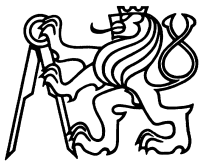
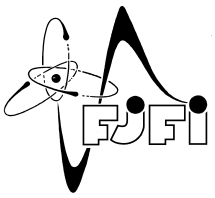


ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
Fakulta jaderná a fyzikálně inženýrská



Určení struktury diferenciálních rovnic z experimentálních dat

Determination of structures of differential equations from experimental data

Bakalářská práce

Autor: **Filip Bár**

Vedoucí práce: **Ing. Tomáš Oberhuber, Ph.D.**

Konzultant: **doc. Ing. Václav Šmídl, Ph.D.**

Akademický rok: 2019/2020

Katedra: matematiky

Akademický rok: 2019/2020

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student:	Filip Bár
Studijní program:	Aplikace přírodních věd
Obor:	Matematické inženýrství
Zaměření:	Aplikované matematicko-stochastické metody
Název práce (česky):	Určení struktury diferenciálních rovnic z experimentálních dat
Název práce (anglicky):	Determination of structures of differential equations from experimental data

Pokyny pro vypracování:

1. Seznamte se se základními metodami pro řešení diferenciálních rovnic. Zvláštní pozornost věnujte způsobům výpočtu gradientu řešení vzhledem k parametrům rovnice.
2. Seznamte se s Bayesovským odhadem parametrů, tj. výpočtem aposteriorní distribuce parametrů. Zvláštní pozornost věnujte variačním bayesovským metodám a zejména variantě zvané ELBO.
3. Seznamte se s konceptem apriorních rozdělení zvýhodňujících řídká řešení (constructive priors). Aplikujte metodu ELBO na odhad parametrů normálního rozdělení z dat s apriorním rozdělením, konkrétně na střední hodnotu, známou jako 'automatic relevance determination'.
4. Uvažujte příklad určení struktury lineární diferenciální rovnice z naměřených dat. Navrhněte vhodnou volbu parametrizace aposteriorní distribuce a určete její odhad metodou ELBO.
5. Aplikujte vyvinutou metodu na zvolená reálná data. Diskutujte výsledky a vlastnosti vyvinuté metody.

Doporučená literatura:

1. Ch. M. Bishop, Pattern recognition and machine learning. Springer, 2006.
2. Ch. C. Aggarwal, Neural Networks and Deep Learning: A Textbook. Springer, 2019.
3. L. S. Brunton, J. L. Proctor, J. N. Kutz, Discovering governing equations from data by sparse identification of nonlinear dynamical systems. In 'Proceedings of the National Academy of Sciences of the United States of Amerika' 113(15), 2016, 3932–3937.

Jméno a pracoviště vedoucího bakalářské práce:

Ing. Tomáš Oberhuber, Ph.D.
České vysoké učení technické v Praze , Fakulta jaderná a fyzikálně inženýrská , Katedra matematiky, Trojanova 339/13, 120 00 Praha 2 - Nové Město

Jméno a pracoviště konzultanta:

Datum zadání bakalářské práce: 31.10.2019

Datum odevzdání bakalářské práce: 7.7.2020

Doba platnosti zadání je dva roky od data zadání.

V Praze dne 24. října 2019

.....
garant oboru

.....
vedoucí katedry

.....
děkan

Poděkování:

Rád bych zde poděkoval především svému školiteli Ing. Tomáši Oberhuberovi, Ph.D. za pečlivost, ochotu, vstřícnost a odborné i lidské zázemí při vedení mé bakalářské práce. Dále bych rád poděkoval svému konzultantovi doc. Ing. Václavu Šmídlovi, Ph.D. za trpělivost a užitečné rady, bez kterých by tato práce nikdy nemohla vzniknout.

Rád bych také poděkoval své rodině a přítelkyni za podporu během mého celého studia.

A nakonec bych rád poděkoval svým přátelům Ing. Josefu Uchytilovi a Ing. Filipu Petráskovi, Ph.D. za morální i věcnou podporu během náročného období v začátcích studia.

Čestné prohlášení:

Prohlašuji, že jsem tuto práci vypracoval samostatně a uvedl jsem všechnu použitou literaturu.

V Praze dne 24. července 2020 Filip Bár

Název práce:

Určení struktury diferenciálních rovnic z experimentálních dat

Autor: Filip Bár

Obor: Matematické inženýrství

Zaměření: Aplikované matematicko-stochastické metody

Druh práce: Bakalářská práce

Vedoucí práce: Ing. Tomáš Oberhuber, Ph.D., České vysoké učení technické v Praze, Fakulta jaderná a fyzikálně inženýrská, Katedra matematiky, Trojanova 339/13, 120 00 Praha 2 - Nové Město

Konzultant: doc. Ing. Václav Šmídl, Ph.D., Ústav teorie informace a automatizace AV ČR, v.v.i., Pod Vodárenskou věží 4, 182 00, Praha 8

Abstrakt: Různé fyzikální systémy jsme schopni popsat diferenciální rovnicí nebo soustavou diferenciálních rovnic. Jevy v těchto systémech můžeme měřit v laboratořích a získávat tak data, která by měla odpovídat našemu matematickému popisu. Ve své práci budu za pomoci různých metod tyto data zpracovávat a vygenerovat z nich soustavy diferenciálních rovnic, podle kterých se data řídí.

Klíčová slova: Variační Bayes, lineární regrese, soustavy diferenciálních rovnic

Title:

Determination of structures of differential equations from experimental data

Author: Filip Bár

Abstract: Various physical systems can be described by a differential equation or a system of differential equations.Phenomena occuring within these systems can be measured in laboratories and thus data that should correspond to our mathematical description can be obtained. In this thesis, In this thesis, various methods for generating differential equations from experimental data are discussed.

Key words: Variational Bayesian methods, linear regresion, system of differential equations

Obsah

Úvod

Ve své práci se zabývám kombinací numerických metod, statistických metod a metod strojového učení k tvoření matematických modelů, které dokáží popsat určitý fyzikální systém. Tyto fyzikální systémy jsou obvykle popsatelné nějakou soustavou diferenciálních rovnic. Mým úkolem bude najít takový tvar diferenciálních rovnic, který bude nejlépe odpovídat experimentálním datům.

V začátku práce popisuji numerické metody pro řešení soustav diferenciálních rovnic. Tyto metody jsou užitečné zejména proto, že velké množství soustav diferenciálních rovnic je analyticky neřešitelných. Proto se v praxi využívá těchto metod pro aproximaci exaktního řešení.

Celá práce je poměrně hodně matematického ražení, proto je součástí práce také určité matematické zázemí, které popisuje některé ze základních pojmů matematické analýzy a matematické pravděpodobnosti.

Stěžejní částí většiny algoritmů strojového učení je takzvaná ztrátová funkce. Ta jistým způsobem penalizuje, resp. odměňuje stávající stav modelu, který popisuje daný fyzikální systém. Tuto funkci pak obvykle potřebujeme minimalizovat, resp. maximalizovat. Proto se ve své práci dále zabývám popisem některých optimalizačních metod, které dokáží extrémy funkcí najít.

Pro tvorbu matematických modelů popisujících určitý systém existuje více různých přístupů. Já se ve své práci pokusím blíže vysvětlit aproximativní bayesovský model, který vychází z Bayesova teorému. Za pomoci tohoto přístupu pak budu odhadovat parametry funkcí nebo diferenciálních rovnic, které daný fyzikální systém popisují.

Fyzikální systém nemusí být nutně popsán pouze diferenciální rovnicí. Dá se popsat například i klasickou funkcí. K jejímu nalezení se pak obvykle využívají regresní metody. Proto se i já budu v této práci zabývat regresí, která dokáže experimentální data aproximovat.

