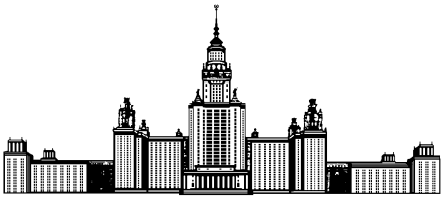


Московский государственный университет имени М. В. Ломоносова
Факультет вычислительной математики и кибернетики
Кафедра математических методов прогнозирования

Измаилов Павел Алексеевич

Алгоритмы обучения гауссовских процессов для больших объемов данных

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

Научный руководитель:
к.ф-м.н.
Ветров Дмитрий Петрович

Научный консультант:
научный сотрудник
Кропотов Дмитрий Александрович

Москва, 2017

Содержание

Аннотация

В данной работе предлагается новый подход TT-GP к обучению моделей на основе гауссовских процессов для задач регрессии и классификации. Предлагаемый подход основан на стохастическом вариационном выводе, интерполяции ядровой функции и использовании тензорного разложения Tensor Train. Метод TT-GP позволяет использовать огромное число вспомогательных точек, что в свою очередь позволяет выучивать экспрессивные ядровые функции на основе нейронных сетей с миллионами параметров. В конце работы приводится экспериментальное сравнение предлагаемого подхода с ведущими существующими методами, и на ряде задач удается показать существенное улучшение качества.

1 Введение

Гауссовские процессы (см. раздел 1.1) задают априорное распределение на множестве функций и позволяют находить сложные закономерности в данных. На основе гауссовских процессов построено множество моделей, успешно применяющихся для решения различных задач машинного обучения — регрессии, классификации, понижения размерности. Эти методы позволяют автоматически настраивать сложность модели, а также позволяют оценивать неопределенность в прогнозе.

Стандартные методы на основе гауссовских процессов имеют кубическую сложность от размера выборки, что не позволяет использовать их в задачах с большими объемами данных. В связи с этим начиная с 2000-х годов, многие исследователи занимаются разработкой приближенных схем для обучения моделей гауссовских процессов. Разработка методов на основе вспомогательных точек (inducing inputs) позволила применять гауссовские процессы к задачам с большими выборками (более 100000 объектов). В работе [1] предлагается подход, позволяющий трактовать значения гауссовского процесса во вспомогательных точках как латентные переменные, и осуществлять настройку модели с помощью вариационного вывода. Данная работа получила развитие в [2], [3], где был предложен способ настройки модели с помощью стохастической оптимизации и обобщение метода на задачу классификации.

Другим важным направлением исследований в области масштабируемых гауссовских процессов является использование структуры матриц ковариации. В работе [4] предлагается метод, позволяющий существенно ускорить обучение моделей гауссовских процессов за счет представления матрицы ковариации в виде произведения Кронекера. К сожалению, методы данного класса не применимы для задач, в которых признаковое пространство имеет высокую размерность.

Кроме того, ведутся исследования в области построения экспрессивных ядровых функций для гауссовских процессов (например, [5]), в том числе с применением глубоких нейронных сетей ([6]). С помощью таких ядровых функций удается успешно

применять модели на основе гауссовских процессов к данным высокой размерности (в том числе к изображениям).

Другой подход к объединению глубинного обучения с гауссовскими процессами основан на построении нейронных сетей из гауссовских процессов [7] (модель Deep Gaussian Process). В последнее время предложен ряд аппроксимаций, позволяющий применять данную модель к задачам с большими объемами данных (см. [8], [9]).

В данной работе предлагается новый приближенный метод TT-GP для обучения моделей на основе гауссовских процессов для задач регрессии и классификации. Данный метод основан на стохастическом вариационном выводе для модели со вспомогательными точками, интерполяции ядровой функции, и тензорном разложении Tensor Train ([10]). Данный метод позволяет использовать огромное число вспомогательных точек за счет чего удается эффективно решать задачи регрессии и классификации, для которых существующие методы не применимы или дают низкое качество.

Кроме того, предлагается способ построения ядровых функций на основе нейронных сетей и обучения гауссовских процессов с такими ядровыми функциями. Данный подход позволяет применять гауссовские процессы, например, к задачам компьютерного зрения, и существенно улучшить результаты существующих методов на этих задачах.

В разделе 1.1 дается определение гауссовского процесса и связанных понятий. В разделах 1.2, 1.3, 1.4 и 1.5 описывается стандартная модель для регрессии и классификации на основе гауссовских процессов и ее расширение на основе вспомогательных точек. В разделе 1.6 дается описание метода KISS-GP, основанного на интерполяции ядровой функции. Тензорный формат Tensor Train описывается в разделе 1.7. В секции 2 подробно описывается предлагаемый метод. Наконец, в разделе 3 приводятся результаты экспериментов по сравнению предлагаемого подхода с существующими аналогами.

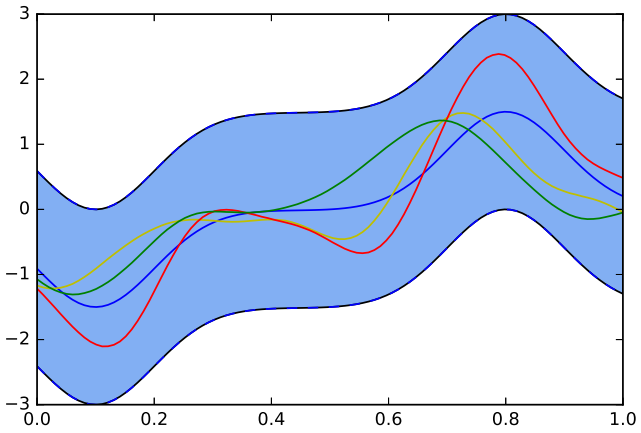


Рис. 1: Одномерный гауссовский процесс

1.1 Гауссовские процессы

Гауссовским процессом называется случайный процесс, чьи конечномерные распределения гауссовские.

На рисунке 1 приводится пример гауссовского процесса. Голубая область показывает 3σ-интервал для значений процесса, а темно-синяя линия — его матожидание. Цветные линии показывают случайные реализации процесса.

В данной работе рассматриваются гауссовские процессы, заданные на вещественных пространствах $\mathbb{R}^D$. В таком случае случайный процесс f называется гауссовским процессом, если для любого n, и для любого набора $\boldsymbol{t}_1, \boldsymbol{t}_2, \ldots, \boldsymbol{t}_n \in \mathbb{R}^D$ совместное распределение имеет вид

$$f(\boldsymbol{t}_1), f(\boldsymbol{t}_2), \ldots, f(\boldsymbol{t}_n) \sim \mathcal{N}(\boldsymbol{m}_t, \boldsymbol{K}_t),$$

для некоторых $\boldsymbol{m}_t \in \mathbb{R}^n$, $\boldsymbol{K}_t \in \mathbb{R}^{n \times n}$.

Матожидание $\boldsymbol{m}_t$ определяется функцией среднего $m : \mathbb{R}^D \to \mathbb{R}$ гауссовского процесса:

$$\boldsymbol{m}_t = (m(\boldsymbol{t}_1), m(\boldsymbol{t}_2), \ldots, m(\boldsymbol{t}_n))^T.$$

Функция матожидания может быть произвольной.

Аналогично, матрица ковариации $\boldsymbol{K}$ определяется ковариационной (ядровой) функцией процесса $k : \mathbb{R}^D \times \mathbb{R}^D \to \mathbb{R}$:

$$\boldsymbol{K}_t = \begin{pmatrix} k(\boldsymbol{t}_1, \boldsymbol{t}_1) & k(\boldsymbol{t}_1, \boldsymbol{t}_2) & \dots & k(\boldsymbol{t}_1, \boldsymbol{t}_n) \\ k(\boldsymbol{t}_2, \boldsymbol{t}_1) & k(\boldsymbol{t}_2, \boldsymbol{t}_2) & \dots & k(\boldsymbol{t}_2, \boldsymbol{t}_n) \\ \dots & \dots & \dots & \dots \\ k(\boldsymbol{t}_n, \boldsymbol{t}_1) & k(\boldsymbol{t}_n, \boldsymbol{t}_2) & \dots & k(\boldsymbol{t}_n, \boldsymbol{t}_n) \end{pmatrix}.$$

Ядровая функция должна быть ядром — матрица ковариации для любого набора значений $\boldsymbol{t}_1, \boldsymbol{t}_2, \dots, \boldsymbol{t}_n$ должна быть симметричной и неотрицательно определенной.

Гауссовский процесс полностью определяется своими функцией матожидания и ковариационной функцией. Ниже мы будем использовать запись

$$f \sim \mathcal{GP}(m(\cdot), k(\cdot, \cdot))$$

для обозначения гауссовского процесса с функцией матожидания m и ковариационной функцией k.

1.2 Модель для регрессии

Рассмотрим модель регрессии на основе гауссовских процессов. Пусть $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_n)^T \in \mathbb{R}^{n \times D}$ — признаковое описание выборки из n объектов. Пусть $\boldsymbol{y} = (y_1, y_2, \dots, y_n)^T \in \mathbb{R}^n$ — соответствующие им значения целевой переменной. Будем считать, что наблюдаемые переменные $\boldsymbol{y}$ — зашумленные значения некоторого скрытого гауссовского процесса $f : \mathbb{R}^D \to \mathbb{R}$.

$$f \sim \mathcal{GP}(0, k(\cdot, \cdot))$$

с нулевым матожиданием и ковариационной функцией $k(\cdot, \cdot)$.

