Байесовская оптимизация

Алексей Зайцев
Руководитель
лаборатории
Сколтех

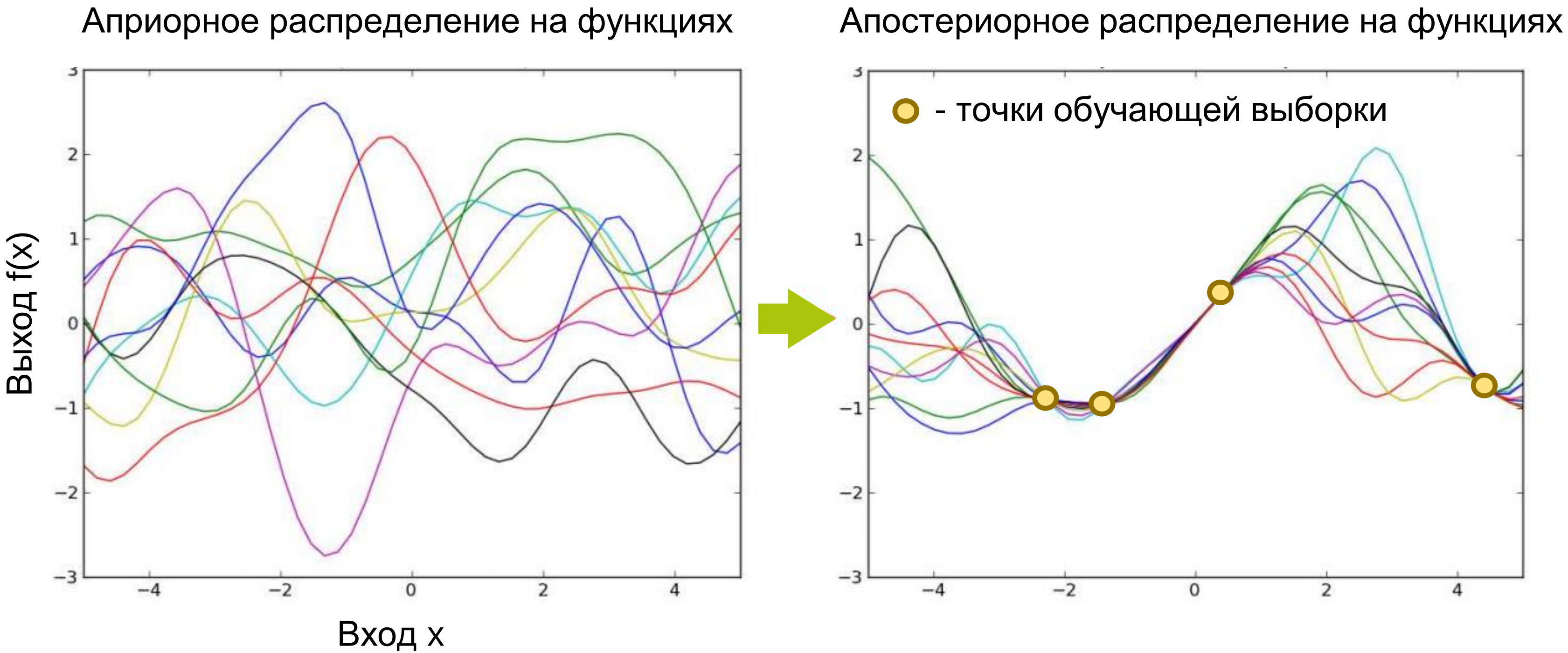
Skoltech

Про что лекция

1. Напоминание про регрессию на основе гауссовских процессов
2. Суррогатное моделирование и Байесовская оптимизация
3. Лучшие практики в Байесовской оптимизации

Идея регрессии на основе гауссовских процессов

Идея регрессии на основе гауссовских процессов

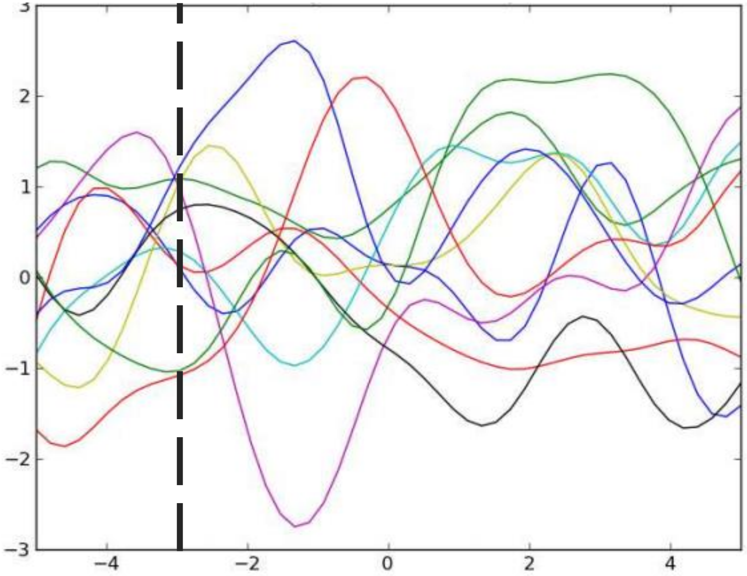


Источник рисунка: https://stats.stackexchange.com/questions/232959/simulating-the-posterior-of-a-gaussian-process