V předposlední kapitole využiji znalostí z předchozích kapitol a pokusím se navrhnout metody, které dokáží na základě dat najít správný tvar diferenciálních rovnic. Na konci své práce pak tyto metody aplikuji na konkrétní úlohy a zhodnotím jejich vlastnosti.

Kapitola 1

Numerické řešení diferenciálních rovnic

V následující kapitole si představíme základní numerické metody pro řešení soustav diferenciálních rovnic. Tato vědní oblast má velice široké uplatnění, zejména proto, že většina soustav diferenciálních rovnic je analyticky neřešitelná. Dané metody nám pak dokáží vrátit přibližné hodnoty řešení diferenciálních rovnic v bodech. Tyto body si můžeme téměř libovolně volit, stejně tak jsme schopni ovlivnit přesnost přibližné hodnoty řešení diferenciálních rovnic.

1.0.1 Dopředná diference

V následujících úlohách budeme potřebovat nahradit první derivaci nějakou diskrétní aproximací. Mějme funkci $y(t)$, která má konečnou druhou derivaci na okolí bodu t. Nejprve si zvolíme krok h. Následně uděláme Taylorův rozvoj funkce v bodě $t + h$

$$y(t+h) = y(t) + h\frac{dy(t)}{dt} + \frac{h^2}{2}\frac{d^2y(t)}{dt^2}\bigg|_{t=\xi} \quad \text{kde} \quad \xi \in [t, h].$$

Poslední člen výrazu se nazývá Taylorův zbytek. Vyjádřením první derivace dostaneme výraz

$$\frac{dy(t)}{dt} = \frac{y(t+h) - y(t)}{h} + O(h),$$

který aproximuje derivaci v bodě t s chybou prvního řádu. To znamená, že chyba aproximace je úměrná kroku h. Tedy čím menší budeme volit krok, tím menší bude chyba aproximace. To vyplývá z toho, že Taylorův zbytek je konečný a nezávisí na t. Výsledný chybový člen pak závisí lineárně na h.

1.1 Eulerova metoda

Pojďme si představit jednu ze základních integračních metod. Úloha, kterou budeme řešit, má tvar

$$\begin{aligned} y'(t) &= f(t, y(t)) \quad \text{na} \quad (t_0, t_k) \\ y(t_0) &= y_0. \end{aligned} \tag{1.1}$$

Eulerova metoda popisuje postup sestavení algoritmu, který řeší obyčejné diferenciální rovnice (ODE) s počátečními podmínkami. Jedná se o nejzákladnější explicitní metodu pro integraci diferenciálních rovnic. Jde o metodu prvního řádu. To znamená, že chyba integrace je úměrná velikosti kroku.

Pro řešení na počítači potřebujeme tuto úlohu diskretizovat. Nejprve si označíme délku řešeného intervalu $T = t_k - t_0$. Následně si zvolíme krok $h \ll T$, pro který budeme tuto soustavu řešit. Tímto krokem si navzorkujeme proměnnou t. Tedy místo $t \in \mathbb{R}$ budeme mít množinu $T_s = [t_0, t_0 + h, t_0 + 2h, \ldots, t_0 + Nh, t_k]$, kde $N = \lfloor T/h \rfloor$, kterou si pro jednoduchost přeznačíme na $t_j = t_0 + jh$. Tyto body budeme nazývat uzly. Pokud je $T/h \in \mathbb{N}$, pak koncový uzel t_k a uzel t_N splynou v jeden, v takovém případě uvažujeme pouze jeden z nich. Pojďme si nyní výraz (1.1) upravit. Derivaci na levé straně výrazu výše nahradíme dopřednou diferencí (1.0.1). Tím dostaneme následující výraz

$$\frac{y(t+h) - y(h)}{h} = f(t, y(t)).$$

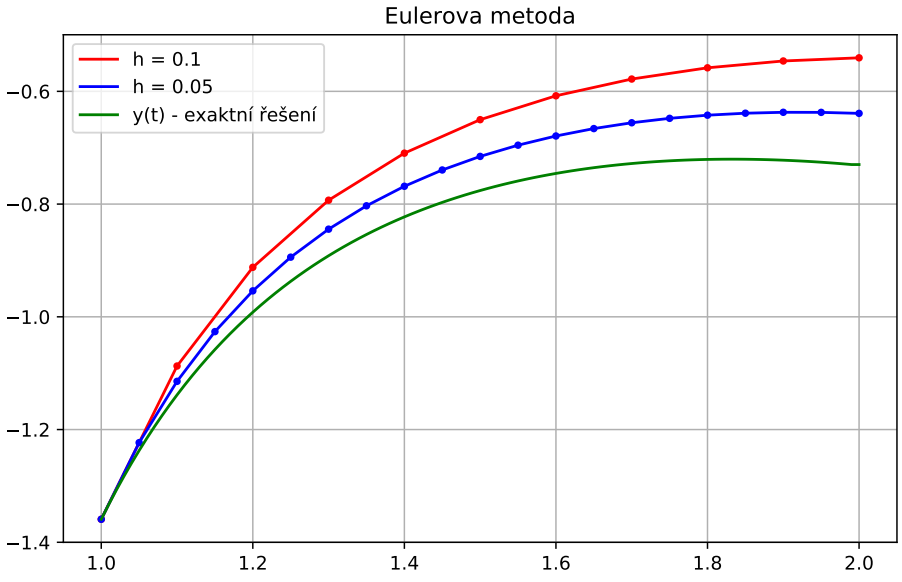
Ten můžeme dále upravit

$$y(t+h) = y(h) + hf(t, y(t)),$$

při přepisu $y_{n+1} = y(t+h)$ a $y_n = y(t)$ dostaneme rekurentní výraz pro výpočet funkce $y(t)$ v uzlech $t \in T_s$ ve tvaru

$$y_{n+1} = y_n + hf(t_n, y_n).$$

Tímto výrazem spolu s počátečními podmínkami jsme schopni spočítat aproximaci řešení diferenciální rovnice v předem zvolených uzlech.



Obrázek 1.1: Vykreslení Eulerovy metody pro různé volby kroků h při řešení Riccatiho rovnice $\frac{dy(t)}{dt} = t^{-4}e^t + y(t) + 2e^{-t}y^2(t)$ na intervalu $(1, 2)$.

1.2 Rungovu-Kuttova metoda

Tato metoda popisuje určité zobecnění Eulerovy metody. Stejně jako v předchozím případě chceme řešit úlohu ve tvaru

$$\begin{aligned} y'(t) &= f(t, y(t)) \quad \text{na} \quad (t_0, t_k) \\ y(t_0) &= y_0. \end{aligned}$$

Pro začátek si navzorkujeme proměnnou t na množinu bodů T_s s krokem h stejným způsobem jako u Eulerovy metody. Nově si označíme aproximační přírůstek funkce, který budeme hledat v následujícím tvaru

$$\Delta y(t, h) = b_1 k_1 + b_2 k_2 + \ldots + b_s k_s,$$

kde

$$\begin{aligned} k_1 &= h \cdot f(t_n, y_n), \\ k_2 &= h \cdot f(t_n + c_2 h, y_n + h(a_{21} k_1)), \\ k_3 &= h \cdot f(t_n + c_3 h, y_n + h(a_{31} k_1 + a_{32} k_2)) \\ &\vdots \\ k_s &= h \cdot f(t_n + c_s h, y_n + h(a_{s1} k_1 + a_{s2} k_2 + \ldots + a_{s,s-1} k_{s-1})). \end{aligned}$$