Введем скрытые переменные $\boldsymbol{f} = (f_1, f_2, \dots, f_n) \in \mathbb{R}^n$ — значения процесса f в точках обучающей выборки. Тогда

$$p(y_i | f_i) = \mathcal{N}(y_i | f_i, \nu^2),$$

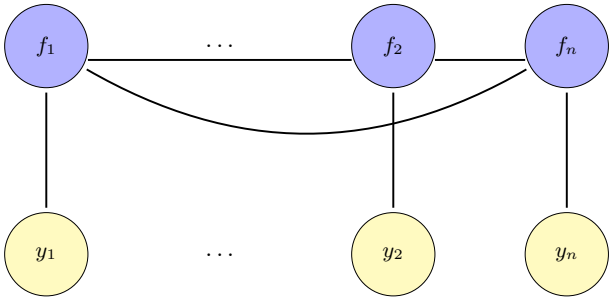


Рис. 2: Графическая модель для модели регрессии (классификаци) на основе гауссовских процессов

где ν — дисперсия шума. Требуется оценить неизвестное значение процесса $\boldsymbol{f}_* \in \mathbb{R}^l$ в наборе новых точек $\boldsymbol{X}_* \in \mathbb{R}^{l \times D}$. Рассмотрим модель

$$p(\boldsymbol{y}, \boldsymbol{f}|\boldsymbol{X}) = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{X}) = p(\boldsymbol{f}|\boldsymbol{X}) \prod_{i=1}^{n} p(y_i|f_i). \tag{1}$$

Соответствующая графическая модель изображена на рис. 2

Введем следующее обозначение. Будем обозначать матрицу попарных значений ковариационной функции, вычисленную на двух наборах точек $\boldsymbol{A} = (\boldsymbol{a}_1, \boldsymbol{a}_2, \dots, \boldsymbol{a}_n)^T \in \mathbb{R}^{n \times D}$ и $\boldsymbol{B} = (\boldsymbol{b}_1, \boldsymbol{b}_2, \dots, \boldsymbol{b}_m)^T \in \mathbb{R}^{m \times D}$ через

$$K(\boldsymbol{A}, \boldsymbol{B}) = \begin{pmatrix} k(\boldsymbol{a}_1, \boldsymbol{b}_1) & k(\boldsymbol{a}_1, \boldsymbol{b}_2) & \dots & k(\boldsymbol{a}_1, \boldsymbol{b}_m) \\ k(\boldsymbol{a}_2, \boldsymbol{b}_1) & k(\boldsymbol{a}_2, \boldsymbol{b}_2) & \dots & k(\boldsymbol{a}_2, \boldsymbol{b}_m) \\ \dots & \dots & \dots & \dots \\ k(\boldsymbol{a}_n, \boldsymbol{b}_1) & k(\boldsymbol{a}_n, \boldsymbol{b}_2) & \dots & k(\boldsymbol{a}_n, \boldsymbol{b}_m) \end{pmatrix} \in \mathbb{R}^{n \times m}.$$

Тогда по определению гауссовского процесса распределение

$$\begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{f}_* \end{bmatrix} \sim \mathcal{N}\left(\boldsymbol{0}, \begin{bmatrix} K(\boldsymbol{X}, \boldsymbol{X}) & K(\boldsymbol{X}, \boldsymbol{X}_*) \\ K(\boldsymbol{X}_*, \boldsymbol{X}) & K(\boldsymbol{X}_*, \boldsymbol{X}_*) \end{bmatrix}\right).$$

Так как $\boldsymbol{y}$ представляет из себя зашумленную версию $\boldsymbol{f}$, то

$$\begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{f}_* \end{bmatrix} \sim \mathcal{N}\left(\boldsymbol{0}, \begin{bmatrix} K(\boldsymbol{X}, \boldsymbol{X}) + \nu^2 \boldsymbol{I} & K(\boldsymbol{X}, \boldsymbol{X}_*) \\ K(\boldsymbol{X}_*, \boldsymbol{X}) & K(\boldsymbol{X}_*, \boldsymbol{X}_*) \end{bmatrix}\right).$$

Можем выразить условное распределение на $\boldsymbol{f}_*$ при условии данных

$$\boldsymbol{f}_*|\boldsymbol{y} \sim \mathcal{N}(\hat{\boldsymbol{m}}, \hat{\boldsymbol{K}}),$$

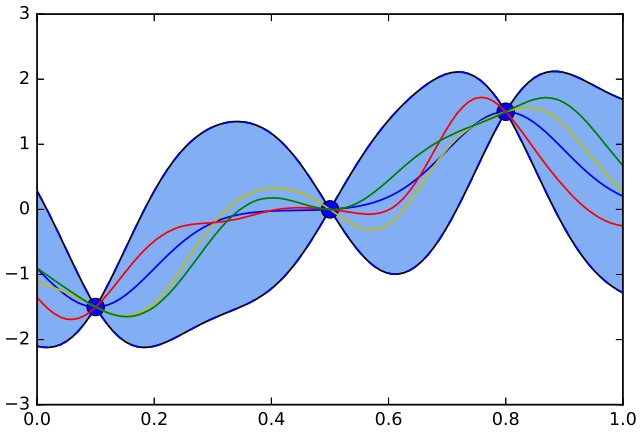


Рис. 3: Предсказательное распределение в задаче регрессии

где

$$\mathbb{E}[\boldsymbol{f}_*|\boldsymbol{y}] = \hat{\boldsymbol{m}} = K(\boldsymbol{X}_*, \boldsymbol{X})(K(\boldsymbol{X}, \boldsymbol{X}) + \nu^2 \boldsymbol{I})^{-1}\boldsymbol{y},$$

$$\text{cov}(\boldsymbol{f}_*|\boldsymbol{y}) = \hat{\boldsymbol{K}} = K(\boldsymbol{X}_*, \boldsymbol{X}_*) - K(\boldsymbol{X}_*, \boldsymbol{X})(K(\boldsymbol{X}, \boldsymbol{X}) + \nu^2 \boldsymbol{I})^{-1}K(\boldsymbol{X}, \boldsymbol{X}_*).$$

Таким образом, сложность получения распределения на $\boldsymbol{f}_*$ определяется сложностью обращения матрицы $K(\boldsymbol{X}, \boldsymbol{X}) + \nu^2 \boldsymbol{I} \in \mathbb{R}^{n \times n}$ и имеет асимптотику $\mathcal{O}(n^3)$.

На рис. 3 показан пример предсказательного распределения для гауссовского процесса с $\nu = 0$, настроенного по трем точкам, обозначенным синими кругами. Голубая область показывает 3σ-интервал для значений процесса, а темно-синяя линия — его матожидание. Цветные линии показывают случайные реализации процесса.

Более подробное описание методов для настройки модели регрессии на основе гауссовских процессов приводится в [11].

1.3 Модель для классификации

Рассмотрим задачу классификации. В этом случае данные представляют из себя матрицу $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n)^T \in \mathbb{R}^{n \times D}$, а значения целевой переменной $\boldsymbol{y} = (y_1, y_2, \ldots, y_n)^T \in \{1, 2, \ldots, C\}^n$, где C — число классов. В данном разделе мы кратко

опишем задачу бинарной классификации ($C = 2$), случай с произвольным числом классов подробно разбирается в разделе 2.2.

Вероятностная модель для бинарной классификации совпадает с (1), с тем отличием, что используется другое распределение $p(y_i|f_i)$:

$$p(y_i = 1|f_i) = \sigma(f_i),$$

где $\sigma(\cdot)$ — некоторая сигмоидная функция. Для определенности, можно считать, что

$$\sigma(z) = (1 + \exp(-z))^{-1}.$$

Графическая модель для задачи бинарной классификации в точности совпадает с моделью для регрессии, и показана на рисунке 2.

Для получения предсказания в новом наборе точек $\boldsymbol{X}_*$, требуется вычислить

$$p(\boldsymbol{f}_*|\boldsymbol{y}) = \int p(\boldsymbol{f}_*|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{y})d\boldsymbol{f}.$$

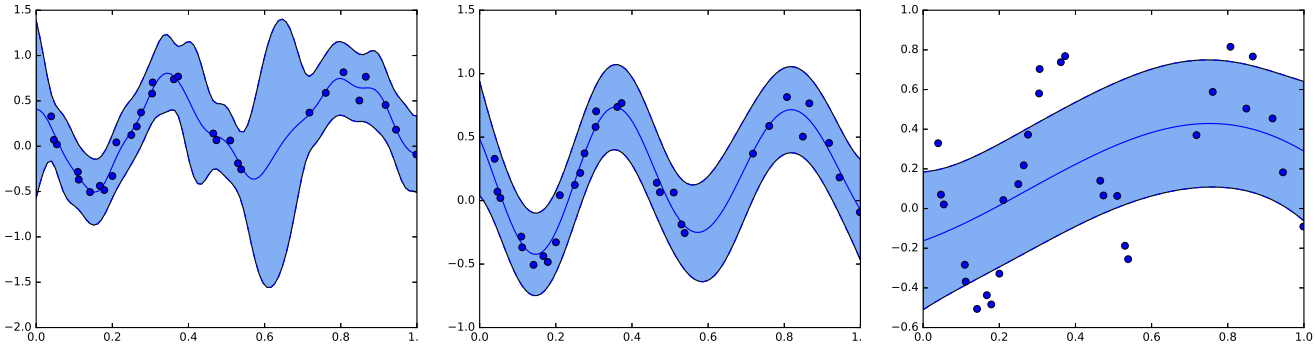
Далее, для оценки вероятности принадлежности объектов $\boldsymbol{X}_*$ положительному классу, необходимо вычислить интеграл

$$p(\boldsymbol{y}_* = +1|\boldsymbol{y}) = \int \sigma(\boldsymbol{f}_*)p(\boldsymbol{f}_*|\boldsymbol{y})d\boldsymbol{f}_*.$$

