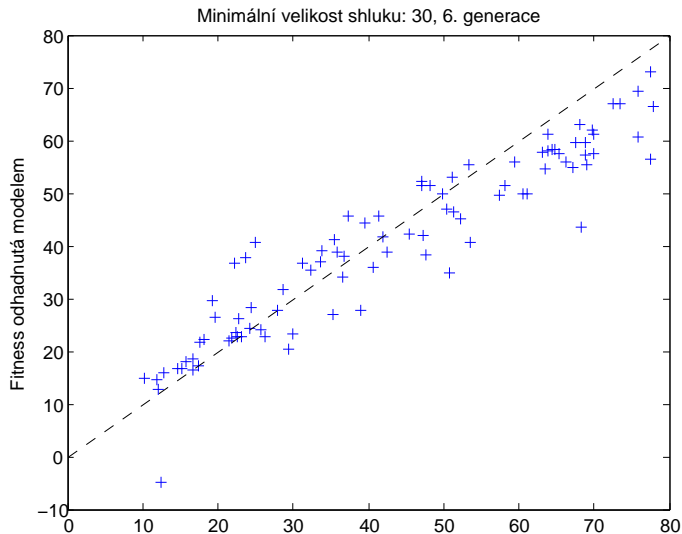
Testování na empirických datech

30 jedinců na 5. generaci



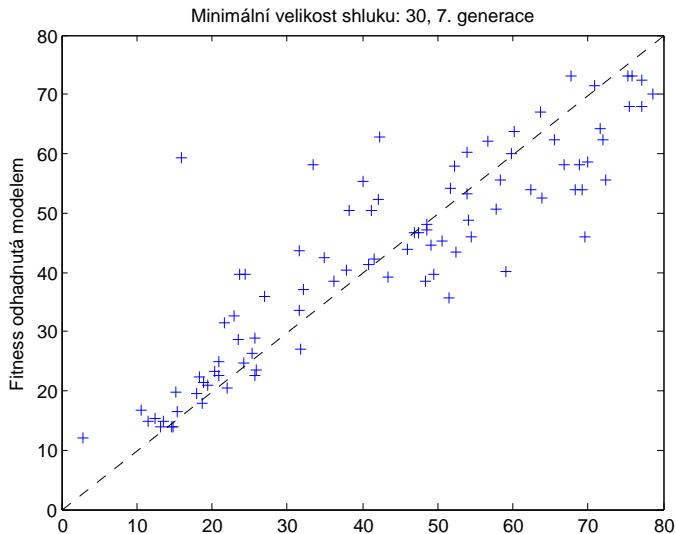
Testování na empirických datech

30 jedinců na 6. generaci



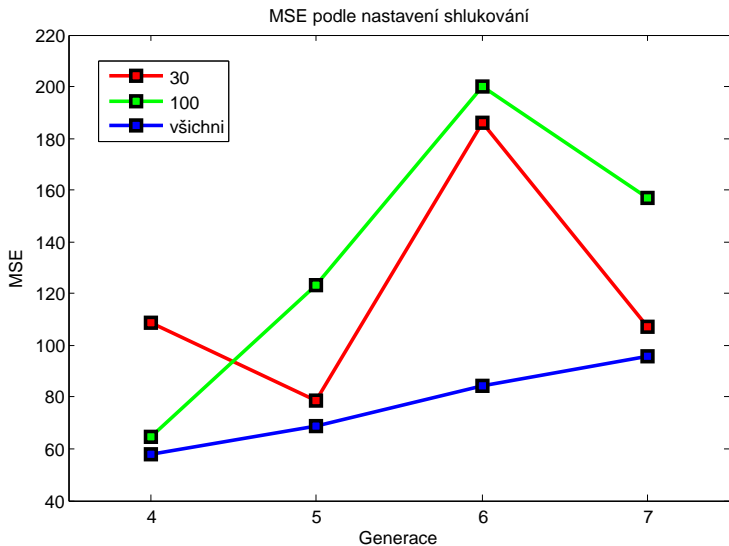
Testování na empirických datech

30 jedinců na 7. generaci



Testování na empirických datech

MSE v závislosti na minimálním počtu jedinců ve shluku

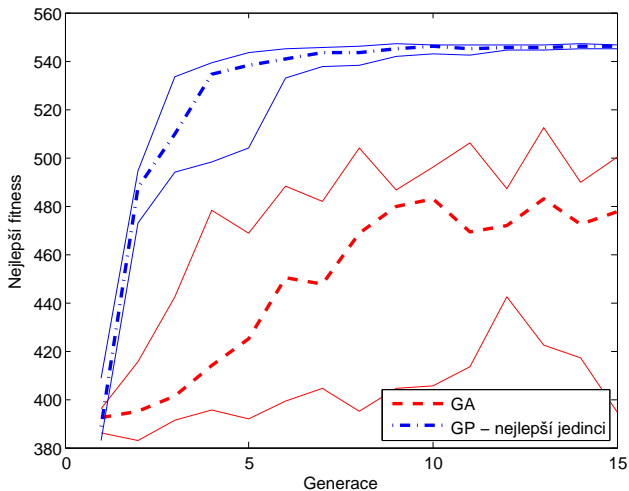


Testování na funkci valero

- Optimalizace umělé funkce *valero*
- 15. generací
- Porovnání genetického algoritmu bez náhradního modelu a algoritmu s náhradním modelem s gaussovskými procesy