Výsledné hodnoty řešení diferenciální rovnice v uzlech pak budou mít tvar

$$y_{n+1} = y_n + h \sum_{i=1}^{s} b_i k_i.$$

Koeficienty b_i, c_i, a_{ij}, kde $i \in \{0, \ldots, s\}$ a $j \in \{0, \ldots, i-1\}$, se musí zvolit vhodně tak, aby přírůstek $\Delta y(t, h)$ co nejlépe aproximoval skutečný přírůstek řešení. To uděláme tím způsobem, že ho porovnáme s Taylorovým rozvojem skutečného přírůstku. Pro tento účel si vytvoříme pomocnou funkci

$$\begin{aligned} \phi_s(h) &= [y(t+h) - y(t)] - [b_1 k_1 + b_2 k_2 + \cdots + b_s k_s] \\ &= h y'(t) + \frac{h^2}{2} y''(t) + \ldots - [b_1 k_1 + b_2 k_2 + \cdots + b_s k_s] \\ &= h f(t, y(t)) + \frac{h^2}{2} \frac{d}{dt} f(t, y(t)) + \ldots - [b_1 k_1 + b_2 k_2 + \cdots + b_s k_s], \end{aligned}$$

kde $t \in [t_0, t_k)$. Koeficienty pak budeme volit tak, aby platilo

$$\phi_s^{(l)} = 0,$$

pro $l \in \{0, \ldots, m\}$ pro co možná nejvyšší $m \in \mathbb{N}$. To děláme proto, že pokud máme dvě hladké funkce f a g, u kterých se rovná prvních l derivací v bodě t, pak platí, že

$$\begin{aligned} f(t+h) - g(t+h) &= f(t) + h f'(t) + \ldots + \frac{h^{l+1}}{(l+1)!} f^{(k+1)}(\xi_f) \\ &\quad - g(t) + h g'(t) + \ldots + \frac{h^{l+1}}{(l+1)!} g^{(k+1)}(\xi_g) \\ &= \frac{h^{l+1}}{(l+1)!} \left[f^{(k+1)}(\xi_f) - g^{(k+1)}(\xi_g) \right], \end{aligned}$$

kde $\xi_f, \xi_g \in [t_0, t_k)$. Z toho vyplývá, že chyba aproximace funkce f funkcí g je řádu $k+1$. Tedy platí, že $|f(t+h) - g(t+h)| = O(k+1)$.

Vraťme se k Rungově-Kuttově metodě. U té tedy platí, že

$$y(t+h) - y(t) = [b_1 k_1 + b_2 k_2 + \ldots + b_s k_s] + \phi_s(h)$$
$$y(t+h) - y(t) = \Delta y(t,h) + O(h^{m+1}).$$

Po nalezení koeficientů získáme metodu s přesností řádu $s+1$.

Pojďme si nyní uvést nějaké možné tvary metody pro různé řády přesnosti. Například pro volbu $s = 1$ napočítáme koeficient $b_1 = 1$ a získáme Eulerovu metodu. Pokud zvolíme $s = 2$ můžeme nalézt koeficienty $b_1 = b_1 = 0.5$ a $c_2 = a_{12} = 1$. Tento tvar koeficientů však není jediný možný, pro vyšší řády přesnosti můžeme nalézt více různých kombinací hodnot koeficientů.

Metodu pak lze zobecnit na řešení soustavy diferenciálních rovnic, tedy pro úlohu

$$\boldsymbol{y}'(t) = \boldsymbol{f}(t, \boldsymbol{y}(t)) \quad \text{na} \quad (t_0, t_k)$$
$$\boldsymbol{y}(t_0) = \boldsymbol{y}_0,$$

kde $\boldsymbol{y}, \boldsymbol{f} \in \mathbb{R}^K$. Aproximace přírůstku se pak hledá ve tvaru

$$\Delta \boldsymbol{y}(t,h) = b_1 \boldsymbol{k}_1 + b_2 \boldsymbol{k}_2 + \cdots + b_s \boldsymbol{k}_s,$$

kde

$$\begin{aligned}
k_1 &= h \cdot \boldsymbol{f}(t_n, \boldsymbol{y}_n), \\
k_2 &= h \cdot \boldsymbol{f}(t_n + c_2 h, \boldsymbol{y}_n + h(a_{21} \boldsymbol{k}_1)), \\
k_3 &= h \cdot \boldsymbol{f}(t_n + c_3 h, \boldsymbol{y}_n + h(a_{31} \boldsymbol{k}_1 + a_{32} \boldsymbol{k}_2)) \\
&\vdots \\
k_s &= h \cdot \boldsymbol{f}(t_n + c_s h, \boldsymbol{y}_n + h(a_{s1} \boldsymbol{k}_1 + a_{s2} \boldsymbol{k}_2 + \ldots + a_{s,s-1} \boldsymbol{k}_{s-1})).
\end{aligned}$$

Koeficienty b_i, c_i, a_{ij}, kde $i \in \{0, \ldots, s\}$ a $j \in \{0, \ldots, i-1\}$, je pak možno volit stejným způsobem jako v předchozím případě.

Nejčastěji používanou volbou v praxi bývá $s = 4$ nebo $s = 5$.

Kapitola 2

Matematický úvod

2.1 Základní matematické pojmy

Na začátek si pojďme připomenout některé z pojmů matematické analýzy, které se nám budou v průběhu této nebo následujících kapitol hodit.

Definice 2.1.1. *Nechť* $F(X) : \mathcal{A} \to \overline{\mathbb{R}}$ *je množinová funkce na* $\mathcal{A}$ *splňující*

1. $\emptyset \in \mathcal{A}$
2. $F(\emptyset) = 0$
3. $\forall X \in \mathcal{A} : F(X) \geq 0$
4. $\forall X, Y \in \mathcal{A} : (X \cap Y = \emptyset \wedge X \cup Y \in \mathcal{A}) \Rightarrow F(X \uplus Y) = F(X) + F(Y)$
5. $\forall X, Y \in \mathcal{A} : X \subset Y \Rightarrow F(X) \leq F(Y)$

pak $F(x)$ *nazveme* ***mírou***.

Definice 2.1.2. *Nechť* Ω *je libovolná množina, pak systém podmnožin* $\mathcal{A}$ *splňující*

1. $\emptyset \in \mathcal{A}$
2. $A \in \mathcal{A} \Rightarrow A^c \in \mathcal{A}$
3. $A_1, A_2, A_3, ... \in \mathcal{A} \Rightarrow \bigcup_{i=1}^{\infty} A_i \in \mathcal{A}$

nazveme ***σ-algebrou***.

Definice 2.1.3. *Mějme vektorový prostor* V *nad tělesem* T. *Pak* ***normu*** *nazveme nezápornou funkci* $f : V \to \mathbb{R}$ *splňující pro* $\forall c \in T$ *a* $\forall \boldsymbol{u}, \boldsymbol{v} \in V$ *následující:*

1. $f(\boldsymbol{u} + \boldsymbol{v}) \leq f(\boldsymbol{u}) + f(\boldsymbol{v})$
2. $f(c\boldsymbol{v}) = |c| f(\boldsymbol{v})$
3. $f(\boldsymbol{v}) = 0 \Rightarrow \boldsymbol{v} = \overrightarrow{0}$

Příkladem normy může být například klasická Euklidovská vzdálenost vektorů

$$L_2(x) = \|x\|_2 = \left(\sum_{i=1}^{n}(x_i)^2\right)^{1/2} \quad \text{nebo pro funkce} \quad L_2(f(x)) = \|f(x)\|_2 = \left(\int_x f(x)^2 dx\right)^{1/2}.$$

Dalším příkladem může být Manhattanská norma pro vektory

$$L_m(x) = \|x\|_m = \sum_{i=1}^{n}|x_i| \quad \text{nebo pro funkce} \quad L_n(f(x)) = \|f(x)\|_m = \int_x |f(x)|dx.$$

Definice 2.1.4. *Mějme libovolnou neprázdnou množinu S a zobrazení $m : S \times S \to [0, +\infty)$, které splňuje následující:*

1. $m(x, y) = 0 \Leftrightarrow x = y$
2. $\forall x, y \in S$ *platí* $\quad m(x, y) = m(y, x)$
3. $\forall x, y, z \in S$ *platí* $\quad m(x, z) \leq m(x, y) + m(y, z)$.