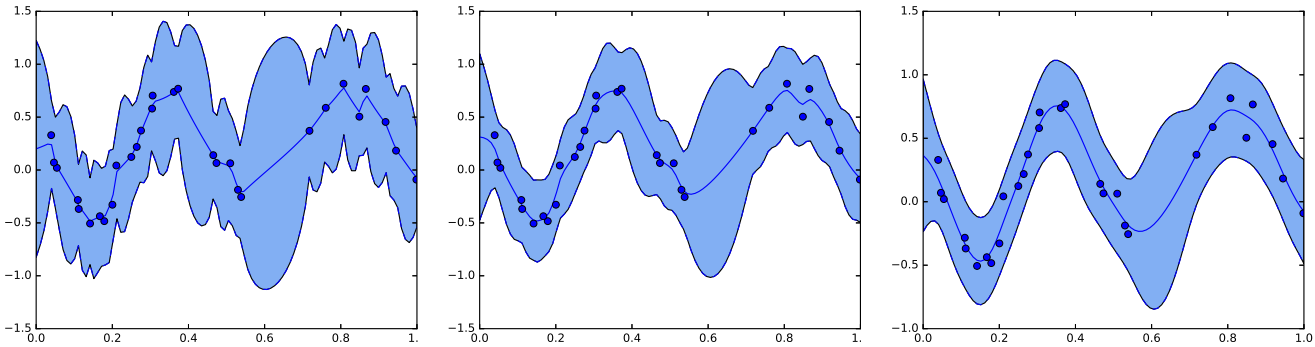
К сожалению, оба эти интеграла не берутся аналитически, поэтому приходится прибегать к приближенным методам их вычисления. Мы не будем здесь подробно останавливаться на приближенных схемах для вывода в данной модели, так как ниже будут рассмотрены более эффективные методы для решения задачи классификации на основе гауссовских процессов. Подробное описание приближенных схем для вывода в данной модели можно найти в [11].

1.4 Адаптация модели под данные

Выше мы рассматривали гауссовские процессы с фиксированной ядровой функцией. Как правило, чтобы модель на основе гауссовских процессов хорошо описывала данные, ее ковариационную функцию настраивают по ним.



(a) SE, $l = 0.05$ (b) SE, $l = 0.15$ (c) SE, $l = 0.5$



(d) Matern, $\nu = 0.05$ (e) Matern, $\nu = 1$ (f) Matern, $\nu = 2$

Рис. 4: Гауссовские процессы с ядрами SE и Matern, восстановленные по одним и тем же данным для разных значений параметров ядер

Большинство популярных ядровых функций имеет набор параметров, которые мы будем называть параметрами ядра. Например, ядро Squared Exponential (SE)

$$k_{SE}(\boldsymbol{x}, \boldsymbol{x}') = \sigma^2 \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{l^2}\right)$$

имеет два параметра — σ и l. Примером более сложной популярной ядровой функции является функция Matern

$$k_{Matern}(\boldsymbol{x}, \boldsymbol{x}') = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}\|\boldsymbol{x} - \boldsymbol{x}'\|}{l}\right)^{\nu} K_\nu \left(\frac{\sqrt{2\nu}}{\|\boldsymbol{x} - \boldsymbol{x}'\|} l\right),$$

с двумя неотрицательными параметрами ν и l. Здесь K_ν это модифицированная функция Бесселя.

На рисунке 4 показаны примеры гауссовских процессов, обученных на одних и тех же данных для разных ядер. Можно заметить, что при неправильном выборе параметров ядра процесс получается либо переобученным, либо недообученным.

Стандартным подходом для настройки гиперпараметров модели в Байесовской парадигме является максимизация обоснованности модели

$$p(\boldsymbol{y}) = \int p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f})d\boldsymbol{f}$$

по этим параметрам.

В случае регрессии возможно аналитически вычислить обоснованность. Обозначим через $\boldsymbol{\theta}$ параметры ядровой функции, а через $K_{\boldsymbol{\theta}}(\cdot,\cdot)$ — ядровую функцию, соответствующую заданному значению $\boldsymbol{\theta}$. Тогда

$$\log p(\boldsymbol{y}) = -\frac{1}{2}\boldsymbol{y}^T(K_{\boldsymbol{\theta}}(\boldsymbol{X},\boldsymbol{X}) + \nu^2\boldsymbol{I})^{-1}\boldsymbol{y} - \frac{1}{2}\log|K_{\boldsymbol{\theta}}(\boldsymbol{X},\boldsymbol{X}) + \nu^2\boldsymbol{I}| - \frac{n}{2}\log 2\pi.$$

Далее, обоснованность можно максимизировать по параметрам $\boldsymbol{\theta}$ и дисперсии шума ν.

Для задачи классификации обоснованность аналитически подсчитать невозможно, но существуют методы, позволяющие ее оценить (см. [11]).

Сложность вычисления обоснованности имеет ассимптотику $\mathcal{O}(n^3)$, как для задачи регрессии, так и для задачи классификации.

1.5 Расширенная модель на основе вспомогательных точек

Как было показано выше, методы для обучения стандартных моделей на основе гауссовских процессов в задачах регрессии и классификации имеют сложность $\mathcal{O}(n^3)$, где n — размер выборки. Это обстоятельство не позволяет использовать стандартные модели в случае, когда размер выборок превосходит несколько тысяч объектов.

В литературе было предложено множество приближенных методов. В данной работе рассматриваются методы на основе так называемых вспомогательных точек

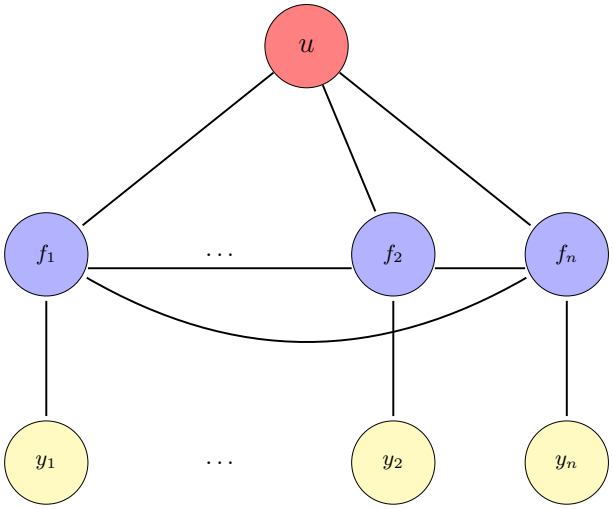


Рис. 5: Графическая модель для модели регрессии (классификаци) на основе гауссовских процессов на основе вспомогательных точек

(inducing inputs). Эти методы строят приближение на основе значений процесса в $m < n$ точках, которые и называются вспомогательными. Первые методы данного класса брали в качестве вспомогательных точек подмножество обучающей выборки, которое выбиралось с помощью того или иного информационного критерия. Подробный обзор методов данного класса можно найти, например, в [12].

В работе [1] было предложено трактовать значения процесса во вспомогательных точках как латентные переменные модели. Пусть $\boldsymbol{Z} = (\boldsymbol{z}_1, \boldsymbol{z}_2, \dots, \boldsymbol{z}_m)^T \in \mathbb{R}^{m \times D}$ — позиции вспомогательных точек в признаковом пространстве. Пусть $\boldsymbol{u} = (u_1, u_2, \dots, u_m)^T$ — соответствующие им значения гауссовского процесса. Введем расширенную модель

$$p(\boldsymbol{y}, \boldsymbol{f}, \boldsymbol{u}) = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{u})p(\boldsymbol{u}) = \prod_{i=1}^{n} p(y_i|f_i)p(\boldsymbol{f}|\boldsymbol{u})p(\boldsymbol{u}). \tag{2}$$

Соответствующая графическая модель приводится на рисунке 5.

Введем для краткости записей обозначения

$$\boldsymbol{K}_{nm} = K(\boldsymbol{X}, \boldsymbol{Z}), \quad \boldsymbol{K}_{mn} = K(\boldsymbol{Z}, \boldsymbol{X}) = \boldsymbol{K}_{nm}^T, \quad \boldsymbol{K}_{nn} = K(\boldsymbol{X}, \boldsymbol{X}), \quad \boldsymbol{K}_{mm} = K(\boldsymbol{Z}, \boldsymbol{Z}).$$

Тогда, так как $\boldsymbol{u}$ — значения гауссовского процесса $\mathcal{GP}(0, k(\cdot,\cdot))$ в точках $\boldsymbol{Z}$, а $\boldsymbol{f}$ — в точках $\boldsymbol{X}$, то

$$p(\boldsymbol{u}) = \mathcal{N}(\boldsymbol{u}|\boldsymbol{0}, \boldsymbol{K}_{mm}),$$

$$p(\boldsymbol{f}|\boldsymbol{u}) = \mathcal{N}(\boldsymbol{f}|\boldsymbol{K}_{nm}\boldsymbol{K}_{mm}^{-1}\boldsymbol{u}, \boldsymbol{K}_{nn} - \boldsymbol{K}_{nm}\boldsymbol{K}_{mm}^{-1}\boldsymbol{K}_{mn}).$$

Как и выше, распределение $p(y_i|f_i)$ зависит от решаемой задачи. В случае регрессии

$$p(y_i|f_i) = \mathcal{N}(y_i|f_i, \nu^2).$$

Таким образом, расширенная модель полностью задана. Заметим, что маргинальное распределение на переменные $\boldsymbol{y}$ и $\boldsymbol{f}$ совпадает с распределением из стандартной модели (1).