- Řízení evoluce
 - generační
 - založené na jedincích
 - výběr nejlepších jedinců
 - výběr reprezentantů
 - kombinace

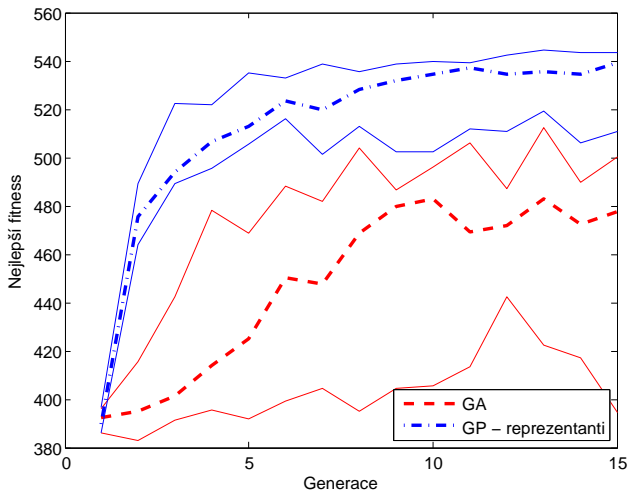
Testování na funkci valero

Závislost fitness funkce na generaci - individuální řízení - výběr nejlepších



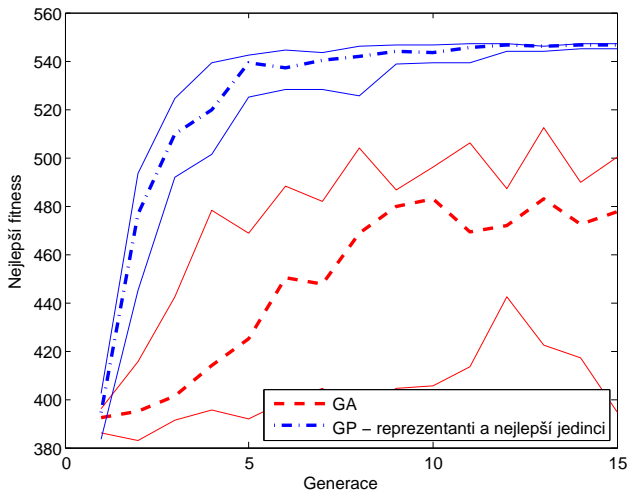
Testování na funkci valero

Závislost fitness funkce na generaci - individuální řízení - výběr reprezentantů



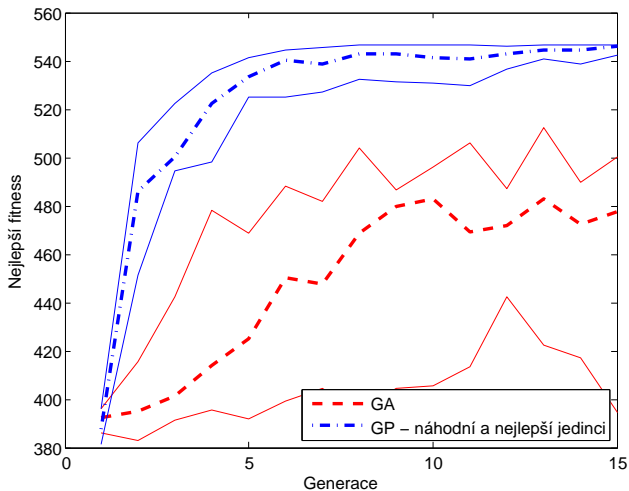
Testování na funkci valero

Závislost fitness funkce na generaci - individuální řízení - výběr nejlepších + reprezentantů