Pak takové zobrazení nazveme ***metrikou***.

Metrika se dá vytvořit například tak, že vezmeme dva prvky z množiny S, které od sebe následně odečteme a aplikujeme na ně normu. Tímto způsobem se pak dá porovnávat odlišnost různých matematických prvků.

2.2 Úvod do matematické pravděpodobnosti

Jedním ze stěžejních pojmů matematické pravděpodobnosti je **náhodná veličina**. Její přesné a matematicky korektní zavedení je poměrně náročný a zdlouhavý počin. Proto případného čtenáře odkážu na [1], kde celé korektní zavedení může nalézt. Já se zde pokusím jen o hrubé nastínění tohoto pojmu.

Jde o proces, kdy nějakému jevu (třeba fyzikálnímu) přiřadíme číselnou hodnotu. Například u hodu mincí, můžeme stavu, kdy padne panna, přiřadit číselnou hodnotu 0 a pokud padne orel, přiřadíme hodnotu 1. V takovém případě pak náhodná veličina může nabývat pouze dvou hodnot. Nyní se můžeme ptát s jakou **pravděpodobností** mohou jednotlivé jevy nastat, což je další ze stěžejních pojmů, které je potřeba si zadefinovat.

Definice 2.2.1. *Buď tedy Ω neprázdná množina a $\mathcal{A} \subset \Omega$ σ-algebra (definici lze nalézt v [6]). Potom funkci $P : \mathcal{A} \to \mathbb{R}$, která splňuje*

1. $(\forall A \in \mathcal{A})(\, P(A) \geq 0\,)$
2. $P(\Omega) = 1$
3. *Buď* $(A_k)_{k=1}^{\infty} \in \mathcal{A}$ *systém neslučitelných jevů, potom*

$$P\left(\sum_{k=1}^{\infty} A_k\right) = \sum_{k=1}^{\infty} P(A_k),$$

nazveme pravděpodobností.

Bystrý čtenář si může povšimnout, že se jedná o míru.

Definice 2.2.2. *Množinu Ω nazveme prostor elementárních jevů, prvky množiny $\omega \in \Omega$ nazveme elementární jevy. Pravděpodobnostním prostorem pak budeme mít na mysli trojici $(\Omega, \mathcal{A}, P)$.*

Pokud bychom se vrátili k hodu mincí, pravděpodobnost (za předpokladu pravidelné mince) by se vyjádřila jako $P(\textit{padla panna}) = P(0) = 0.5$, $P(\textit{padl orel}) = P(1) = 0.5$. Zde se jedná o takzvanou diskrétní náhodnou veličinu a její pravděpodobnost.

Může se také stát, že náhodná veličina bude schopna nabývat nekonečného množství hodnot. Například vzdálenost zásahu od středu terče při střelbě z luku. Taková náhodná veličina pak teoreticky může nabývat jakékoli kladné hodnoty. Náhodné veličiny, u kterých pozorujeme podobné chování, nazveme absolutně spojitými, pro korektní popis čtenáře odkážu opět na [1]. V dalším výkladu se budeme výhradně zaměřovat právě na absolutně spojité náhodné veličiny. Pojďme si tedy zavést dva stěžejní pojmy pro jejich popis.

Definice 2.2.3. *Nechť X je absolutně spojitá náhodná veličina. Nechť existují funkce $f_X(x)$ a $F_X(x)$ takové, že*

$$P(X \in W) = F_X(x) = \int_{-\infty}^{x} f_X(t)\,dt\,,$$

*kde $W = (-\infty, x\,]$. Funkci $f_X(x)$ pak nazveme **hustotou pravděpodobnosti** náhodné veličiny X a funkci $F_X(x)$ **distribuční funkcí** náhodné veličiny X.*

Poznámka 2.2.4. *Obě tyto funkce jsou nezáporné. Distribuční funkce je pak neklesající s oborem hodnot $Ran(F_X) \subset [0, 1]$. Oproti tomu hustota pravděpodobnosti je shora neomezená. Může tedy nabývat jakýchkoli kladných hodnot.*

Definice 2.2.5. *Nechť $f_X(x|\theta)$ je hustota pravděpodobnosti závisející na parametru θ. Řekneme, že tato hustota spadá do **exponenciální třídy hustot** pokud ji leze zapsat ve tvaru*

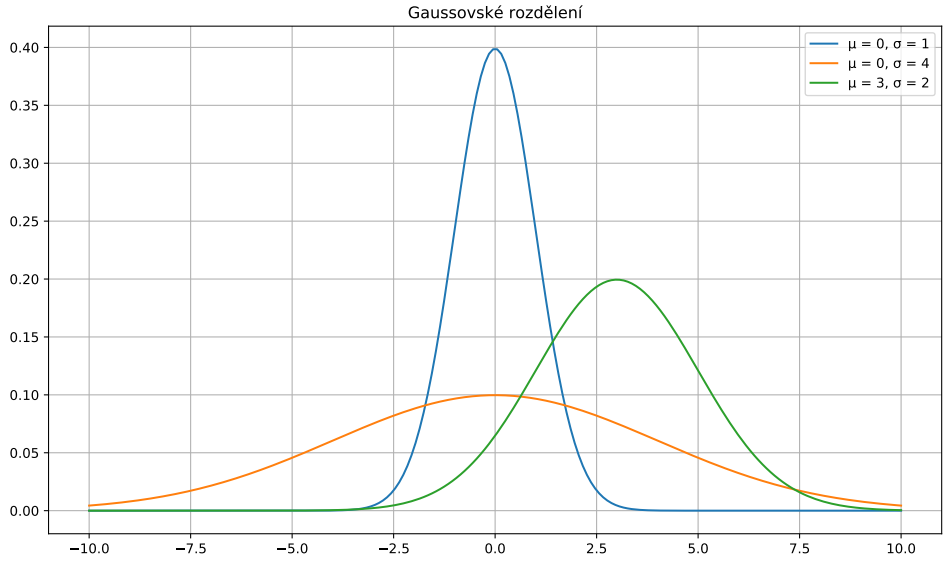
$$f_X(x|\theta) = h(x)exp(\eta(\theta) \cdot T(x) - A(\theta)),$$

kde $h(x), \eta(\theta), T(x)$ a $A(\theta)$ jsou nějaké neznámé funkce.

Většina běžně užívaných hustot pravděpodobnosti spadá do exponenciální třídy hustot. Jednou z nejpoužívanějších je pak Gaussovo rozdělení

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \quad \text{kde} \quad \mu \in \mathbb{R}, \sigma \in \mathbb{R}^+,$$

které se při volbě parametrů $(\mu, \sigma^2) = (0, 1)$ nazývá také normální rozdělení. Pokud se náhodná veličina X bude podle tohoto rozdělení chovat, označíme ji jako $X \sim N(\mu, \sigma^2)$ nebo taky $p(X) = N(X; \mu, \sigma^2)$.



Obrázek 2.1: Příklady různých hustot pravděpodobnosti popisující Gaussovo rozdělení

Další, ne tak často používanou hustotou, je inverzní gamma rozdělení

$$f(x) = \frac{\beta^{\alpha}}{\Gamma(\alpha)}(1/x)^{\alpha+1}exp(-\beta/x) \quad \text{kde} \quad \alpha > 0, \beta > 0,$$

které budeme v následujících kapitolách poměrně často využívat. Pro jeho značení budeme užívat $X \sim \text{Inv-Gamma}(\alpha, \beta)$ nebo taky $p(X) = \text{Inv-Gamma}(X; \alpha, \beta)$. Na rozdíl od Gaussovy hustoty, která je definovaná na celém $\mathbb{R}$, je inverzní gamma hustota definována pouze pro $x > 0$.