Применим технику вариационного вывода для оценки апостериорного распределения на скрытые переменные $q(\boldsymbol{f}, \boldsymbol{u}) \approx p(\boldsymbol{f}, \boldsymbol{u}|\boldsymbol{y})$. Подробное описание этой техники можно найти в [13]. Стандартная вариационная нижняя оценка на обоснованность модели

$$\log p(\boldsymbol{y}) \geq \mathbb{E}_{q(\boldsymbol{u},\boldsymbol{f})} \log \frac{p(\boldsymbol{y}, \boldsymbol{u}, \boldsymbol{f})}{q(\boldsymbol{u}, \boldsymbol{f})} = \mathbb{E}_{q(\boldsymbol{u},\boldsymbol{f})} \log p(\boldsymbol{y}|\boldsymbol{f}) - \mathrm{KL}(q(\boldsymbol{u}, \boldsymbol{f})||p(\boldsymbol{u}, \boldsymbol{f})), \qquad (3)$$

где KL — дивергенция Кульбака-Лейблера.

Будем искать вариационное распределение $q(\boldsymbol{u}, \boldsymbol{f})$ в виде

$$q(\boldsymbol{u}, \boldsymbol{f}) = p(\boldsymbol{f}|\boldsymbol{u})q(\boldsymbol{u}),$$

где $q(\boldsymbol{u}) = \mathcal{N}(\boldsymbol{u}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ — нормальное распределение с некоторыми параметрами $\boldsymbol{\mu} \in \mathbb{R}^m$ и $\boldsymbol{\Sigma} \in \mathbb{R}^{m\times m}$. Тогда дивергенция Кульбака-Лейблера

$$\mathrm{KL}(q(\boldsymbol{u}, \boldsymbol{f})||p(\boldsymbol{u}, \boldsymbol{f})) = \mathrm{KL}(p(\boldsymbol{f}|\boldsymbol{u})q(\boldsymbol{u})||p(\boldsymbol{f}|\boldsymbol{u})p(\boldsymbol{u})) = \mathrm{KL}(q(\boldsymbol{u})||p(\boldsymbol{u})). \qquad (4)$$

Маргинальное распределение

$$q(\boldsymbol{f}) = \int q(\boldsymbol{u}, \boldsymbol{f}) d\boldsymbol{u} = \mathcal{N}(\boldsymbol{f}|\boldsymbol{K}_{nm}\boldsymbol{K}_{mm}^{-1}\boldsymbol{\mu}, \boldsymbol{K}_{nn} + \boldsymbol{K}_{nm}\boldsymbol{K}_{mm}^{-1}(\boldsymbol{\Sigma} - \boldsymbol{K}_{mm})\boldsymbol{K}_{mm}^{-1}\boldsymbol{K}_{mn}). \quad (5)$$

С учетом (4) и (5) можем переписать оценку (3) в виде

$$\log p(\boldsymbol{y}) \geq \sum_{i=1}^{n} \mathbb{E}_{q(f_i)} \log(p(y_i|f_i)) - \mathrm{KL}(q(\boldsymbol{u})||p(\boldsymbol{u})). \tag{6}$$

Дивергенция Кульбака-Лейблера между двумя нормальными распределениями $q(\boldsymbol{u})$ и $p(\boldsymbol{u})$ вычисляется аналитически и имеет вид

$$\mathrm{KL}(q(\boldsymbol{u})||p(\boldsymbol{u})) = \frac{1}{2}\left(\log\frac{|\boldsymbol{K}_{mm}|}{|\boldsymbol{\Sigma}|} - m + \mathrm{tr}(\boldsymbol{K}_{mm}^{-1}\boldsymbol{\Sigma}) + \boldsymbol{\mu}^T\boldsymbol{K}_{mm}^{-1}\boldsymbol{\mu}\right).$$

Сложность вычисления этого члена в оценке (6) имеет асимптотику $\mathcal{O}(m^3)$.

В случае задачи регрессии первый член в оценке (6) также можно вычислить аналитически. В этом случае оценка окончательно принимает вид

$$\begin{aligned}\log p(\boldsymbol{y}) \geq \sum_{i=1}^{n}\left(\log\mathcal{N}(y_i|\boldsymbol{k}_i^T\boldsymbol{K}_{mm}^{-1}\mu, \nu^2) - \frac{1}{2\nu^2}\tilde{\boldsymbol{K}}_{ii} - \frac{1}{2\nu^2}\mathrm{tr}(\boldsymbol{k}_i^T\boldsymbol{K}_{mm}^{-1}\boldsymbol{\Sigma}\boldsymbol{K}_{mm}^{-1}\boldsymbol{k}_i)\right) - \\ -\frac{1}{2}\left(\log\frac{|\boldsymbol{K}_{mm}|}{|\boldsymbol{\Sigma}|} - m + \mathrm{tr}(\boldsymbol{K}_{mm}^{-1}\boldsymbol{\Sigma}) + \boldsymbol{\mu}^T\boldsymbol{K}_{mm}^{-1}\boldsymbol{\mu}\right),\end{aligned} \tag{7}$$

где $\boldsymbol{k}_i = K(\boldsymbol{x}_i, \boldsymbol{Z}) \in \mathbb{R}^m$ — i-й столбец матрицы $\boldsymbol{K}_{mn}$,

$$\tilde{\boldsymbol{K}} = \boldsymbol{K}_{nn} - \boldsymbol{K}_{nm}\boldsymbol{K}_{mm}^{-1}\boldsymbol{K}_{mn}.$$

Таким образом, в случае задачи регрессии сложность вычисления оценки (7) имеет асимптотику $\mathcal{O}(nm^2 + m^3)$. Как правило, используют $m \ll n$, в результате чего сложность метода получается существено ниже, чем у стандартных методов.

Оценку (7) максимизируют по вариационным параметрам $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$, по параметрам ядровой функции $\boldsymbol{\theta}$ и по дисперсии шума ν. Также, можно производить оптимизацию по позициям вспомогательных точек $\boldsymbol{Z}$, однако на практике часто фиксируют этот параметр. Например, в качестве точек $\boldsymbol{Z}$ можно взять центры кластеров, полученных запуском метода K-means на m компонент. В работе [1] максимизация оценки (7) по вариационным параметрам $\boldsymbol{\mu}$ и $\boldsymbol{\Sigma}$ проводится аналитически, а по оставшимся параметрам ведется численная оптимизация. В работе [2] предлагается производить настройку всех параметров модели методом стохастического градиента.

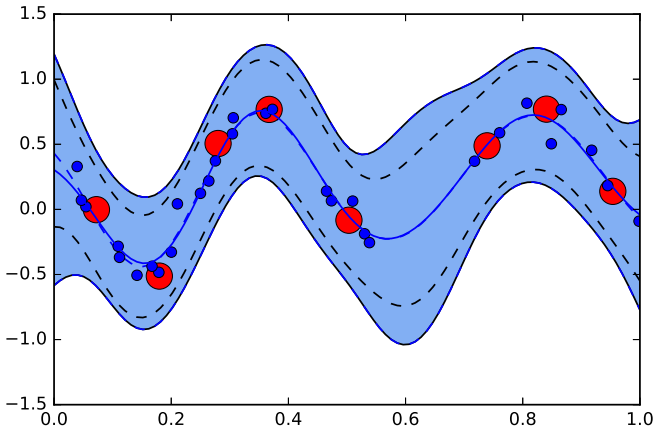


Рис. 6: Приближенное предсказательное распределение в модели регрессии с помощью гауссовских процессов со вспомогательными точками

В случае бинарной классификации в явном виде получить оценку на обоснованность не удается, однако для оценки одномерных интегралов

$$\mathbb{E}_{q(f_i)} \log p(y_i|f_i)$$

можно применять квадратуры Гаусса-Эрмита, или строить квадратичные оценки подынтегральной функции. В работе [14] предложен ряд методов данного типа, а также проводится их подробное экспериментальное сравнение.

На этапе предсказания значения процесса в новых точках $\boldsymbol{X}_*$ можно воспользоваться вариационным приближением к апостериорному распределению

$$p(\boldsymbol{f}_*|\boldsymbol{y}) = \int p(\boldsymbol{f}_*|\boldsymbol{f},\boldsymbol{u})p(\boldsymbol{f},\boldsymbol{u}|\boldsymbol{y})d\boldsymbol{u}d\boldsymbol{f} \approx \int p(\boldsymbol{f}_*|\boldsymbol{f},\boldsymbol{u})q(\boldsymbol{f},\boldsymbol{u})d\boldsymbol{u}d\boldsymbol{f} =$$

$$= \int p(\boldsymbol{f}_*,\boldsymbol{f}|\boldsymbol{u})q(\boldsymbol{u})d\boldsymbol{u}d\boldsymbol{f} = \int p(\boldsymbol{f}_*|\boldsymbol{u})q(\boldsymbol{u})d\boldsymbol{u}.$$

На рис. 6 показано предсказательное распределение, полученное с помощью вспомогательных точек. Здесь синие точки — точки обучающей выборки, красные — вспомогательные точки (по вертикали отложено матожидание $\boldsymbol{\mu}$ процесса в этих точках). Синяя пунктирная линия изображает матожидание процесса, восстановленного по обучающей выборке для стандартной модели (см. 1.2), а темно-синяя линия

обозначает матожидание процесса полученного с помощью вспомогательных точек. Светло-голубая область показывает 3σ-интервал для значений процесса. Черными пунктирными линиями обозначены границы 3σ-интервала для полного гауссовского процесса восстановленного по данным.