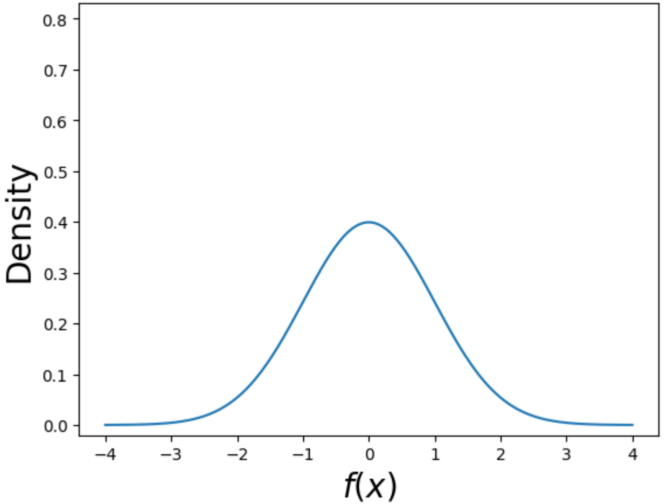
Skoltech

Априорные знания про значения функции

Априорное распределение на функциях

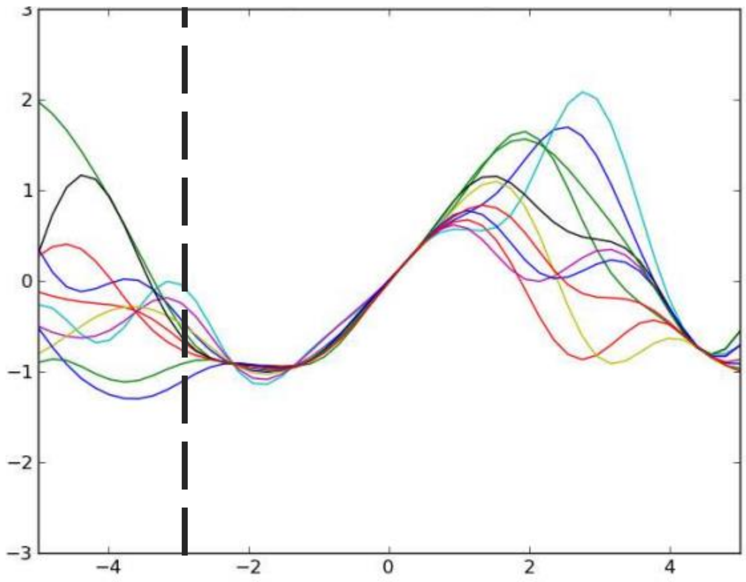


Распределение в точке x=-3

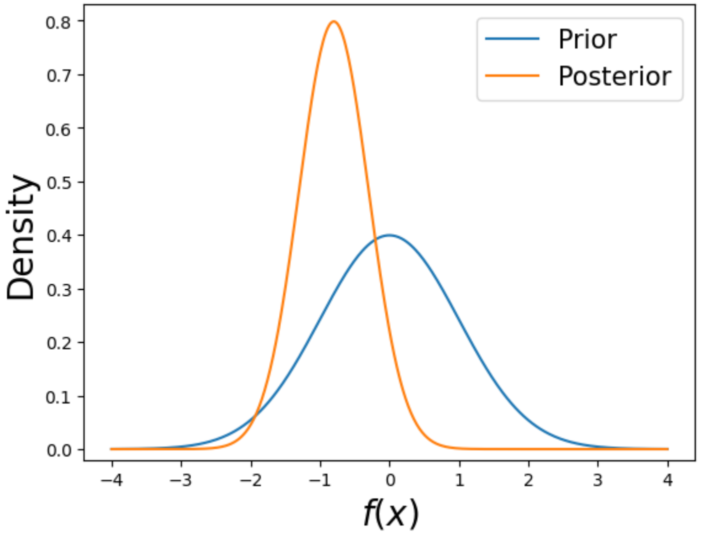


Апостериорные знания про значения функции

Апостериорное распределение на функциях

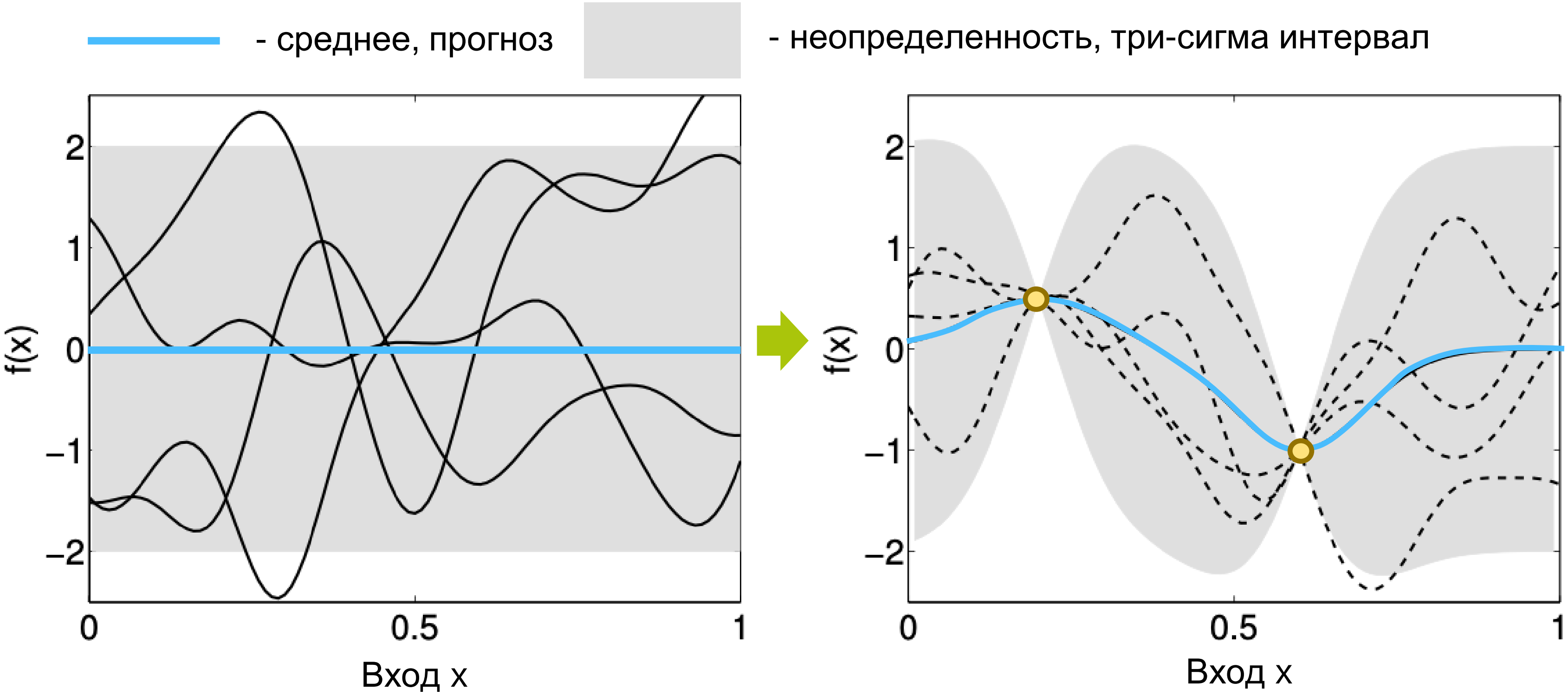


Распределение в точке x=-3



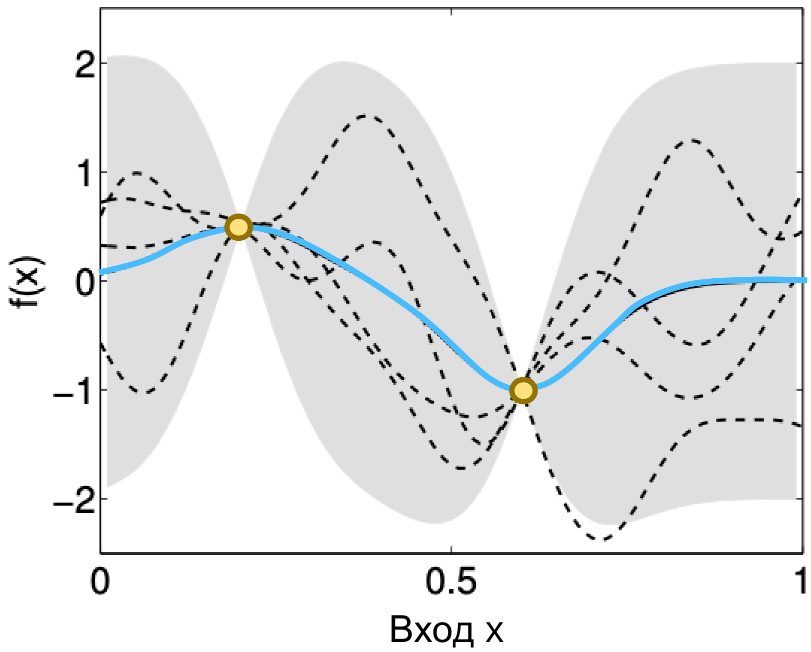
Skoltech

Усредним по всем функциям



Источник рисунка: Rasmussen and Christopher K. I. Williams. Chapter 1 Introduction

Свойства регрессии на основе гауссовских процессов



- Нелинейные прогноз
- Интерполяция
- Гладкость (зависит от ковариационной функции)
- Оценка неопределенности
- Выбор параметров с помощью метода максимума правдоподобия
- Явные и быстрые формулы для прогноза

Skoltech

Источник рисунка: Rasmussen and Christopher K. I. Williams. Chapter 1 Introduction