Nyní, když máme sestrojený matematický popis spojité náhodné veličiny, se můžeme ptát, co se stane, když se pokusíme náhodnou veličinu transformovat. Způsob, jakým se pak hustota pravděpodobnosti změní, je následující.

Věta 2.2.6. *Nechť X je náhodná veličina se spojitou distribuční funkcí F_X. Nechť dále $f_X(x) = F'_X(x)$ existuje všude s výjimkou nanejvýš konečně mnoha bodů. Buď $h(x)$ ryze monotónní funkce, jejíž derivace $h'(x) \neq 0$ na celém definičním oboru. Položme $Y = h(X)$. Pak Y má hustotu*

$$g_Y(y) = f_X[h^{-1}(y)]\left[\frac{\partial h^{-1}(y)}{\partial y}\right].$$

Tato věta popisuje pouze transformaci jednorozměrné náhodné veličiny. Nám se však bude hodit i vícerozměrný případ. Pojďme si tedy zadefinovat nový pojem.

Definice 2.2.7. *Uspořádanou n-tici náhodných veličin*

$$\boldsymbol{X}(\omega) = (X_1(\omega), ..., X_n(\omega))$$

budeme nazývat ***náhodný vektor***.

Nyní si můžeme zavést transformaci vícerozměrné náhodné veličiny.

Věta 2.2.8. *Nechť $X = (X_1, ..., X_n)^T$ je náhodný vektor s hustotou pravděpodobnosti $f_X(X)$. Mějme dále funkci $h(x) : \mathbb{R}^n \to \mathbb{R}^n$, která je regulární a prostá na otevřené množině G, pro niž platí, že $\int_G h(x)dx = 1$. Pak náhodný vektor $Y = h(X)$ má pravděpodobnostní hustotu rovnu*

$$g_Y(y) = \begin{cases} f_X[h^{-1}(y)]|D_{h^{-1}}(y)| & pro \quad y \in h(G) \\ 0 & pro \quad y \notin h(G) \end{cases},$$

kde $D_{h^{-1}}(y)$ je jakobián funkce h^{-1}, názorně tedy

$$D_{h^{-1}}(y) = \begin{bmatrix} \frac{\partial h_1^{-1}}{\partial y_1} & \cdots & \frac{\partial h_1^{-1}}{\partial y_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_n^{-1}}{\partial y_1} & \cdots & \frac{\partial h_n^{-1}}{\partial y_n} \end{bmatrix}.$$

Jedním z dalších velmi důležitých pojmů matematické pravděpodobnosti je nezávislost náhodných veličin. Její definice je následující.

Definice 2.2.9. *Nechť $X_1, ..., X_n$ jsou náhodné **nezávislé** veličiny, právě tehdy když pro libovolné borelovské množiny (definici lze nalézt v [6]) platí vztah*

$$P[\cap_{i=1}^n [\omega : X_i(\omega) \in B_i]] = \prod_i^n P[\omega : X_i(\omega) \in B_i],$$

Věta 2.2.10. *Nechť X je náhodný vektor se sdruženou distribuční funkcí F a nechť F_i je marginální distribuční funkce náhodné veličiny $X_i, i \in \{1, \ldots, n\}$. Pak řekneme, že $X_1, \ldots, X_n$ jsou nezávislé právě tehdy, když platí*

$$F(x_1, \ldots, x_n) = F_1(x_1) \cdots F_n(x_n).$$

Věta 2.2.11. *Nechť náhodné veličiny $X_1, \ldots, X_n$ mají sdruženou hustotu pravděpodobnosti f a marginální hustoty $f_1, \ldots, f_n$. Veličiny $X_1, \ldots, X_n$ nazveme nezávislými právě tehdy, když platí že*

$$f(x_1, \ldots, x_n) = f_1(x_1) \cdots f_n(x_n) \quad skoro\ všude.$$

Jak lze vydedukovat ze samotného pojmu nezávislost, jde o vyjádření stavu, kdy se náhodné veličiny neovlivňují. Tedy hustota pravděpodobnosti nebo distribuční funkce jednotlivých náhodných veličin závisí jen a pouze na sobě samé.

Další z pojmů bez kterých se neobejdeme je střední hodnota, někdy také nazývaná očekávaná hodnota (z anglického *Expected*, odtud značka $\mathbb{E}$). Její definice pak vypadá následovně.

Definice 2.2.12. *Nechť $(\Omega, \mathcal{A}, P)$ je pravděpodobnostní prostor a $X : \Omega \to \mathbb{R}$ je náhodná veličina, pak její **střední hodnotu** definujeme jako*

$$\mathbb{E}(X) = \int_\Omega X(\omega)dP(\omega),$$

speciálně pak pro spojité rozdělení X s pravděpodobnostní hustotou $f(x)$ jako

$$\mathbb{E}(X) = \mathbb{E}_{f(x)}(X) = \int_{\mathbb{R}} xf(x)dx,$$

nebo pro diskrétní rozdělení X, kde $P[X = x_i] = p_i$ *pro* $i \in I$ *jako*

$$\mathbb{E}(X) = \sum_{i \in I} x_i p_i.$$

Tato definice nám usnadní zavedení obecného a centrálního momentu, které jsou jednu z charakteristik pravděpodobnostního rozdělení. Ty nám mohou pomoct popsat vlastnosti pravděpodobnostních rozdělení, jako jsou například střední hodnota, rozptyl, šikmost a špičatost.

Definice 2.2.13. *Nechť X je náhodná veličina, její k-tý* ***obecný moment*** *definujeme jako*

$$\mu'_k = \mathbb{E}[X^k]$$

a k-tý ***centrální moment*** *veličiny X definujeme jako*

$$\mu_k = \mathbb{E}[(X - \mu'_1)^k].$$

Například parametry u Gaussova rozdělení jsou první obecný moment a druhý centrální moment, tedy $\mu = \mu'_1$ a $\sigma^2 = \mu_2$.

Poslední pojem bez kterého se v našem povídání neobejdeme, je podmíněná pravděpodobnost.

Definice 2.2.14. *Nechť* $(\Omega, \mathcal{A}, P)$ *je pravděpodobnostní prostor s mírou P. Nechť* $A, B \in \mathcal{A}$. *Pak* ***podmíněnou pravděpodobnost*** *jevu A za podmínky, nastání jevu B definujeme jako*

$$P(A|B) = \frac{P(A \cap B)}{P(B)},$$

kde $P(B) > 0$.

Slovy tedy jde o pravděpodobnost jevu A za předpokladu, že nastal jev B. Je zřejmé, že pokud jsou jevy nezávislé, pak je podmíněná pravděpodobnost jevu A rovna klasické pravděpodobnosti jevu A.

Při práci s podmíněnými pravděpodobnostmi se nám bude hodit takzvané Řetězové pravidlo.

Věta 2.2.15. *Mějme opět pravděpodobnostní prostor* $(\Omega, \mathcal{A}, P)$ *a* $A_1, ..., A_n \in \mathcal{A}$ *jevy, pro které platí, že* $P(A_1, ..., A_{n-1}) > 0$. *Potom*

$$P(A_1, ..., A_n) = P(A_n|A_{n-1}, ..., A_1) \cdot P(A_{n-2}|A_{n-2}, ..., A_1) \cdots P(A_2|A_1) \cdot P(A_1).$$

Jeho tvar umožňuje přepsat sdruženou pravděpodobnost do součinu podmíněných pravděpodobností.

Poslední vyslovená věta této kapitoly se nazývá **Bayesův teorém**.