1.6 Метод KISS-GP

В предыдущем разделе мы подробно описали методы на основе вспомогательных точек. Сложность вычисления оценки на обоснованность модели для этих методов имеет асимптотику $\mathcal{O}(nm^2+m^3)$. Данный класс методов позволяет решать задачи регрессии и классификации для больших выборок данных. Однако, число m вспомогательных точек при этом не может быть слишком большим, что сильно ограничивает экспрессивность итоговой модели.

Метод KISS-GP (см. [4]) позволяет использовать большое число вспомогательных точек за счет использования эффективных операций с матрицами специального вида, а также интерполяции ядровой функции. Пусть используемая ядровая функция $k(\cdot,\cdot)$ раскладывается в произведение по размерностям

$$k(\boldsymbol{x},\boldsymbol{x}') = k^1(x^1,x'^1)\cdot k^2(x^2,x'^2)\cdot\ldots\cdot k^D(x^D,x'^D).$$

Пусть вспомогательные точки $\boldsymbol{Z}$ расположены на многомерной сетке в признаковом пространстве

$$\boldsymbol{Z} = \boldsymbol{Z}^1\times\boldsymbol{Z}^2\times\ldots\times\boldsymbol{Z}^D,$$

где $\boldsymbol{Z}^i\in\mathbb{R}^{m_i}$ для всех $i=1,2,\ldots,D$. Тогда матрица ковариации между вспомогательными точками представляется в виде произведения Кронекера.

$$\boldsymbol{K}_{mm} = \boldsymbol{K}^1_{m_1m_1}\otimes\boldsymbol{K}^2_{m_2m_2}\otimes\ldots\otimes\boldsymbol{K}^D_{m_Dm_D},$$

где

$$\boldsymbol{K}^i_{m_im_i} = K^i(\boldsymbol{Z}^i,\boldsymbol{Z}^i),\quad m=\prod_{i=1}^D m_i.$$

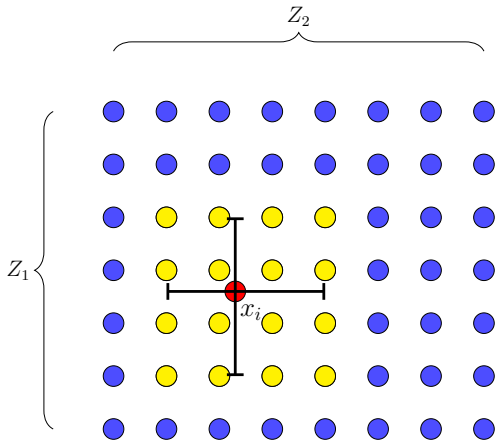


Рис. 7: Ненулевые коэффициенты в интерполяции

Произведения Кронекера обратимых матриц можно эффективно обращать:

$$(\boldsymbol{A}_1 \otimes \boldsymbol{A}_2 \otimes \ldots \otimes \boldsymbol{A}_D)^{-1} = \boldsymbol{A}_1^{-1} \otimes \boldsymbol{A}_2^{-1} \otimes \ldots \otimes \boldsymbol{A}_D^{-1}.$$

Аналогично, определитель матрицы, представимой в виде произведения Кронекера квадратных матриц, вычисляется эффективно:

$$|\boldsymbol{A}_1 \otimes \boldsymbol{A}_2 \otimes \ldots \otimes \boldsymbol{A}_D| = |\boldsymbol{A}_1|^{c_1} \cdot |\boldsymbol{A}_2|^{c_2} \cdot \ldots \cdot |\boldsymbol{A}_D|^{c_D},$$

где

$$\boldsymbol{A}_i \in \mathbb{R}^{k_i \times k_i}, \quad c_i = \prod_{j \neq i} k_j, \quad i = 1, 2, \ldots, D.$$

Далее, вспомогательные точки $\boldsymbol{Z}$ можно рассматривать как узлы интерполяции для ковариационной функции $k(\cdot, \boldsymbol{z}_i)$. Тогда имеем приближение

$$\boldsymbol{K}_{nm} \approx \boldsymbol{W} \boldsymbol{K}_{mm}, \tag{8}$$

где матрица $\boldsymbol{W} \in \mathbb{R}^{n \times m}$ содержит веса интерполяции. В методе KISS-GP используется сверточная кубическая интерполяция (см. [15]). Для такой интерполяции строки W_i матрицы весов (то есть веса интерполяции для i-го объекта) представляются в виде произведения Кронекера по размерностям

$$\boldsymbol{w}_i = \boldsymbol{w}_i^1 \otimes \boldsymbol{w}_i^2 \otimes \ldots \otimes \boldsymbol{w}_i^D.$$

По каждой размерности k находятся 4 ближайшие к $\boldsymbol{x}_i$ вспомогательные точки $\hat{z}_1^k$, $\hat{z}_2^k$, $\hat{z}_3^k$, $\hat{z}_4^k \in \boldsymbol{Z}^k$. Соответствующие им координаты вектора $\boldsymbol{w}_i^k$ вычисляются по заданным формулам зависящим от разности $s = |\boldsymbol{x}_i^k - \hat{z}_j^k|$, а остальные координаты кладутся равными 0.

На рисунке 7 красным цветом отмечена точка x_i в двумерном признаковом пространстве. Желтым цветом показаны вспомогательные точки (узлы интерполяции), для которых вес интерполяции отличен от 0, а синим — остальные вспомогательные точки.

В оригинальной работе [4] предлагалось использовать аппроксимацию (8) в комбинации с методом SOR (см. [16]). Здесь мы не будем рассматривать подробно этот метод. Его сложность составляет $\mathcal{O}(n + m \log m)$, что существенно ниже сложности методов, основанных на вспомогательных точках, рассмотренных выше. Однако так как вспомогательные точки размещены на многомерной сетке в признаковом пространстве, то их число растет экспоненциально с размерностью D. Это обстоятельство делает метод KISS-GP неприменимым в случае, когда число признаков велико $D \gg 4$.

1.7 Разложение Tensor Train

Метод тензорного поезда (Tensor Train, TT), предложенный в работе [10], позволяет хранить многомерные массивы данных и большие матрицы или векторы в сжатом формате. При этом для матриц и векторов в TT-формате существую эффективные реализации операций линейной алгебры. Формат TT уже был эффективно использован в ряде методов машинного обучения (см. например [17], [18]).

Рассмотрим D-мерный тензор $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times \ldots \times n_D}$. Говорят, что $\mathcal{A}$ задан в TT-формате, если

$$\mathcal{A}(i_1, \ldots, i_d) = \boldsymbol{G}_1[i_1] \cdot \boldsymbol{G}_2[i_2] \cdot \ldots \cdot \boldsymbol{G}_D[i_D], \quad i_k \in \{1, \ldots, n_k\}, \tag{9}$$

где

$$\boldsymbol{G}_k[i_k] \in \mathbb{R}^{r_k \times r_{k+1}} \text{ для всех } k, i_k, \quad r_1 = r_{D+1} = 1.$$

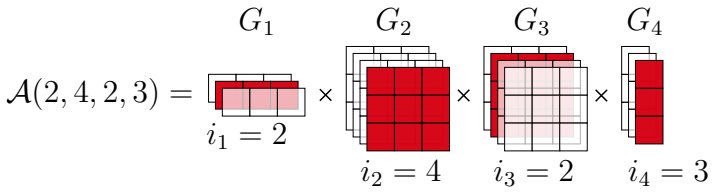


Рис. 8: Формат Tensor Train

Матрицы $\boldsymbol{G}_k$ называются TT-ядрами, а r_k — TT-рангами.

Для хранения векторов в формате TT предлагается преобразовать их в тензоры, при необходимости дополнив нулями, после чего использовать формат (9). Ниже мы будем использовать формат TT для вектора $\boldsymbol{\mu}$ матожиданий значений гауссовского процесса во вспомогательных точках, расположенных на D-мерной сетке в признаковом пространстве. В этом случае вектор $\boldsymbol{\mu}$ естественным образом представляется в виде тензора и, соответственно, в формате (9).

Для матриц представление TT имеет вид

$$\boldsymbol{M}(i_1, i_2, \ldots, i_D; j_1, j_2, \ldots, j_D) = \boldsymbol{G}_1[i_1, j_1] \cdot \boldsymbol{G}_2[i_2, j_2] \cdot \ldots \cdot \boldsymbol{G}_D[i_D, j_D], \tag{10}$$

где

$$\boldsymbol{G}_k[i_k, j_k] \in \mathbb{R}^{r_k \times r_{k+1}} \text{ для всех } k, i_k, j_k, \quad r_1 = r_{D+1} = 1.$$

Заметим, что произведение Кронекера — частный случай представления Tensor Train, соответствующий ситуации, когда все TT-ранги $r_1 = r_2 = \ldots = r_{D+1} = 1$.

Для векторов и матриц в формате TT возможно эффективное выполнение многих операций линейной алгебры. Рассмотрим подробнее те, которые потребуются нам ниже.