Оценка среднего и дисперсии для регрессии на основе гауссовских процессов

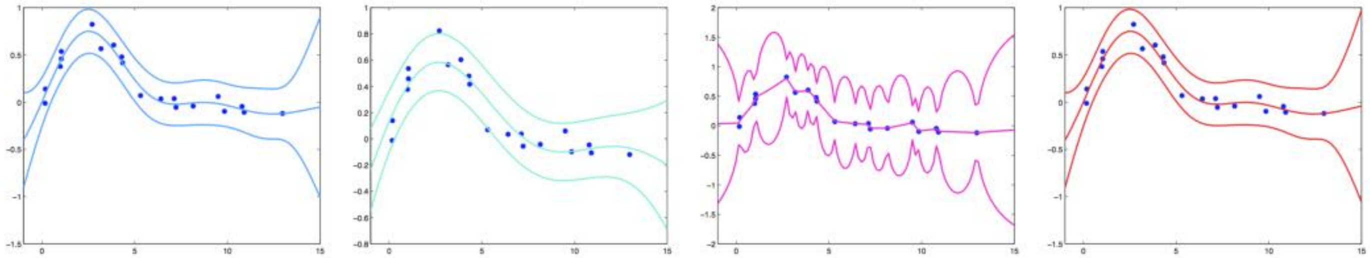
Аналитические формулы для среднего и дисперсии:

$$p(f_*|\mathbf{y}) = \mathcal{N}\left(f_*|\mu_*, \sigma_*^2\right),$$

$$\mu_* = \mathbf{k}_*^{\mathrm{T}}[\mathbf{K} + \sigma^2 \mathrm{I}_m]^{-1}\mathbf{y}, \ \ \sigma_*^2 = K_{**} - \mathbf{k}_*^{\mathrm{T}}[\mathbf{K} + \sigma^2 \mathrm{I}_m]^{-1}\mathbf{k}_*$$

$$\mathbf{K} = \{K(x_i, x_j)\}_{i,j=1}^{n} \quad \mathbf{k}_* = \{K(\mathbf{x}_*, \mathbf{x}_i)\}_{i=1}^{m} \text{ and } K_{**} = K(\mathbf{x}_*, \mathbf{x}_*)$$

$$\mu_* = \sum_{i=1}^{m} \alpha_i K(\mathbf{x}_*, \mathbf{x}_i), \ \ \boldsymbol{\alpha} = [\mathbf{K} + \sigma^2 \mathrm{I}_m]^{-1}\mathbf{y}$$



Skoltech

Pipeline Байесовской оптимизации

Какую функцию хотим оптимизировать?

Минимизируем тяжелую функцию в многомерном пространстве. Можем сделать ограниченное количество *B* запусков.

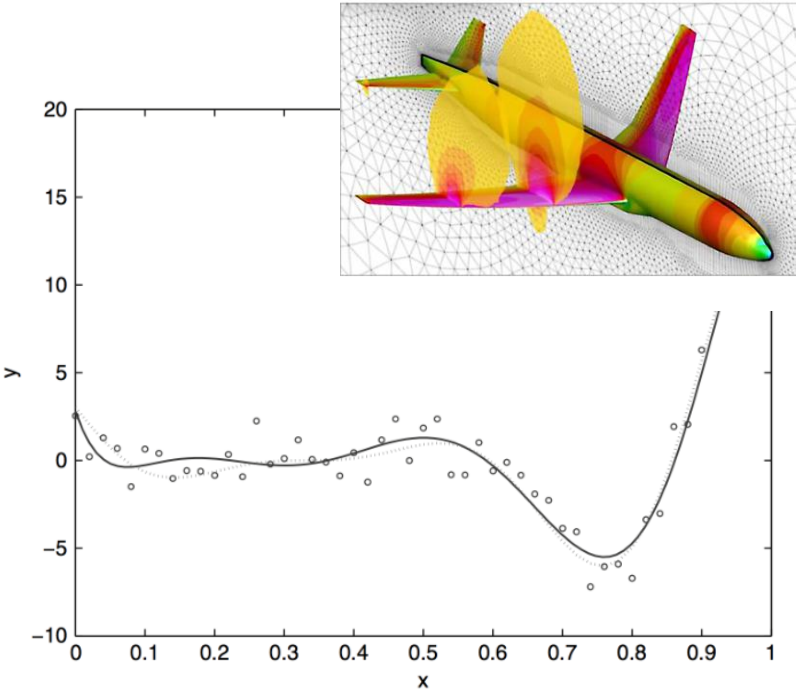
$$f(\boldsymbol{x}) \to \min_{x \in X}$$

Функция обычно:

- Тяжелая
- Гладкая
- Но наблюдаем мы ее с шумом

Наша задача:

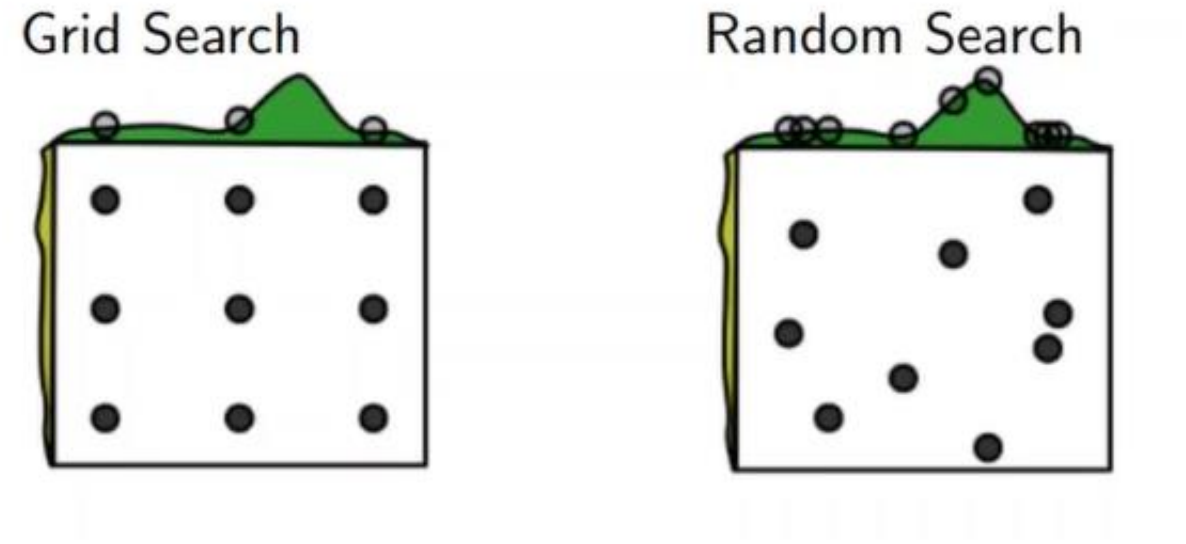
- Оптимизация гиперпараметров модели



AlphaGo

Skoltech

Базовые подходы



- Не принимают во внимание прошлые вычисления
- В общем случае нужно экспоненциальное по размерности количество вычислений целевой функции

Источник картинки: https://youtu.be/bcy6A57jAwI

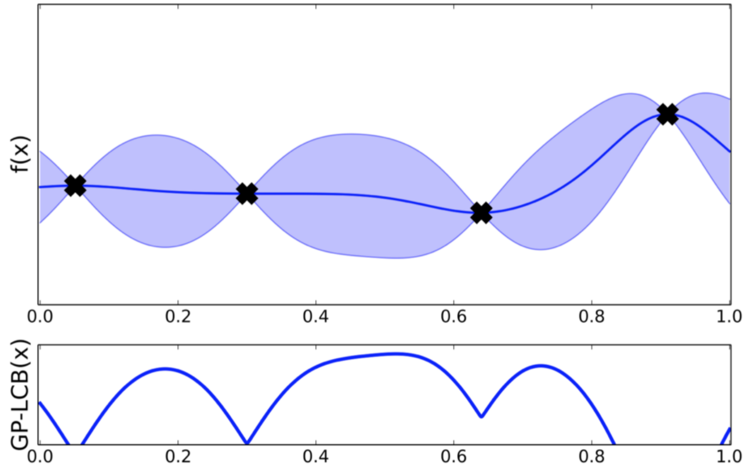
Алгоритм: как такую оптимизацию устроить

1. Генерируем начальную выборку
2. Учим суррогатную регрессионную модель на этой выборке
3. Оптимизируем суррогатную модель + ее неопределенность вместо исходной тяжелой модели, получили еще одну точку
4. Переучили суррогатную модель
5. Повторяем пока не закончится бюджет

Проблемы нет: теперь работает нормально

$$\alpha_{LCB}(\mathbf{x}) = \underbrace{-\mu_*(\mathbf{x})}_{\text{Прогноз}} + \zeta \cdot \underbrace{\sigma_*(\mathbf{x})}_{\text{Оценка неопределенности}}$$