Věta 2.2.16. *Mějme jevy* A, B *přičemž* $P(B) > 0$ *pak*

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}.$$

Jde o velice důležitý výraz, který budeme i nadále používat. Jeho tvar je klíčovou myšlenkou v řadě algoritmů strojového učení.

2.3 Integrace Monte Carlo

Monte Carlo je numerická metoda pro odhad hodnoty určitého integrálu za pomoci generování bodů z určitého náhodného rozdělení.

Předpokládejme, že chceme spočítat integrál z funkce $f(x)$ v mezích od a do b, tedy

$$F = \int_a^b f(x)dx.$$

Tento integrál můžeme aproximovat průměrem funkčních hodnot vzorků rovnoměrného rozdělení pravděpodobnosti $x_i \sim I(a, b)$, které je definované následovně.

Definice 2.3.1. ***Rovnoměrným rozdělením*** *na intervalu (a, b), kde $-\infty < a < b < \infty$, nazveme takové rozdělení, které má ve všech bodech intervalu konstantní hustotu pravděpodobnosti. Platí tedy, že*

$$p(x) = \begin{cases} \dfrac{1}{b-a}, & pro\ x \in (a, b) \\ 0, & jinak \end{cases},$$

kde $p(x)$ je hustota pravděpodobnosti.

Vezměme si tedy N vzorků $x_i \in (a, b)$ generovaných z rovnoměrného rozdělení $I(a, b)$. Výsledný odhad integrálu bude mít tvar

$$\langle F^N \rangle = (b-a)\frac{1}{N}\sum_{i=1}^{N}(f(x_i)). \tag{2.1}$$

Poměrně snadno se dá ukázat, že tento odhad integrálu je nestranný (definici lze nalézt v [1]), tedy že $\mathbb{E}[\langle F^N \rangle] = F$. Můžeme se také ptát, s jakou chybou jsme integrál odhadli. K tomu s v praxi používá nestranný odhad druhého centrálního momentu

$$\sigma_N = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(f(x_i) - \langle F^N \rangle)^2}.$$

Pro řešenou úlohu pak tedy platí, že

$$\int_a^b f(x)dx \approx \langle F^N \rangle \pm \sigma_N.$$

Tento postup můžeme mírně modifikovat pro výpočet odhadu středních hodnot funkcí nějaké náhodné veličiny. Mějme tedy opět nějakou obecnou funkci $f(x)$ definovanou na množině V. Mějme také pravděpodobnostní rozdělení s příslušnou hustotu pravděpodobnosti $q(x)$, která je rovněž definovaná na množině V. Naším úkolem bude spočítat

$$\mathbb{E}_{x\sim q(x)}[f(x)] = \int_V f(x)q(x)dx.$$

V tomto případě k odhadu hodnoty integrálu nebudeme brát vzorky z rovnoměrného rozdělení, nýbrž z rozdělení odpovídajícího hustotě $q(x)$. Ty pak vložíme do vzorce (2.1) a dostaneme výsledný odhad integrálu

$$\langle F^N \rangle = (b-a)\frac{1}{N}\sum_{i=1}^{N}(f(x_i^q)) \quad \text{kde} \quad x_i^q \sim q(x) \quad \text{pro} \quad i \in 1, \ldots, N.$$

2.4 Maximálně věrohodný odhad

Nyní si pojďme představit statistickou metodu maximálně věrohodných odhadů (MLE). V matematické statistice jde o často používaný způsob hledání parametrů. Základní myšlenkou je maximalizování sdružené hustoty pravděpodobnosti dat vůči hledaným parametrům.

Předpokládejme tedy, že známe rodinu pravděpodobnostních hustot popisující rozdělení dat, ale neznáme její parametry. Například víme, že naměřená data se chovají podle Gaussova rozdělení, ale neznáme parametry μ a σ.

Definice 2.4.1. *Nechť $X = [X_1, ..., X_n]$ jsou data s odpovídající rodinou hustot $\mathcal{F} = \{f_X(X, \theta) : \theta \in \Theta \subset \mathbb{R}^k\}$. Pak funkci*

$$L(\theta) = f(X, \theta) \quad \forall \theta \in \Theta, \forall x \in \mathbb{R}^n$$

*nazveme **věrohodnostní funkcí** a funkci*

$$l(\theta) = logL(\theta) = logf(X, \theta) \quad \forall \theta \in \Theta, \forall x \in \mathbb{R}^n$$

*nazveme **logaritmickou věrohodnostní funkcí**.*

Pokud máme nezávislý náhodný výběr dat, můžeme věrohodnostní funkci zavést jako sdruženou hustotu pravděpodobnosti následujícím způsobem.

$$L(\theta) = \prod_{i=1}^{n} f_{X_i}(X_i, \theta)$$

To se nám hodí zejména proto, že v případě logaritmické věrohodnostní funkce přechází produkt v sumu, což může další počítání značně zjednodušit.

Pojďme si nyní zadefinovat **maximálně věrohodný odhad**.

$$\theta_{ML}(X) = \arg\sup_{\theta \in \Theta} L(\theta)$$

Jde tedy o odhad parametru, jehož hodnota závisí na naměřených datech. V praxi se pak jeho tvar hledá pomocí systému věrohodnostních rovnic.

$$\frac{\partial L(\theta_1, ..., \theta_k)}{\partial \theta_i} = \frac{\partial l(\theta_1, ..., \theta_k)}{\partial \theta_i} = 0 \quad i \in \widehat{k}$$

Všimněme si, že při hledání maximálně věrohodných odhadů nezáleží na tom, zda jej počítáme z logaritmické věrohodnostní funkce nebo pouze z věrohodnostní funkce. To je zapříčiněno tím, že všechny hustoty pravděpodobnosti jsou nezáporné a ryzí monotonií logaritmu na celém jeho definičním oboru.

Kapitola 3

Numerické hledání extrému funkcí

V této kapitole si představíme některé z optimalizačních algoritmů. Tyto algoritmy se využívají pro hledání extrému funkcí.

3.1 Metoda největšího spádu

Metoda největšího spádu je iterační optimalizační metoda pro nalezení lokálního minima diferencovatelné funkce. Její hlavní myšlenka spočívá v tom, že derivace v bodě nám udává směr největšího růstu funkce.

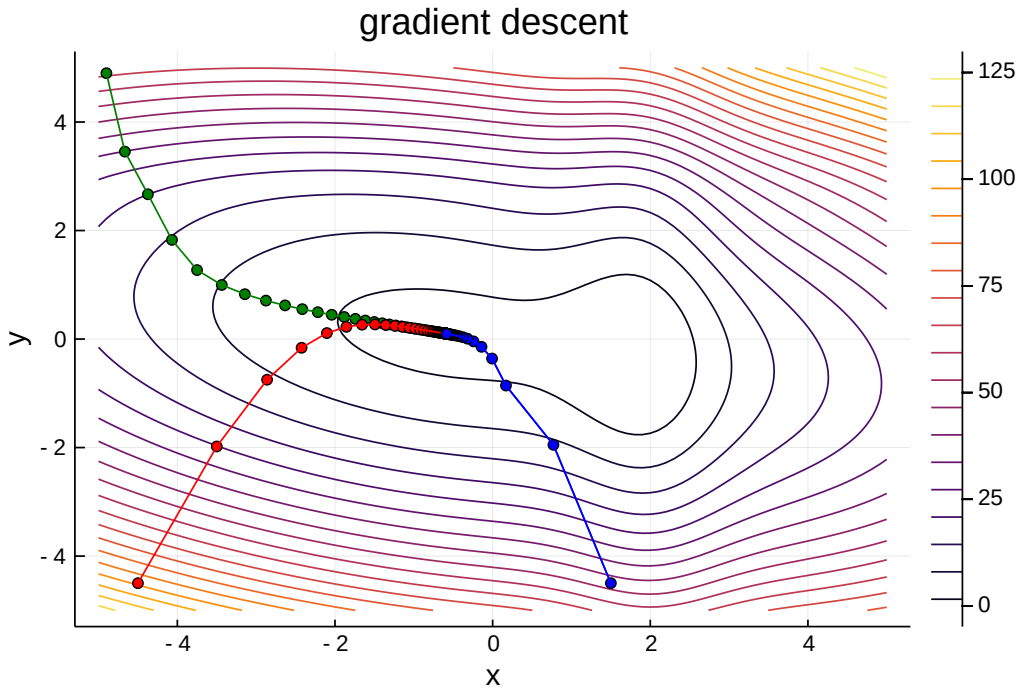
Mějme tedy nějaký libovolný bod $\boldsymbol{x}_0$ ležící v definičním oboru skalární funkce $F(\boldsymbol{x}) : \mathbb{R}^k \to \mathbb{R}$. Pokud bude tato funkce na okolí bodu $\boldsymbol{x}_0$ diferencovatelná, můžeme spočítat její gradient v tomto bodě. Jelikož gradient můžeme interpretovat jako směr a velikost největšího přírůstku funkce, záporně vzatý gradient pak bude mít směr a velikost největšího úbytku. Ten pak lze přičíst k původnímu bodu $\boldsymbol{x}_0$. Při opakování tohoto postupu vznikne posloupnost s následujícím tvarem.