Пусть векторы $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^{n_1 \cdot n_2 \cdot \ldots \cdot n_D}$ представляются в формате TT с рангами не больше r.

$$\boldsymbol{u}(i_1, \ldots, i_d) = \boldsymbol{u}_1[i_1] \cdot \boldsymbol{u}_2[i_2] \cdot \ldots \cdot \boldsymbol{u}_D[i_D], \quad i_k \in \{1, \ldots, n_k\},$$

и аналогично для $\boldsymbol{v}$. Пусть матрицы $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{n_1 \cdot n_2 \cdot \ldots \cdot n_D \times n_1 \cdot n_2 \cdot \ldots \cdot n_D}$ представляется в виде произведения Кронекера

$$\boldsymbol{A} = \boldsymbol{A}_1 \otimes \boldsymbol{A}_2 \otimes \ldots \otimes \boldsymbol{A}_D, \quad \boldsymbol{A}_k \in \mathbb{R}^{n_k \times n_k},$$

и аналогично для $\boldsymbol{B}$. Пусть $n = \max_k n_k$. Тогда сложность вычисления квадратичной формы $\boldsymbol{u}^T \boldsymbol{A}\boldsymbol{v}$ имеет асимптотику $\mathcal{O}(Dnr^3)$. Сложность операции $\mathrm{tr}(\boldsymbol{A}\boldsymbol{B})$ имеет асимптотику $\mathcal{O}(Dn^2)$.

2 Предлагаемый метод TT-GP

В разделе 1 приводится описание ряда методов для задач регрессии и классификации с помощью гауссовских процессов. Все описанные методы страдают от тех или иных недостатков. Стандартные методы не применимы для больших выборок, метод KISS-GP не может работать с признаковыми пространствами больших размерностей, а другие методы на основе вспомогательных точек не могут восстанавливать сложные закономерности в данных, так как количество вспомогательных точек m в них не может быть слишком большим. В данном разделе описывается новый метод для обучения моделей регрессии и классификации, способный работать с выборками больших размеров, и большим количеством признаков. Кроме того, предлагается способ построения экспрессивных ядровых функций на основе нейронных сетей, позволяющий успешно применять новый метод к задачам с признаковыми пространствами, в которых евклидово расстояние не является хорошей мерой схожести объектов (например, к задачам компьютерного зрения).

2.1 Аппроксимация вариационных параметров

Рассмотрим задачу регрессии на основе гауссовских процессов. Пусть вспомогательные точки $\boldsymbol{Z}$ размещены на решетке в многомерном признаковом пространстве (см. раздел 1.6). Пусть по каждой размерности в сетке m_0 точек, то есть

$$m = m_0^D.$$

В разделе 1.5 была выведена вариационная нижняя оценка на правдоподобие (7). Подставим в эту оценку аппроксимацию (8), предложенную в методе KISS-GP

$$\boldsymbol{K}_{nm} \approx \boldsymbol{W}\boldsymbol{K}_{mm}, \quad \boldsymbol{k}_i \approx \boldsymbol{w}_i \boldsymbol{K}_{mm}.$$

Получим

$$\log p(\boldsymbol{y}) \geq \sum_{i=1}^{n} \left(\log \mathcal{N}(y_i | \boldsymbol{w}_i^T \boldsymbol{\mu}, \nu^2) - \frac{1}{2\nu^2} \tilde{\boldsymbol{K}}_{ii} - \frac{1}{2\nu^2} \mathrm{tr}(\boldsymbol{w}_i^T \boldsymbol{\Sigma} \boldsymbol{w}_i) \right) - \\ - \frac{1}{2} \left(\log \frac{|\boldsymbol{K}_{mm}|}{|\boldsymbol{\Sigma}|} - m + \mathrm{tr}(\boldsymbol{K}_{mm}^{-1} \boldsymbol{\Sigma}) + \boldsymbol{\mu}^T \boldsymbol{K}_{mm}^{-1} \boldsymbol{\mu} \right), \tag{11}$$

где

$$\tilde{\boldsymbol{K}}_{ii} = K(\boldsymbol{x}_i, \boldsymbol{x}_i) - \boldsymbol{w}_i^T \boldsymbol{K}_{mm} \boldsymbol{w}_i.$$

Пусть используется ядровая функция, которая представляется в виде произведения по размерностям. Тогда

$$\boldsymbol{K}_{mm} = \boldsymbol{K}^1_{m_0 m_0} \otimes \boldsymbol{K}^2_{m_0 m_0} \otimes \ldots \otimes \boldsymbol{K}^D_{m_0 m_0}.$$

Тогда обращение матрицы $\boldsymbol{K}_{mm}$ и подсчет ее определителя имеют сложность $\mathcal{O}(Dm_0^3) = \mathcal{O}(Dm^{3/D})$. Основную сложность при вычислении оценки (11) теперь составляют операции с вариационными параметрами $\boldsymbol{\mu}, \boldsymbol{\Sigma}$.

Сузим семейство вариационных распределений. Пусть матрица $\boldsymbol{\Sigma}$ представляется в виде произведения Кронекера по размерностям признакового пространства, а вектор $\boldsymbol{\mu}$ — в формате Tensor Train с TT-рангами не превосходящими r. Тогда, в соответствии с оценками из раздела 1.7, сложность вычисления оценки (11) имеет асимптотику $\mathcal{O}(nDm_0r^2 + Dm_0r^3 + Dm_0^3) = \mathcal{O}(nDm^{1/D}r^2 + Dm^{1/D}r^3 + Dm^{3/D})$. Оценку (11) можно максимизировать по параметрам ядровой функции $\boldsymbol{\theta}$, TT-ядрам вектора $\boldsymbol{\mu}$ и сомножителям в представлении матрицы $\boldsymbol{\Sigma}$ в виде произведения Кронекера. Тогда общее число оптимизируемых параметров составляет $\mathcal{O}(\sharp\boldsymbol{\theta} + dm_0r^2 + dm_0^2)$ или $\mathcal{O}(\sharp\boldsymbol{\theta} + dm^{1/D}r^2 + dm^{2/D})$, где $\sharp\boldsymbol{\theta}$ — число параметров ядровой функции.

Таким образом, предложенный подход имеет сложность линейно зависящую от размерности признакового пространства D, несмотря на экспоненциально возрастающее количество вспомогательных точек. Также, метод имеет низкую сложность от размера выборки n, а оценка (11) представляется в виде суммы по объектам, что делает возможным применение стохастической оптимизации для настройки параметров модели.

2.2 Обобщение на задачу многоклассовой классификации

Рассмотрим обобщение предложенного метода на задачу классификации на C классов. Пусть, как и выше, $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n)^T \in \mathbb{R}^{n \times D}$ — признаковые описания объектов обучающей выборки, а $\boldsymbol{y} = (y_1, y_2, \ldots, y_n) \in \{1, 2, \ldots, C\}^n$ — значения целевой переменной.

Рассмотрим C гауссовских процессов, действующих из признакового пространства $\mathbb{R}^D$. Каждый процесс соответствует своему классу. Рассмотрим расширенную модель (см. раздел 1.5). Вспомогательные точки $\boldsymbol{Z}$ разместим на многомерной сетке в признаковом пространстве, и сделаем общими для всех процессов. Для каждого класса c имеем свой набор скрытых переменных — значения процесса в точках обучающей выборки $\boldsymbol{f}^c \in \mathbb{R}^n$, и во вспомогательных точках $\boldsymbol{u}^c \in \mathbb{R}^m$.

В качестве распределения $p(y_i|\boldsymbol{f}_i^{1\ldots C})$ возьмем дискретное распределение с вероятностями

$$p(y = c|\boldsymbol{f}^{1\ldots C}) = \frac{\exp(f^c)}{\sum_{j=1}^{C} \exp(f^j)}, \quad c = 1, 2, \ldots, C.$$

Вариационное распределение будем искать в факторизованном по процессам виде

$$q(\boldsymbol{f}^1, \boldsymbol{f}^2, \ldots, \boldsymbol{f}^C, \boldsymbol{u}^1, \boldsymbol{u}^2, \ldots, \boldsymbol{u}^C) = q(\boldsymbol{f}^1, \boldsymbol{u}^1) \cdot q(\boldsymbol{f}^2, \boldsymbol{u}^2) \cdot \ldots \cdot q(\boldsymbol{f}^C, \boldsymbol{u}^C),$$

где

$$q(\boldsymbol{f}^c, \boldsymbol{u}^c) = p(\boldsymbol{f}^c|\boldsymbol{u}^c)\mathcal{N}(\boldsymbol{u}^c|\boldsymbol{\mu}^c, \boldsymbol{\Sigma}^c), \quad c = 1, \ldots, C.$$

В таком случае по аналогии с (6) получим вариационную нижнюю оценку

$$\log p(\boldsymbol{y}) \geq \sum_{i=1}^{n} \mathbb{E}_{q(\boldsymbol{f}_i^{1\ldots C})} \log(p(y_i|\boldsymbol{f}_i^{1\ldots C})) - \sum_{c=1}^{C} \mathrm{KL}(q(\boldsymbol{u}^c)||p(\boldsymbol{u}^c)). \tag{12}$$

Второе слагаемое в оценке (12) вычисляется аналитически, как сумма дивергенций Кульбака-Лейблера между нормальными распределениями. Первое же слагаемое не может быть вычислено аналитически. Пусть объект y_i имеет класс c. Тогда можно переписать

$$\mathbb{E}_{q(\boldsymbol{f}_i^{1\ldots C})} \log(p(y_i|\boldsymbol{f}_i^{1\ldots C})) = \mathbb{E}_{q(f_i^c)} f_i^c - \mathbb{E}_{q(\boldsymbol{f}_i^{1\ldots C})} \log\left(\sum_{j=1}^{C} \exp f_i^j\right), \tag{13}$$

где

$$q(\boldsymbol{f}_i^{1\ldots C}) = q(f_i^1) \cdot q(f_i^2) \cdot \ldots \cdot q(f_i^C),$$

и все распределения $q(f_i^c)$ нормальные. Очевидно, первое слагаемое в (13) вычисляется аналитически. Рассмотрим подробнее второе слагаемое. Для удобства обозначений не будем писать индекс i. Требуется оценить

$$\mathbb{E}_{q(\boldsymbol{f}^{1\ldots C})} \log \left(\sum_{j=1}^{C} \exp(f^j) \right), \quad q(f^j) = \mathcal{N}(f^j | m_j, s_j^2), \tag{14}$$

где параметры m_j и s_j вычисляются по формулам, аналогичным (5).