Максимизируем $\alpha_{LCB}(\mathbf{x})$



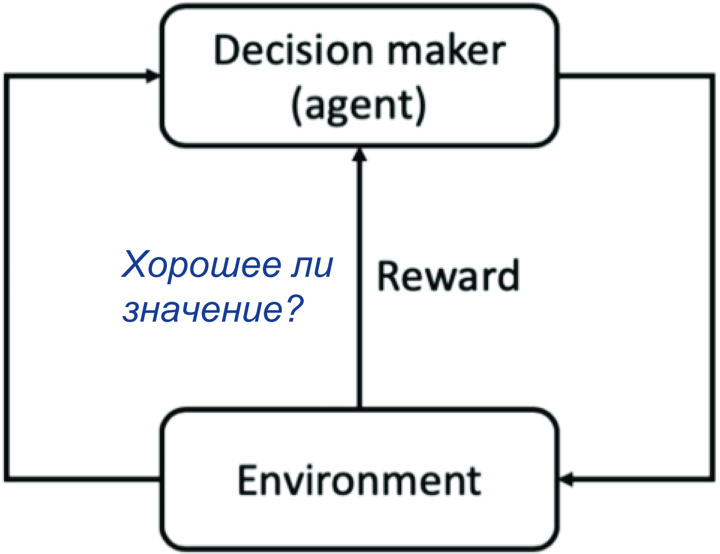
Почти обучение с подкреплением

Мы!

Наблюдение значения функции в новой точке

Observation

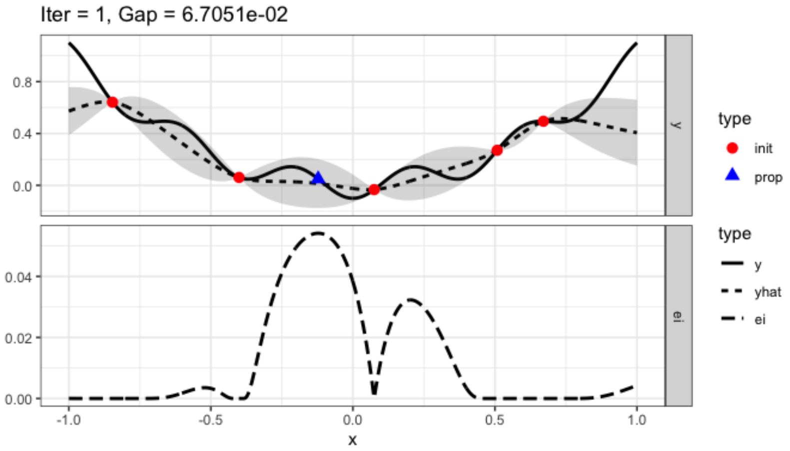
Хорошее ли значение?



Мы выбираем новую точку

Action

Доступная нам информация



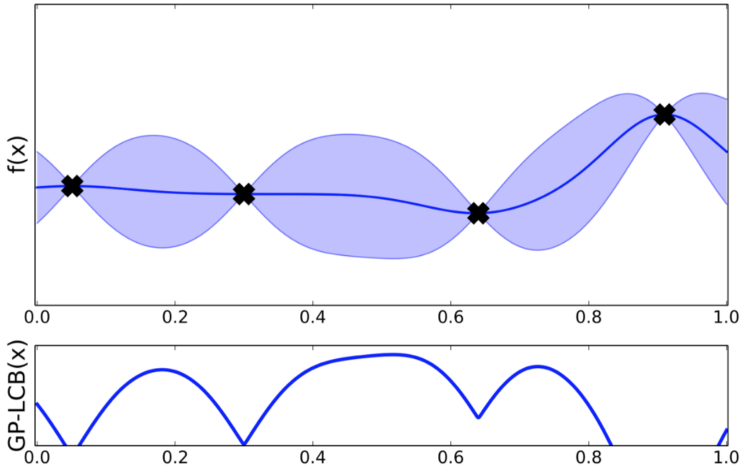
Skoltech

Acquisition function, функция выгоды LCB

Прогноз

Оценка неопределенности

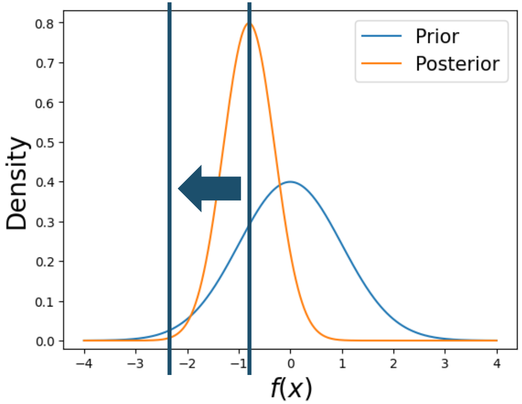
$$\alpha_{LCB}(\mathbf{x}) = -\mu_*(\mathbf{x}) + \zeta \cdot \sigma_*(\mathbf{x})$$



Максимизируем Lower Confidence Bound

$$\alpha_{LCB}(\mathbf{x})$$

Если максимизируем, то смотрим на Upper Confidence Bound

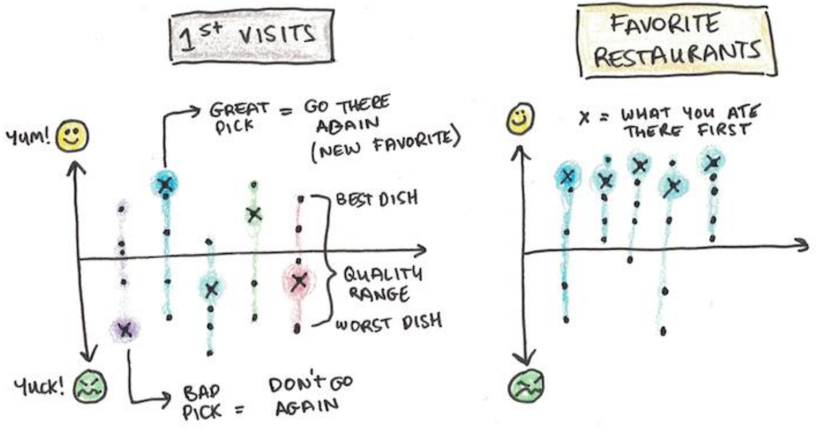


Теорема: мы достаточно быстро сойдемся к минимуму, ζ должна расти как корень из количества итераций.

Функция выгоды балансирует exploration и exploitation

Exploitation — Exploration

$$\alpha_{LCB}(\mathbf{x}) = -\mu_*(\mathbf{x}) + \zeta \cdot \sigma_*(\mathbf{x})$$



Функция выгоды Expected Improvement

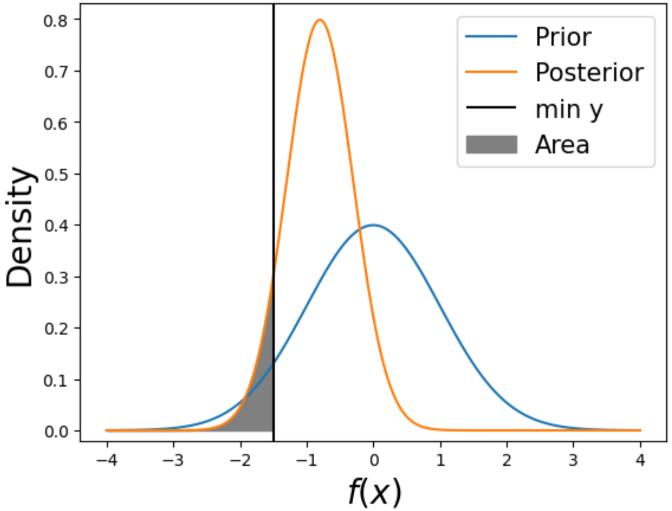
$$\Delta(\mathbf{x}) = y_{\text{best}} - \mu_*(\mathbf{x}),\ y_{\text{best}} = \min_{i=1,\dots,m} y_i,$$

$$\alpha_{EI}(\mathbf{x}) = \int \max(0, y_{\text{best}} - y_*) p(y_*|\mathbf{x}) dy_* =$$

$$= \mathrm{E} \max(0, y_{best} - y_*)$$

$$= \Delta(\mathbf{x}) \Phi\left(-\Delta(\mathbf{x})/\sigma_*(\mathbf{x})\right) + \sigma_*(\mathbf{x}) \varphi\left(\Delta(\mathbf{x})/\sigma_*(\mathbf{x})\right)$$