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n - \nabla F(\boldsymbol{x}_n) \quad \text{kde} \quad n \in \mathbb{N}$$

V případě takového algoritmu může ovšem nastat problém, jestliže v blízkosti minima funkce bude příliš velký spád. Může se pak stát, že hodnotu minima přeskočíme. Proto se tvar posloupnosti upraví přidáním kroku $k \in (0, 1)$.

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n - k \cdot \nabla F(\boldsymbol{x}_n) \quad \text{kde} \quad n \in \mathbb{N}$$

V případě volby dostatečně malého kroku tak zajistíme, že minimum funkce nepřeskočíme. A bude tedy také platit, že $F(\boldsymbol{x}_n) \geq F(\boldsymbol{x}_{n+1})$ pro $\forall n \in \mathbb{N}$. Se zmenšujícím se krokem ovšem roste počet iterací nutných k dosažení minima. Proto u kulturně se chovajících funkci naopak můžeme k zvolit větší a dosáhnout tak minima rychleji.



Obrázek 3.1: Grafické znázornění metody největšího spádu.

3.2 ADAM Algoritmus

Stejně jako v předchozím případě se jedná o optimalizační algoritmus, který je schopný hledat minima skalárních funkcí. Jeho hlavní odlišností je to, že si určitým způsobem pamatuje směr a rychlost z předchozích kroků. To nám v některých případech může pomoci najít minimum tam, kde metoda největšího spádu selže.

Pro tuto metodu je třeba si nejprve zavést dva pomocné vektory v $n+1$-té iteraci.

$$\boldsymbol{m}_x^{(n+1)} = \beta_1 \boldsymbol{m}_x^{(n)} + (1-\beta_1)(\nabla F(\boldsymbol{x}_n)) \quad \text{kde} \quad \boldsymbol{m}_x^{(0)} = \overrightarrow{0}$$
$$\boldsymbol{v}_x^{(n+1)} = \beta_2 \boldsymbol{v}_x^{(n)} + (1-\beta_2)(\nabla F(\boldsymbol{x}_n))^2 \quad \text{kde} \quad \boldsymbol{v}_x^{(0)} = \overrightarrow{0}$$

Koeficienty $\beta_1, \beta_2 \in (0,1)$ nazýváme zapomínající koeficienty. Obvykle se pak volí $\beta_1 = 0.9$ a $\beta_2 = 0.999$. První pomocný vektor $\boldsymbol{m}_x^{(n+1)}$ je počítaný z kombinace všech předchozích gradientů. Jednotlivým gradientům pak exponenciálně klesá významnost spolu s každou novou iterací. Podobně pak vektor $\boldsymbol{v}_x^{(n+1)}$ je kombinací kvadrátu gradientů, kterým významnost klesá obdobným způsobem.

Tyto pomocné vektory jsou následovně normovány

$$\widehat{\boldsymbol{m}}^{(n+1)} = \frac{\boldsymbol{m}_x^{(n+1)}}{1-(\beta_1)^{n+1}}$$
$$\widehat{\boldsymbol{v}}^{(n+1)} = \frac{\boldsymbol{v}_x^{(n+1)}}{1-(\beta_2)^{n+1}}.$$

Když si oba výrazy rozepíšeme

$$\widehat{\boldsymbol{m}}^{(n+1)} = \frac{\beta_1^n \nabla F(\boldsymbol{x}_0) + \beta_1^{n-1} \nabla F(\boldsymbol{x}_1) + \cdots + \beta_1 \nabla F(\boldsymbol{x}_{n-1}) + \nabla F(\boldsymbol{x}_n)}{\beta_1^n + \beta_1^{n-1} + \cdots + \beta_1 + 1}$$

$$\widehat{\boldsymbol{v}}^{(n+1)} = \frac{\beta_2^n \nabla F(\boldsymbol{x}_0)^2 + \beta_2^{n-1} \nabla F(\boldsymbol{x}_1)^2 + \cdots + \beta_2 \nabla F(\boldsymbol{x}_{n-1})^2 + \nabla F(\boldsymbol{x}_n)^2}{\beta_2^n + \beta_2^{n-1} + \cdots + \beta_2 + 1},$$

můžeme v nich rozeznat vzorec pro vážený průměr s exponenciálními vahami. Pomocný vektor $\widehat{\boldsymbol{m}}^{(n+1)}$ má pak ve výsledném vzorci význam určité setrvačnosti. Na druhou stranu $\widehat{\boldsymbol{v}}^{(n+1)}$ zde zajišťuje adaptivní délku kroku, tedy takový, který se v průběhu iterací mění.

Z obou pomocných vektorů pak vytvoříme výsledný tvar algoritmu pro výpočet $\boldsymbol{x}_{n+1}$, který má tvar

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n + k \cdot \frac{\widehat{\boldsymbol{m}}^{(n+1)}}{\sqrt{\widehat{\boldsymbol{v}}^{(n+1)}} + e}.$$

Zde je e nějaké hodně malé číslo (řádově 10^{-8}), které je přidáno jako pojistka proti dělení nulou. Koeficient k pak splňuje stejný význam jako u metody největšího spádu. Pro úplné a korektní vysvětlení algoritmu lze nahlédnout do [4].

3.3 Stochastický gradient

Na začátek si pojďme představit, co je to ztrátová funkce. Mějme nějakou obecnou funkci $L(w, D) : \mathbb{R}^k \times \mathbb{R}^{n \times l} \to \mathbb{R}$, kde w představuje takzvané váhy a D soubor data. Tato funkce pak nějakým logickým způsobem penalizuje, resp. odměňuje stávající stav popisovaného systému svou hodnotou.

Pro lepší představu si uvedeme příklad. Mějme soubor dat $D = [d_1, ..., d_n]$, kde $d_i \in \mathbb{R}^l$ pro $\forall i \in \widehat{n}$. Ztrátová funkce pak může vypadat následovně

$$L(w) = L(w, D) = \sum_{i=1}^{n} (d_i - f(w))^2 .$$

Zde se nějakou obecnou funkcí $f(w)$ snažím co nejlépe aproximovat všechny data v souboru.

Ztrátové funkce jsou stěžejní částí valné většiny algoritmů strojového učení. V praxi se pak postupuje tak, že se funkci $L(w)$ snažíme minimalizovat, resp. maximalizovat přes váhy w. Navržení této funkce a následná optimalizace se však pro různé systémy může výrazně lišit.