В работе [19] приводится сравнение нескольких оценок для функционала вида (14). Ниже выводится одна из этих оценок, которая используется в предлагаемом методе.

Воспользуемся вогнутостью логарифма, и выпишем его линеаризацию в точке $\dfrac{1}{\varphi}$:

$$\log \left(\sum_{j=1}^{C} \exp(f^j) \right) \leq \log \frac{1}{\varphi} + \varphi \left(\sum_{j=1}^{C} \exp(f^j) - \frac{1}{\varphi} \right) = \varphi \sum_{j=1}^{C} \exp(f^j) - \log \varphi - 1.$$

Взяв матожидание по распределению $q(\boldsymbol{f}^{1\ldots C})$ и промаксимизировав результат по φ окончательно получим

$$\mathbb{E}_{q(\boldsymbol{f}^{1\ldots C})} \log \left(\sum_{j=1}^{C} \exp(f^j) \right) \leq \log \left(\sum_{j=1}^{C} \exp \left(m_j + \frac{1}{2} s_j^2 \right) \right).$$

Подставляя полученную оценку в (12) получим нижнюю оценку на правдоподобие в замкнутом виде. Как и в случае регрессии, будем использовать формат произведения Кронекера для матриц $\boldsymbol{\Sigma}^c$, и формат TT для векторов $\boldsymbol{\mu}^c$. Сложность метода возрастает в C раз по сравнению с задачей регрессии.

2.3 Обучение ядер на основе нейронных сетей

Популярные ядровые функции для гауссовских процессов (см. например раздел 1.4) используют евклидово растояние между аргументами как меру их схожести.

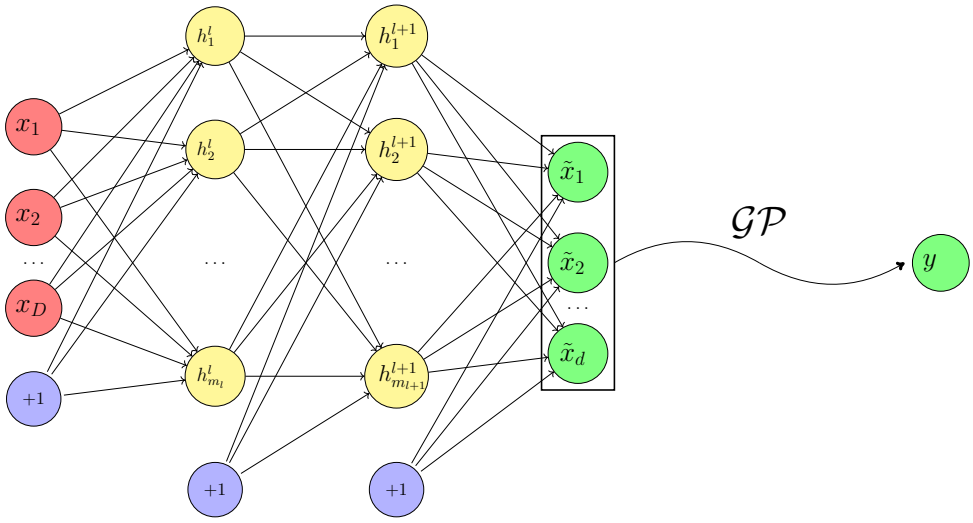


Рис. 9: Гауссовский процесс с ядром на основе нейронной сети

Во многих задачах (например, компьютерного зрения, обработки звука и видео) евклидово расстояние в признаковом пространстве плохо подходит для сравнения объектов. В данном разделе описывается метод построения экспрессивных ядровых функций на основе нейронных сетей, и настройки их параметров. Подобный подход использовался в методе SVDKL, предложенном в работе [6], однако в несколько ином виде. В SVDKL используются одномерные гауссовские процессы, обученные на нейронах последнего слоя нейронной сети, так как авторы брали за основу своего подхода метод KISS-GP, и не могли обучать многомерные процессы. Мы же за счет приближений, предложенных в разделе 2.1, можем обучать многомерные гауссовские процессы, используя все выходы нейронной сети в качестве признаков.

Рассмотрим некоторое параметрическое преобразование

$$\varphi(\cdot, \boldsymbol{\eta}) : \mathbb{R}^D \to \mathbb{R}^d,$$

где $\boldsymbol{\eta}$ — вектор значений параметров. Пусть задана ковариационная функция $k(\cdot, \cdot)$. Тогда можем определить новое ядро

$$k_\varphi(\boldsymbol{x}, \boldsymbol{x}') = k(\varphi(\boldsymbol{x}, \boldsymbol{\eta}), \varphi(\boldsymbol{x}', \boldsymbol{\eta})).$$

В качестве преобразования φ можно взять, например, нейронную сеть (см. рис. 9), или линейное преобразование.

Параметры построенного таким образом ядра (параметры $\boldsymbol{\theta}$ исходной ковариационной функции $k(\cdot,\cdot)$ и $\boldsymbol{\eta}$) можно настраивать с помощью максимизации оценки на обоснованность (6). При этом преобразование φ будет выучивать подходящее представление для данных, а непосредственно предсказание будет осуществлять гауссовский процесс.

3 Вычислительные эксперименты

В данном разделе приводится описание экспериментов по исследованию свойств предложенных методов, а также их сравнение с различными аналогами.

3.1 Набор данных Airline[1]

В данном разделе приводится экспериментальное исследование метода TT-GP с использованием стандартных ядровых функций (без нейронных сетей).

Для экспериментов был выбран набор данных Airline, состоящий из $n = 5\,934\,530$ записей о всех полетах коммерческих авиалиний в США за 2008 год. Данные содержат 8 признаков таких как день недели, расстояние между пунктами маршрута и т.п. Производится классификация на 2 класса — требуется предсказать, будет ли отложен рейс. Набор данных airline хорошо подходит для данного эксперимента, потому что он имеет не слишком большую размерность признакового пространства, и состоит из очень большого числа точек. Для получения данных использовались материалы[2] к статье [6], выложенные в открытом доступе.

На данный момент лидирующим методом для классификации с помощью гауссовских процессов в задачах с большими выборками является KLSP-GP, предложенный

[1] http://stat-computing.org/dataexpo/2009/

[2] https://people.orie.cornell.edu/andrew/code/#SVDKL

в работе [3]. Реализация данного метода не выложена в открытом доступе, поэтому нами были взяты результаты эксперимента, представленные в самой статье. В работе результат приводится в виде графика, и точные цифры для точности предсказания восстановить невозможно (поэтому в таблице приводится интервал). В качестве ядровой функции для метода KLSP-GP в данном эксперименте используется комбинация функции Matern и линейного ядра. Также для сравнения мы приводим результаты обучения линейной модели на этом наборе данных, описанные в работе [3].

Для метода TT-GP в качестве ядровой функции использовалось ядро SE (см. раздел 1.4). Вспомогательные точки были размещены на сетке в признаковом пространстве, по каждой размерности было взято $m_0 = 12$ точек. Таким образом, общее число вспомогательных точек превосходит $4 \cdot 10^8$. Отметим, что для существующих методов использование более 10^4 вспомогательных точек как правило невозможно.

Набор данных	Точность предсказания на тестовой выборке		
	Линейная модель	KLSP-GP	TT-GP
Airline ($n = 6M, d = 8$)	0.63	0.662–0.665	0.682

Таблица 1: Сравнение метода с KLSP

Результаты эксперимента приводятся в таблице 1. Из таблицы можно видеть, что за счет использования большего числа вспомогательных точек метод TT-GP существенно опережает KLSP-GP, несмотря на использование более простой ядровой функции. Отметим, что для достижения указанного качества методу TT-GP потребовалось сделать два прохода по данным.

3.2 Обучение представления данных

Рассмотрим теперь метод DNN-TT-GP, представляющий из себя метод TT-GP с ядровой функцией, построенной на основе нейронных сетей по принципам, описанным в разделе 2.3.

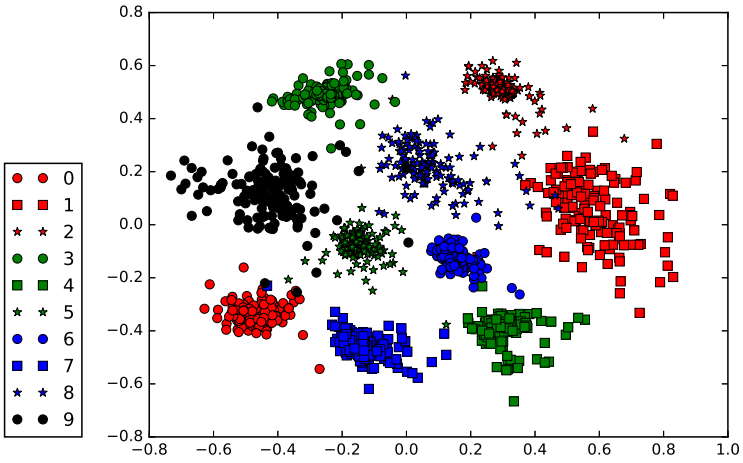


Рис. 10: Представление набора данных Digits, выучиваемое моделью

Сначала рассмотрим представление, которое выучивает нейронная сеть для данных. Эксперимент проводился на наборе данных Digits[3], состоящем их $n = 1797$ изображений цифр 8×8 пикселей. Для преобразования признаков мы использовали нейронную сеть с двумя скрытыми слоями по 50 нейронов, и 2 нейронами в выходном слое. Модель была обучена на задаче классификации на 10 классов.