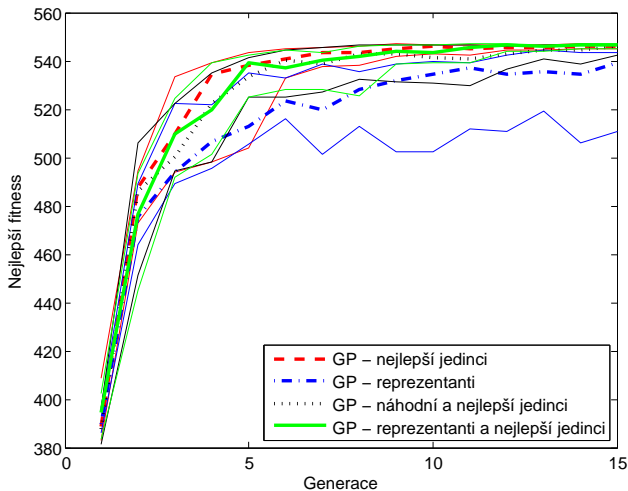
Testování na funkci valero

Závislost fitness funkce na generaci - individuální řízení - výběr nejlepších + náhodný



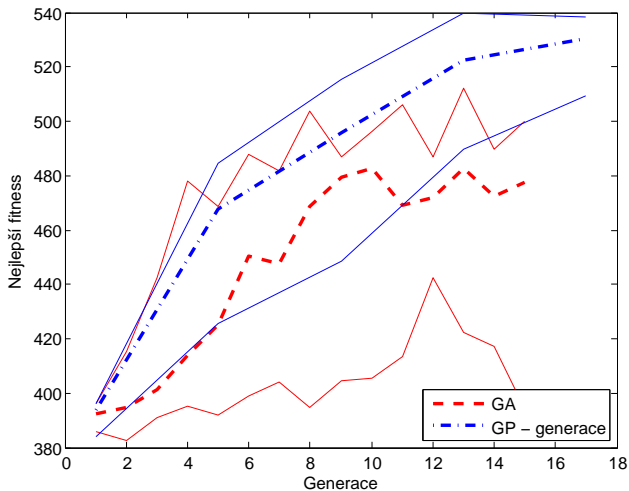
Testování na funkci valero

Závislost fitness funkce na generaci - individuální řízení



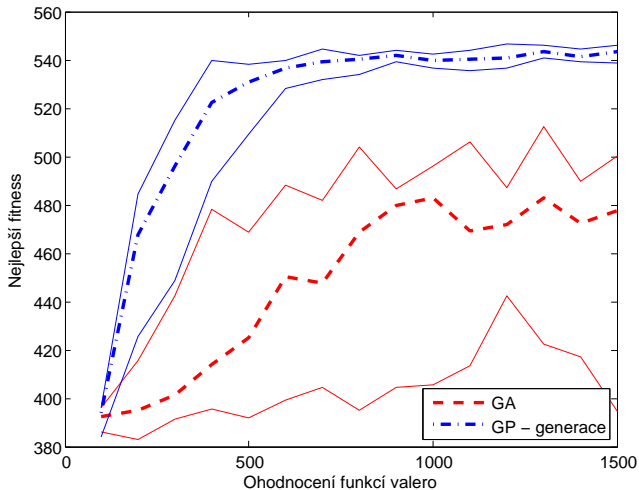
Testování na funkci valero

Závislost fitness funkce na generaci - generační řízení



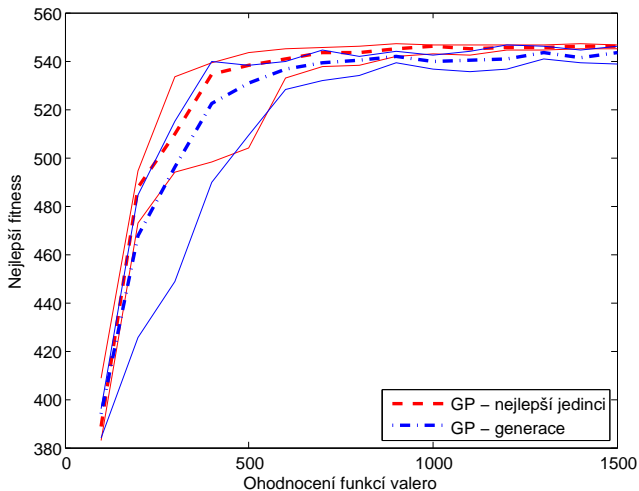
Testování na funkci valero

Závislost fitness funkce na počtu vyhodnocení - generační řízení



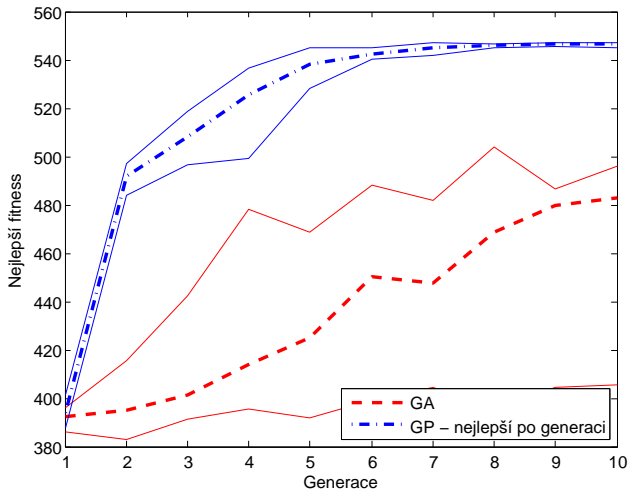
Testování na funkci valero

Závislost fitness funkce na počtu vyhodnocení - generační a individuální řízení