Мы считаем математическое ожидание улучшения



Функция выгоды Tree Parson Estimator TPE

Пусть у нас есть две генеративных модели

$\mathrm{P}(x \mid \mathrm{bad}) = \mathrm{P}(x|y > y_{min})$

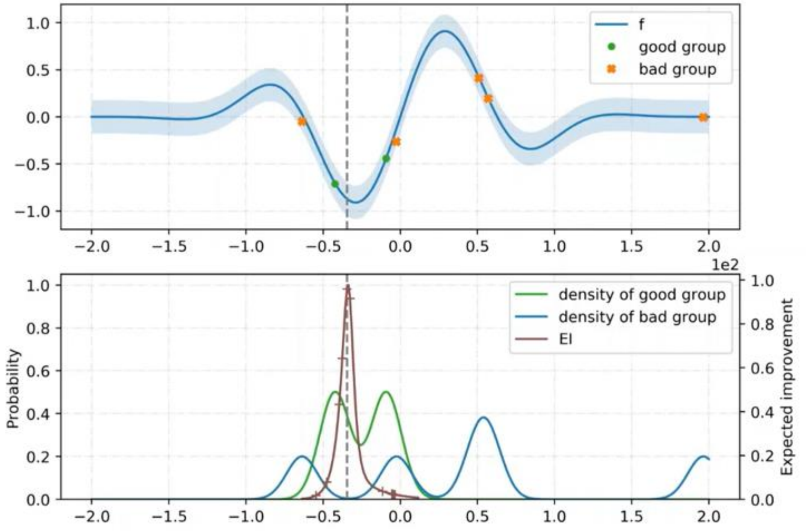
$\mathrm{P}(x \mid \mathrm{good}) = \mathrm{P}(x|y \leq y_{min})$

Перспективность точки:

$$\frac{P(x \mid \mathrm{good})}{P(x \mid \mathrm{bad})}$$

Это пропорционально Expected improvement

Такой метод используется в HyperOpt и Optuna



Функция выгоды Tree Parson Estimator TPE

Пусть у нас есть две генеративных модели

$\mathrm{P}(x \mid \text{bad}) = \mathrm{P}(x|y > y_{min})$

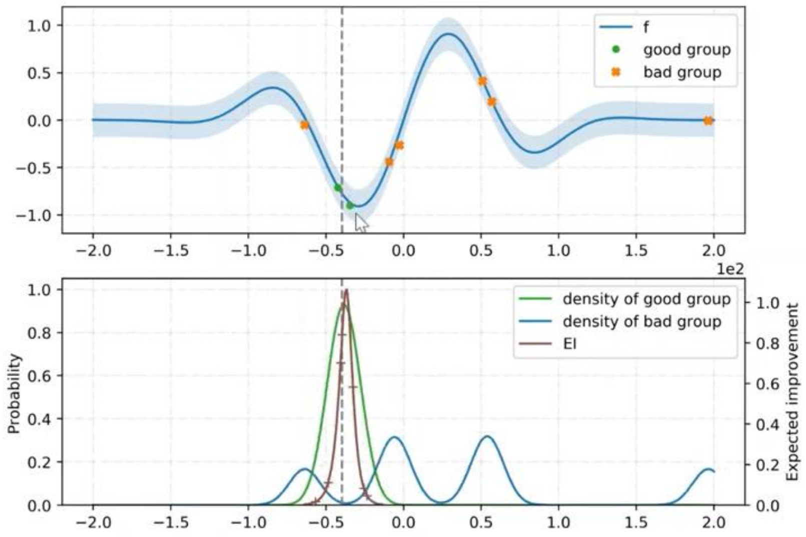
$\mathrm{P}(x \mid \text{good}) = \mathrm{P}(x|y \leq y_{min})$

Перспективность точки:

$$\frac{P(x \mid \text{good})}{P(x \mid \text{bad})}$$

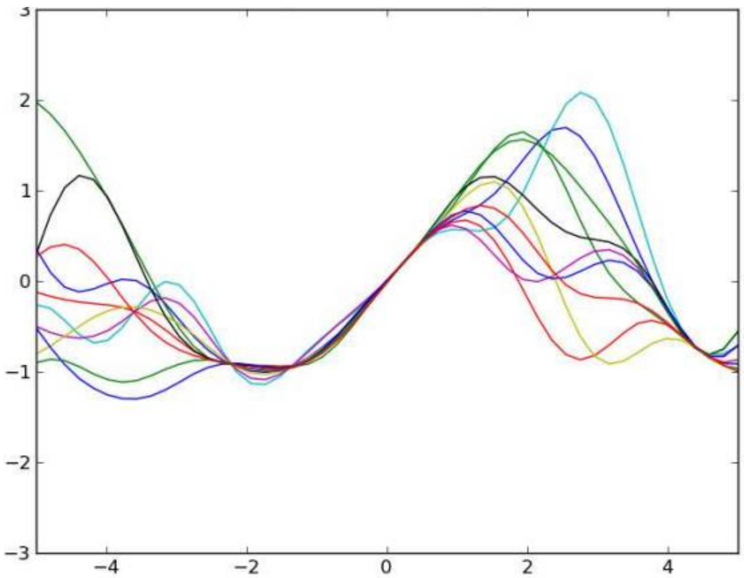
Это пропорционально Expected improvement

Такой метод используется в HyperOpt и Optuna



Сэмплирование Томпсона

Апостериорное распределение на функциях



На каждой итерации:

- Сэмплируем функцию
- Берем ее минимум

На практике медленней, но не застревает в локальных оптимумах

Skoltech

Функция выгоды Прирост энтропии

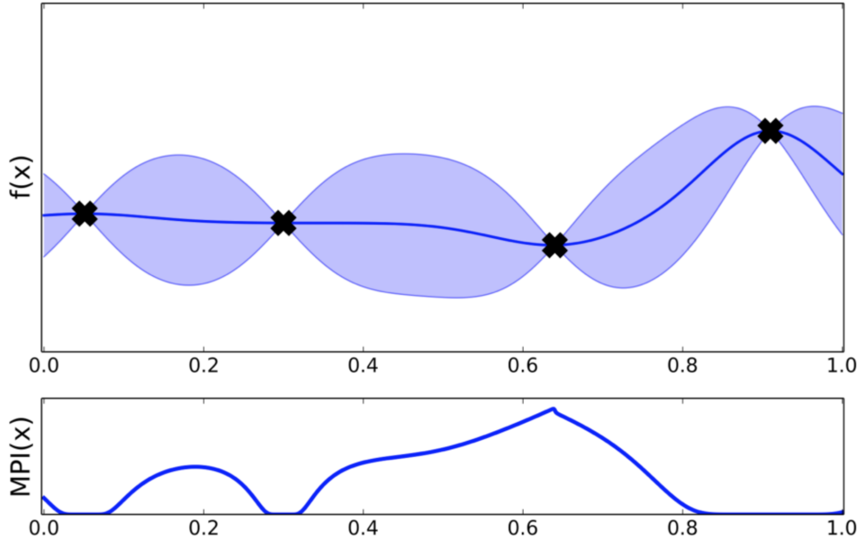
$$\alpha_{ES}(\mathbf{x}) = \mathcal{H}[p(\mathbf{x}_{\min}|\mathbf{y})] - \mathbb{E}_{p(y|\mathbf{y},\mathbf{x})}[\mathcal{H}[p(\mathbf{x}_{\min}|\mathbf{y} \cup \{\mathbf{x}, y\})]]$$