На рис. 10 показано представление для данных, которое было выучено нейронной сетью. Из рисунка видно, что модель научилась группировать объекты, соответствующие одному классу, в компактных областях.

3.3 Задачи классификации изображений

В данном разделе приводятся результаты экспериментов на задачах классификации изображений. Рассматривались наборы данных CIFAR10[4] и MNIST[5].

Набор данных MNIST состоит из $n = 60000$ изображений цифр 28×28 пикселей. Тестовая выборка состоит из 10000 объектов. Для обучения представления на этом

[3] http://scikit-learn.org/stable/auto_examples/datasets/plot_digits_last_image.html

[4] https://www.cs.toronto.edu/~kriz/cifar.html

[5] http://yann.lecun.com/exdb/mnist/

Набор данных	Точность предсказания на тестовой выборке			
	RF-DGP	SV-DKL	DNN	DNN-TT-GP
MNIST ($n = 60K, d = 784$)	0.9814	0.9920	0.993	0.994
CIFAR10 ($n = 50K, d = 3072$)	-	0.7704	0.91	0.908

Таблица 2: Эксперименты на задачах классификации изображений

наборе данных была использована нейронная сеть с двумя сверточными и двумя полносвязными слоями (соответственно 32 и 64 фильтра в сверточных слоях, 1024 и 4 нейрона в полносвязных слоях). Рассматривалась задача классификации на 10 классов.

Набор данных CIFAR10 содержит $n = 50000$ цветных изображений 32×32 пикселей, относящихся к 10 классам. Для обучения представления данных в эксперименте с CIFAR10 использовалась нейронная сеть с шестью сверточными слоями и двумя полносвязными слоями (соответственно $128, 128, 256, 256, 256, 256$ фильтров в сверточных слоях; $1536, 512, 7$ нейронов в полносвязных слоях).

Метод SV-DKL был предложен в работе [6]. Отметим, что в методе DNN-TT-GP мы используем более удачную архитектуру сети, за счет чего удается добиться существенного выигрыша в качестве на наборе данных CIFAR10. Также отметим, что в отличие от предложенного метода DNN-TT-GP метод SV-DKL требует двухэтапной процедуры обучения — сначала должна быть обучена отдельно нейронная сеть, и только потом происходит финальная настройка модели вместе с гауссовскими процессами.

Для сравнения мы также приводим результаты метода RF-DGP, предложенного в работе [8]. Данный метод не использует сверточных слоев, и поэтому проигрывает по качеству в задачах с изображениями.

Наконец мы приводим качество нейронных сетей DNN, которые использовались для обучения представления в ядре гауссовского процесса в методе DNN-TT-GP. Мы заменили в этих сетях последний слой (с числом нейронов d равным размерно-

сти получаемого представления) на линейный слой с числом выходов равным числу классов, после чего обучили нейронные сети стандартными методами.

Результаты экспериментов приводятся в таблице 2. Из таблицы видно, что метод TT-GP успешно решает поставленные задачи. На обоих наборах данных удалось получить качество выше, чем для любой другой модели на основе гауссовских процессов, рассмотренной в литературе. Отметим, что полученное качество для метода DNN-TT-GP сравнимо с лучшими результатами на этих наборах данных.

4 Заключение

В данной работе предложен новый масштабируемый метод TT-GP для обучения моделей на основе гауссовских процессов для задач регрессии и классификации. Предложенный метод позволяет использовать огромное число вспомогательных точек. В работе описывается экспримент на наборе данных Airline, содержащем 6 миллионов объектов. На этом наборе данных удалось обучить метод TT-GP с $4 \cdot 10^8$ вспомогательными точками, в то время как для существующих методов использование более 10^4 вспомогательных точек как правило невозможно. В результате удалось улучшить качество относительно передового существующего метода KLSP-GP.

Кроме того, предложенный метод TT-GP позволяет выучивать экспрессивные ядровые функции на основе глубоких нейронных сетей. В частности, в экспериментальной секции на наборе данных CIFAR10 была показана возможность обучать ядро на основе нейронной сети с 8 слоями, содержащей несколько миллионов параметров. В отличие от единственного существующего на данный момент метода SV-DKL, позволяющего обучать подобные модели, метод TT-GP способен настраивать параметры сети и остальные параметры модели в единой процедуре без предобучения. С помощью предложенного метода на ряде задач классификации изображений удалось получить качество лучше, чем у любой ранее описанной в литературе модели на основе гауссовских процессов.

Список литературы

[1] *Titsias M. K.* Variational Learning of Inducing Variables in Sparse Gaussian Processes // *Journal of Machine Learning Research W & CP*. — 2009. — Vol. 5. — Pp. 567–574. — URL: http://proceedings.mlr.press/v5/titsias09a/titsias09a.pdf.

[2] *Hensman J., Fusi N., Lawrence N.* Gaussian Processes for Big Data // Uncertainty in Artificial Intelligence. — 2013. — URL: http://www.auai.org/uai2013/prints/papers/244.pdf.

[3] *Hensman J., Matthews A., Ghahramani Z.* Scalable Variational Gaussian Process Classification // Artificial Intelligence and Statistics (AISTATS). — 2015. — URL: https://arxiv.org/pdf/1411.2005.pdf.

[4] *Wilson A. G., Nickisch H.* Kernel interpolation for scalable structured Gaussian processes (KISS-GP) // International Conference on Machine Learning (ICML). — 2015. — URL: http://proceedings.mlr.press/v37/wilson15.pdf.

[5] *Wilson A. G., Adams R. P.* Gaussian process kernels for pattern discovery and extrapolation // International Conference on Machine Learning (ICML). — 2013. — URL: http://proceedings.mlr.press/v28/wilson13.pdf.

[6] Stochastic Variational Deep Learning / A. Wilson, Z. Hu, R. Salakhutdinov, E. Xing // Neural Information Processing Systems (NIPS). — 2016. — URL: https://arxiv.org/pdf/1611.00336v2.pdf.

[7] *Damianou A. C., Lawrence N. D.* Deep Gaussian Processes // Artificial Intelligence and Statistics (AISTATS). — 2013. — URL: http://staffwww.dcs.shef.ac.uk/people/A.Damianou/papers/deepGPsAISTATS.pdf.

[8] Variational Auto-encoded Deep Gaussian Processes / Z. Dai, A. C. Damianou, J. Gonzalez, N. D. Lawrence // International Conference on Learning

Representations (ICLR). — 2016. — URL: https://arxiv.org/pdf/1511.06455v1.pdf.

[9] Random Feature Expansions for Deep Gaussian Processes / K. Cutajar, E. Bonilla, P. Michiardi, M. Filippone. — 2017. — URL: https://arxiv.org/pdf/1610.04386.pdf.

[10] *Oseledets I. V.* Tensor-Train decomposition // *SIAM J. Scientific Computing*. — 2011. — Vol. 33, no. 5. — P. 2295–2317. — URL: http://epubs.siam.org/doi/abs/10.1137/090752286.

[11] *Rasmussen C. E., Williams C. K. I.* Gaussian Processes for Machine Learning. — MIT Press, 2006.

[12] *Quiñonero Candela J., Rasmussen C. E.* A Unifying View of Sparse Approximate Gaussian Process Regression // *Journal of Machine Learning Research*. — 2005. — Vol. 6. — Pp. 1939–1959. — URL: http://www.jmlr.org/papers/volume6/quinonero-candela05a/quinonero-candela05a.pdf.

[13] *Jaakkola T.* Tutorial on Variational Approximation Methods // *Advanced Mean Field Methods*. — 2001.

[14] *Izmailov P. A., Kropotov D. A.* Faster variational inducing input Gaussian process classification // *Machine Learning and Data Analysis*. — 2017. — Vol. 3, no. 1. — URL: http://jmlda.org/papers/doc/2017/no1/Izmailov2017SpeedingUpGPC.pdf.

[15] *Keys R. G.* Cubic convolution interpolation for digital image processing // *IEEE Transactions on Acoustics, Speech and Signal Processing*. — 1981. — Vol. 29, no. 6.

[16] *Silverman B. W.* Some aspects of the spline smoothing approach to non-parametric regression curve fitting // *Journal of the Royal Statistical Society B*. — 1985. — Vol. 47, no. 1. — Pp. 1–52. — URL: https://www.jstor.org/stable/2345542.

[17] Putting MRFs on a Tensor Train / A. Novikov, A. Rodomanov, A. Osokin, D. Vetrov // International Conference on Machine Learning (ICML). — 2014. — URL: http://proceedings.mlr.press/v32/novikov14.pdf.

[18] Tensorizing Neural Networks / A. Novikov, D. Podoprikhin, A. Osokin, Vetrov D. // Neural Information Processing Systems (NIPS). — 2015. — URL: https://arxiv.org/pdf/1509.06569.pdf.

[19] *Bouchard Guillaume*. Efficient bounds for the softmax function and applications to approximate inference in hybrid models // NIPS 2007 workshop for approximate Bayesian inference in continuous/hybrid systems. — 2007.