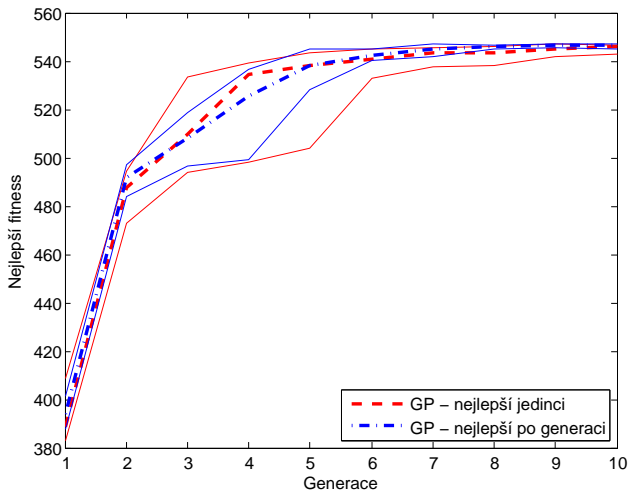
Testování na funkci valero

Závislost fitness funkce na generaci - individuální řízení - opakované trénování



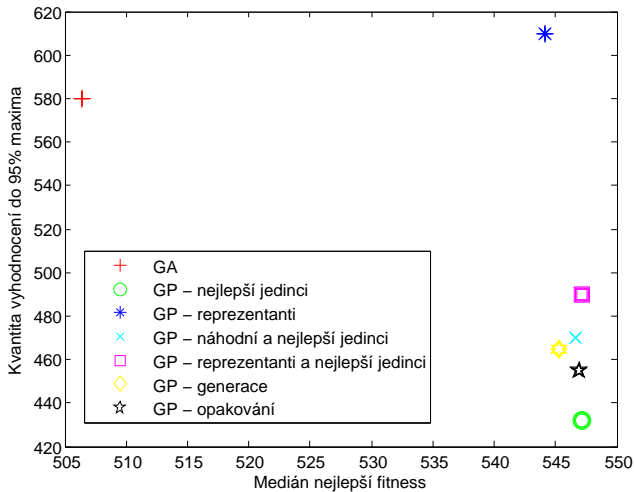
Testování na funkci valero

Závislost fitness funkce na generaci - individuální řízení - opakované trénování a bez opakování



Testování na funkci valero

Porovnání testovaných nastavení



Shrnutí výsledků testování

- Dobré výsledky na empirických datech
- Náhradní model s gaussovskými procesy urychluje konvergenci
- Největší urychlení — výběr nejlepších jedinců

Možnosti dalšího rozvoje systému GENACAT

- Rozšíření možností řízení evoluce
- Další typ náhradního modelu

Děkuji za pozornost!