Мы считаем как улучшается наше знание про локацию минимума

- Явно посчитать уже не получается, нужно использовать методы Монте-Карло для приближенного вычисления интегралов

- Достаточно надежный метод

- Аналогично можно считать прирост знания не про точку минимума, а про значение в этой точке



Пример значений энтропийного критерия

Skoltech

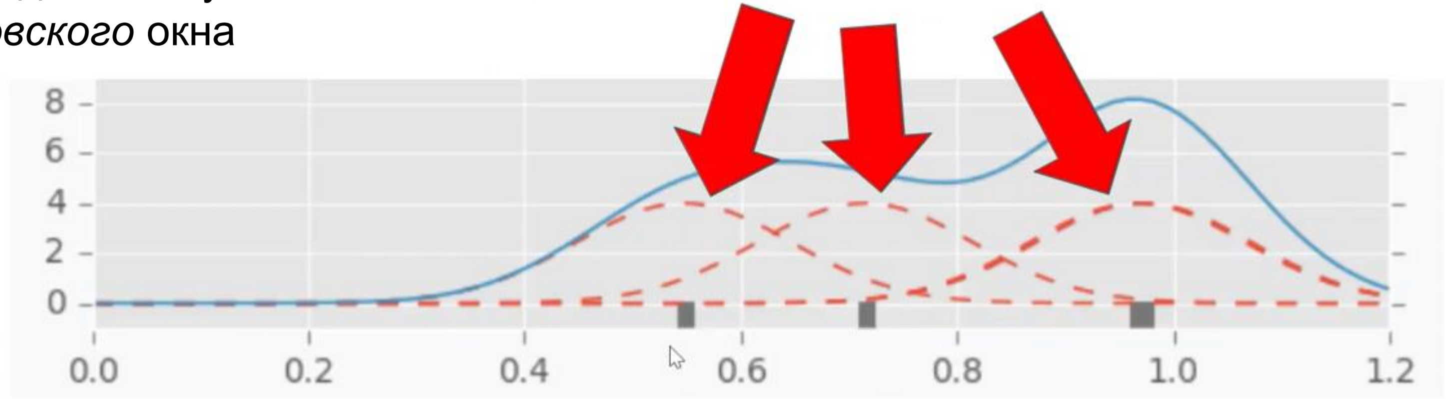
Детали оптимизации

Используем

- усеченные (truncated) нормальные распределения для непрерывных гиперпараметров
- категориальные распределения для дискретных

Байесовская модель – гауссовская смесь, метод *Парзеновского* окна

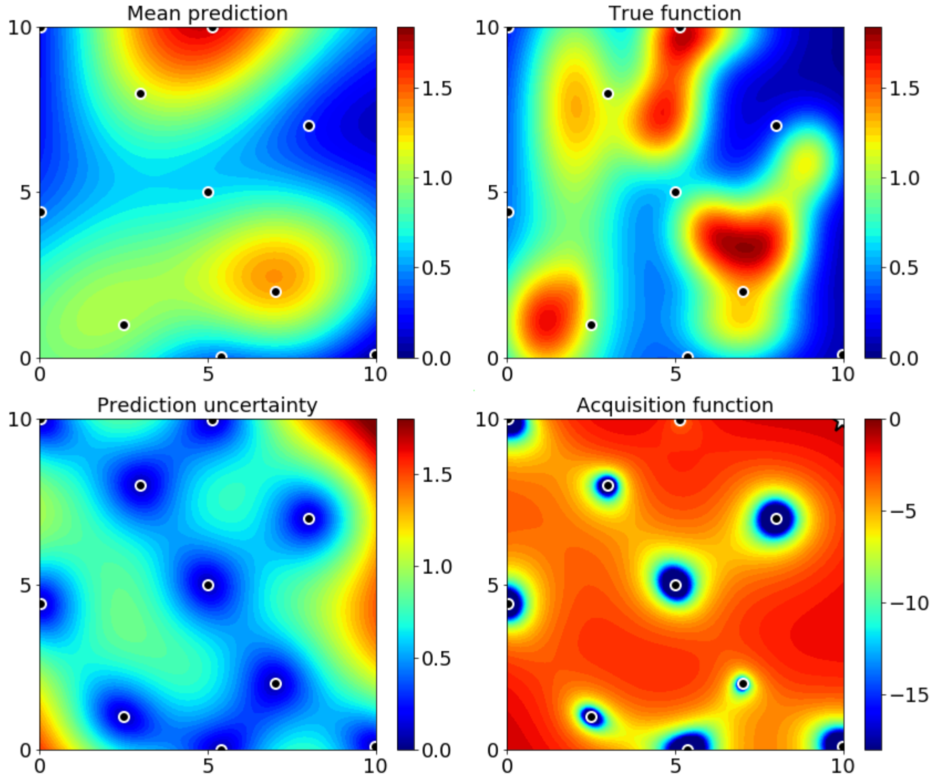
Точки из распределения



Bergstra, James, et al. Algorithms for hyper-parameter optimization. NeurIPS. 2011.

Как выглядит Expected Improvement

- Очень неприятная функция для оптимизации
- Но ее просто считать и нам не нужен истинный максимум
- Поэтому запускаем мультистарт 5-10 итераций и берем лучший результат

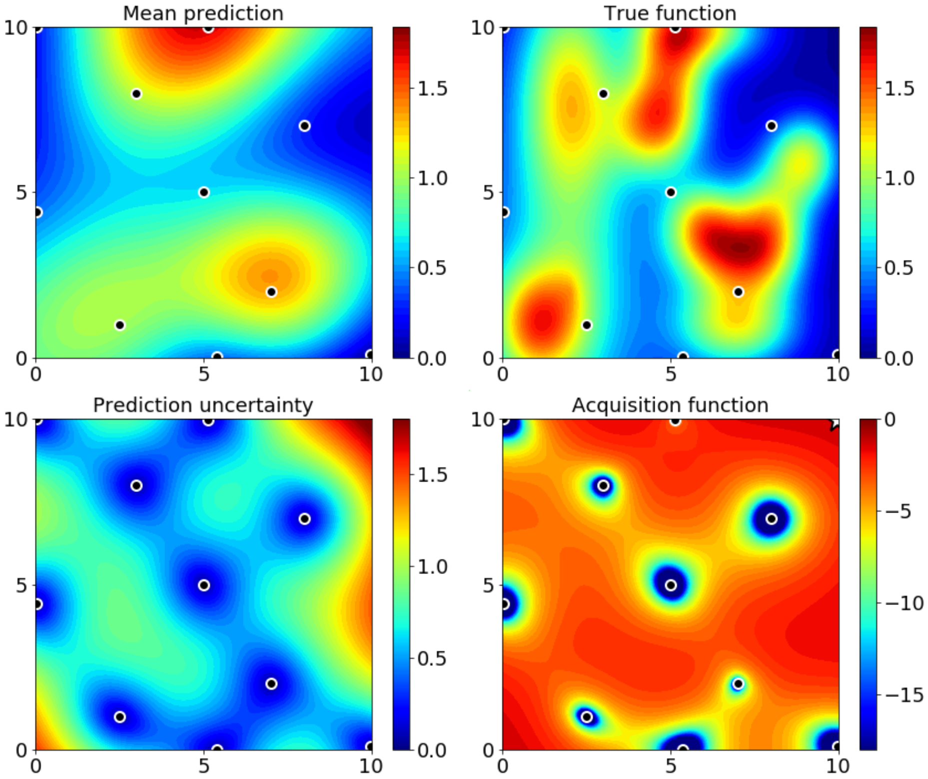


Skoltech

Параллельная оптимизация

- Если мы хотим учить модели параллельно, то мы можем добавить штраф за то, что мы похожи на предыдущую точку

- Все работает неплохо – потому что функция сложная для многомерного входа



Skoltech

Оптимизация с ограничениями

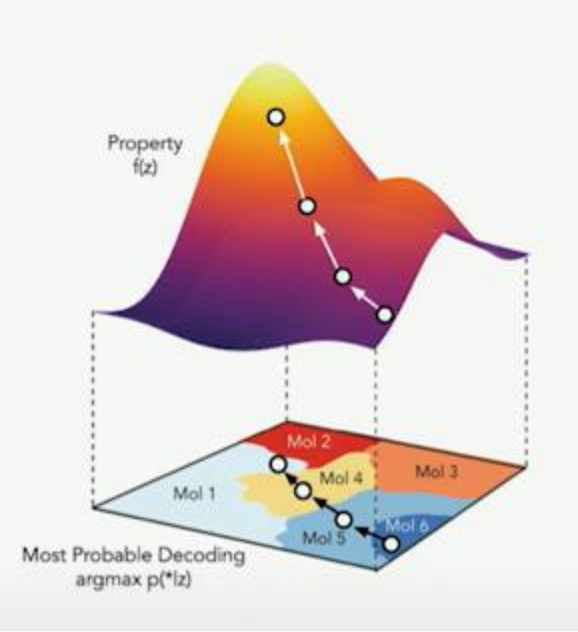
$$f(\boldsymbol{x}) \to \min_{x \in X} \qquad c.t.\ c(\boldsymbol{x}) \geq 0$$

Зависит от ограничений

- В общем – строим и для них модели
- Штрафуем за нарушение ограничений в модели

Приложения Байесовской оптимизации

- Оптимизация гиперпараметров, в том числе AlphaGo
- Суррогатная оптимизация: оптимизация сложных инженерных изделий
- Оптимизация свойств молекул в пространстве представлений
- Оптимизация в робототехнике
- Bayesian Optimization for a Better Desert

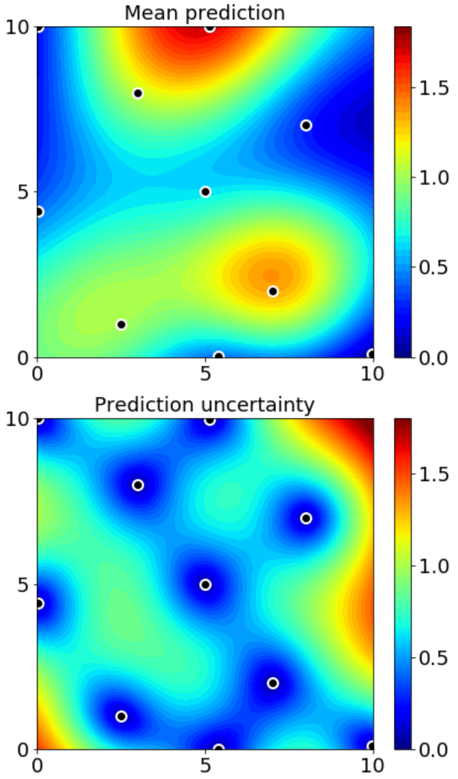


Skoltech

https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/46507.pdf

Связанная задача – активное обучение

- Как нам собрать такую выборку, чтобы построенная для нее модель была самой лучшей?
- Будем собирать данные последовательно, на каждом шаге нужно выбрать точку, которую нужно разметить
- У нас тоже возникает функция выгоды – какую точку брать? В этом случае речь только про exploration
- Оптимальный критерий с точки зрения теории – найти точку с наибольшей оценкой неопределенности
- На практике еще хорошо бы «смотреть в будущее»: как нам улучшить интеграл по неопределенности?



$$p(f_*|\mathbf{y}) = \mathcal{N}\left(f_*|\mu_*, \sigma_*^2\right),$$

$$\mu_* = \mathbf{k}_*^{\mathrm{T}}[\mathbf{K} + \sigma^2 \mathrm{I}_m]^{-1}\mathbf{y}, \;\; \sigma_*^2 = K_{**} - \mathbf{k}_*^{\mathrm{T}}[\mathbf{K} + \sigma^2 \mathrm{I}_m]^{-1}\mathbf{k}_*$$

Skoltech

Выводы

Выводы

- Байесовская (суррогатная) оптимизация позволяет быстро находить оптимумы тяжелых функций

- Она часто используется для оптимизации гиперпараметров

- Вариант по умолчанию – Ожидаемое улучшение EI или Парзеновская оценка на основе деревьев TPE

Ссылки

- Вводный материал про регрессию на основе гауссовских процессов

https://thegradient.pub/gaussian-process-not-quite-for-dummies/

- Основная книжка про использование гауссовских процессов в машинном обучении

https://gaussianprocess.org/gpml/

- Книжка про Байесовскую оптимизацию, 2023

https://bayesoptbook.com/

- Лекция про Байесовскую оптимизацию с конференции UAI

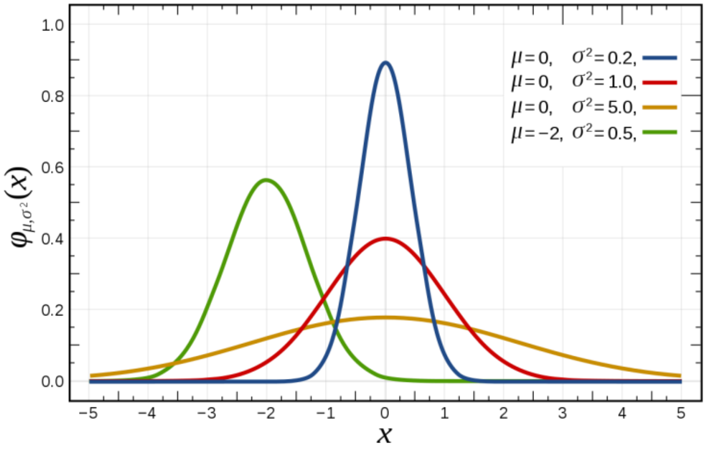
https://www.youtube.com/watch?v=C5nqEHpdyoE

Skoltech

Дополнительные слайды

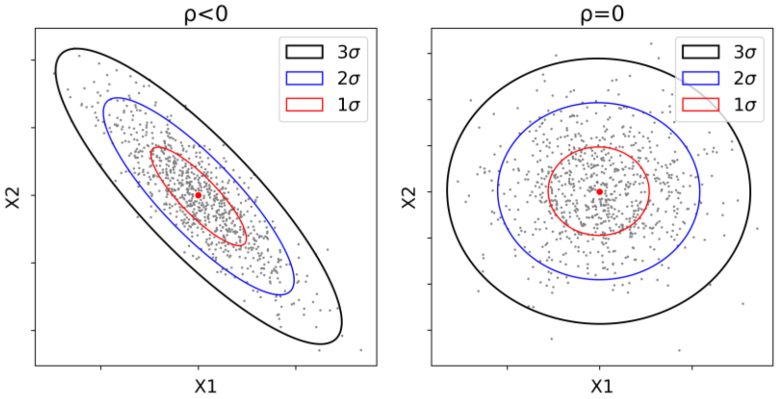
Многомерное гауссовское распределение

Плотность одномерного гауссовского распределения



$$p(x;\mu,\sigma^2) = \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right)$$

Плотность многомерного гауссовского распределения

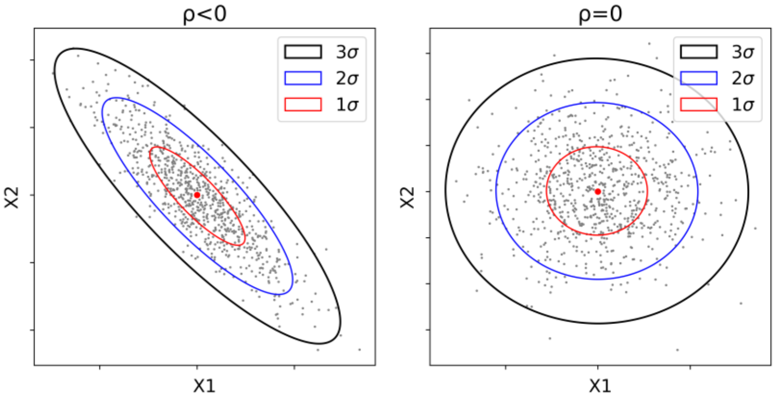


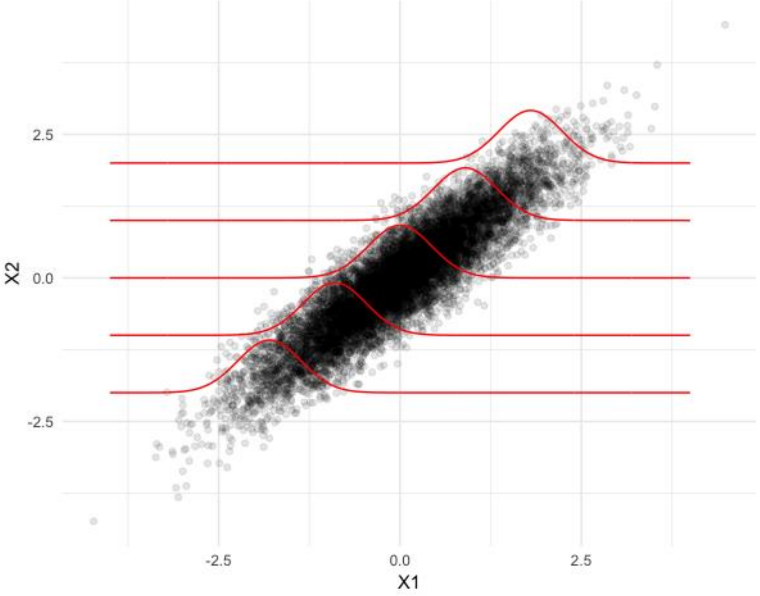
$$p(\boldsymbol{x};\boldsymbol{\mu},\Sigma) = \frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)$$

Skoltech

Условное гауссовское распределение

Плотность многомерного гауссовского распределения

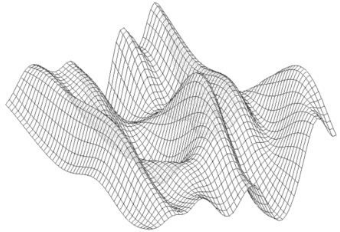




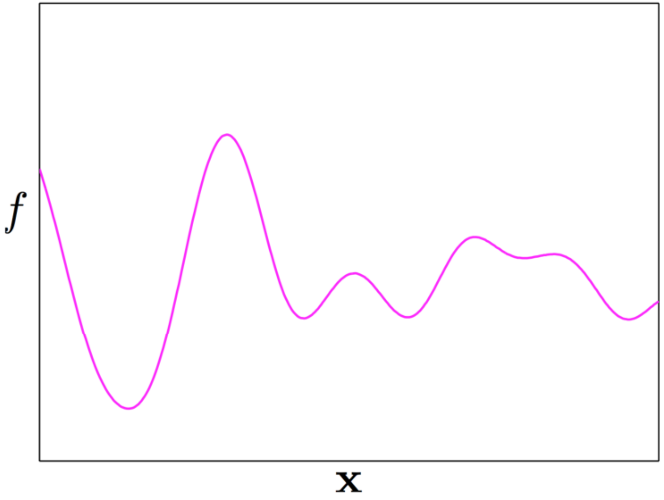
$$p(\boldsymbol{x}; \boldsymbol{\mu}, \Sigma) = \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2} (\boldsymbol{x} - \boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{x} - \boldsymbol{\mu}) \right)$$

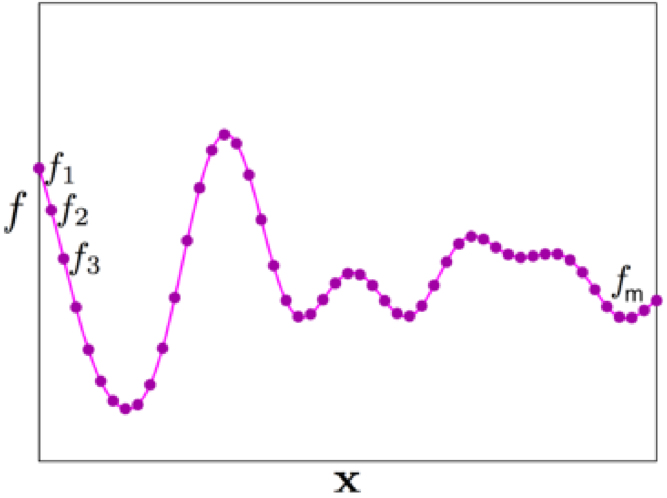
Skoltech

Гауссовский случайный процесс



Совместное распределение для любого множества точек - нормальное



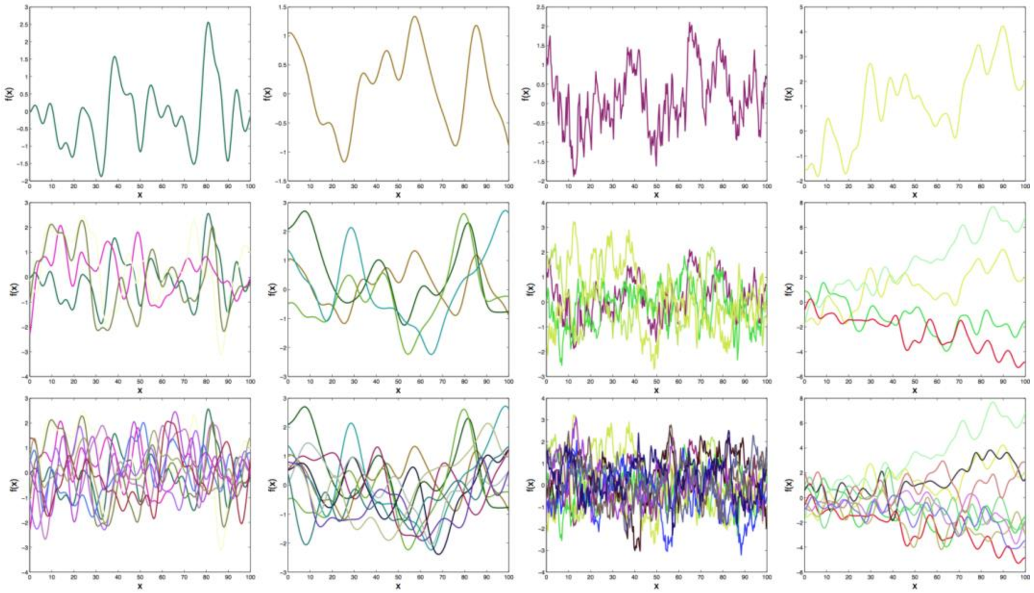


Skoltech

$$f(\mathbf{x}) \sim \mathcal{GP}(\cdot|\mu(\mathbf{x}), K(\mathbf{x}, \mathbf{x}'))$$

Регрессия на основе гауссовских процессов

$$K(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left\{-\sum_{i=1}^{d} \frac{(x_i - x_i')^2}{2r_i^2}\right\}$$



Метод максимума правдоподобия для оценки параметров

$$\mathcal{L} = -\log p(\mathbf{y}|\boldsymbol{\theta}) = \underbrace{\frac{1}{2}\log\det \mathbf{C}(\boldsymbol{\theta})}_{regularization} + \underbrace{\frac{1}{2}\mathbf{y}^{\mathrm{T}}\mathbf{C}^{-1}(\boldsymbol{\theta})\mathbf{y}}_{data-fit} + \frac{m}{2}\log(2\pi),$$

- Умеем считать правдоподобие
- Умеем считать производные
- Запускаем градиентный спуск до сходимости, часто глобальный оптимум один

Skoltech