Mějme tedy vhodně sestavenou funkci $L(w)$, kterou chceme minimalizovat. K tomu lze použít jednu z předchozích metod, kupříkladu metodu největšího spádu.

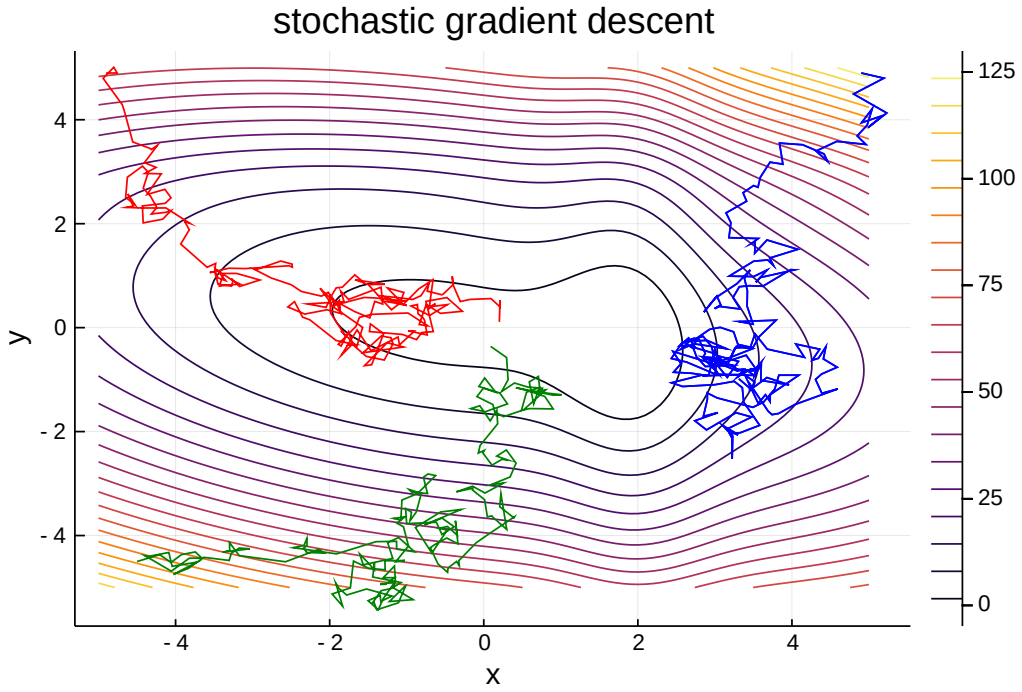
$$w_{n+1} = w_n - k \cdot \nabla_w L(w, D) \tag{3.1}$$

Tyto metody budou fungovat dobře, dokud nebudeme mít příliš velký soubor dat. V případě velkého souboru (řádově $n = 10^6$ a větší) ovšem nastává problém a tyto metody se stávají výpočetně velmi náročné. Proto se místo nich v takových případech používá stochastický gradient. Tato metoda si místo celého souboru dat vezme pouze jeden náhodně vybraný vzorek (popřípadě malou skupinu vzorků). Vůči němu pak vypočítá gradient ztrátové funkce a pozmění hodnotu stávajících vah. Tento postup se pak opakuje dokud se nedostaneme na požadovanou hodnotu ztrátové funkce.

Mějme tedy náhodný výběr $D^\star = [d^\star_{(1)}, ..., d^\star_{(k)}]$ z původního souboru dat, kde $k \ll n$. Ztrátová funkce pak bude mít tvar

$$L^\star(w) = L^\star(w, D) = \sum_{i=1}^{k} \left(d^\star_{(i)} - f(w)\right)^2 .$$

Z této nové ztrátové funkce bude výpočet gradientu mnohem snazší. Ze vztahu (3.1) pak můžeme vypočítat nový člen posloupnosti w_k. Celý tento postup spolu s novým náhodným výběrem $D^\star$ pak opakujeme, dokud se dostatečně nepřiblížíme k minimu ztrátové funkce.



Obrázek 3.2: Grafické znázornění stochastického gradientu.

Kapitola 4

Bayesovská teorie

4.1 Aproximace Bayesova pravidla

V této podkapitole se budeme hlouběji věnovat Bayesově teorému, respektive jednomu z možných přístupů jak Bayesův teorém řešit. Podrobnější vysvětlení celé problematiky je popsáno v literatuře od Bishopa, ve které je celý problém probírán důkladněji.

Pro začátek si vyslovíme Řetězové a Součtové pravidlo pro hustoty pravděpodobnosti.

Věta 4.1.1. *Mějme sdružené hustoty pravděpodobnosti $p(x, y, z)$ a $q(x, y)$, pak jednotlivá pravidla mají tvar:*

$$\textit{Řetězové pravidlo:} \quad p(x, y, z) = p(x|y, z) \cdot p(y|z) \cdot p(z)$$

$$\textit{Součtové pravidlo:} \quad q(y) = \int q(x, y)dx$$

$q(y)$ se pak nazývá marginální hustota pravděpodobnosti.

S využitím Součtového pravidla si přepíšeme tvar Bayesova teorému pro hustoty pravděpodobnosti

$$p_\theta(z|x) = \frac{p_\theta(x|z)p_\theta(z)}{p_\theta(x)} = \frac{p_\theta x|z)p_\theta(z)}{\int_z p_\theta(x|z)p_\theta(z)dz}. \tag{4.1}$$

Zde budeme z považovat za takzvanou latentní proměnnou. Ta pro nás obvykle bude znamenat neznámou skupinu parametrů, kterou se budeme snažit najít. Proměnná x pak obvykle reprezentuje vysoce dimenzionální soubor dat. Proměnná θ je skupina parametrů hustot pravděpodobnosti. Zde se na rozdíl od proměnných z a x bude jednat o deterministickou proměnnou. Celý výraz se pak dá zapsat zkrácenou formou

$$p_\theta(z|x) \propto p_\theta(x|z)p_\theta(z),$$

jelikož jmenovatel ve výrazu výše nezávisí na latentní proměnné a je tedy pouze normalizační konstantou.

Pojďme si celý výraz výše podrobněji rozebrat. Jednotlivé členy se nazývají

$$\textbf{posterior} \propto \textbf{likelihood} \cdot \textbf{prior}.$$

Aposteriorní rozdělení (**posterior**) je pro nás obvykle neznámé. Popisuje pravděpodobnostní rozdělení latentní proměnné podmíněné daty. Naším úkolem obvykle bude ho nalézt. Pravděpodobnostní

hustota **likelihood**, též nazývaná věrohodnost, popisuje předpokládané rozdělení naměřených dat. Apriorní rozdělení (**prior**) pak do modelu přináší informaci o předpokládaném tvaru rozdělení latentních proměnných.

Pojďme si pro názornost uvést příklad. Mějme soubor dat $x = [x_1, ..., x_n]$ generované z hustoty pravděpodobnosti $p(x|z)$ a ptejme se na odhad latentní proměnné (parametru) z.

Obvyklým přístupem je najít maximálně věrohodný odhad.

$$z_{ML} = \text{argmax} \prod_{i=1}^{n} p(x_i|z) = \text{argmax} \prod_{i=1}^{n} log\, p(x_i|z)$$

Zde je odhad z_{ML} deterministickou proměnnou. Bayesovský přístup oproti tomu z považuje také za náhodnou veličinu, u které ale předpokládá určité chování. Informaci tohoto předpokládaného chování pak vloží do apriorního rozdělení $p_\theta(z)$. Následným aplikováním Bayesova teorému dostaneme aposteriorní rozdělení.

$$p_\theta(z|x) = \frac{\prod_{i=1}^{n} p_\theta(x_i|z)p_\theta(z)}{\int \prod_{i=1}^{n} p_\theta(x_i|z)p_\theta(z)dz}$$

Zde však nastává problém, že integrál ve jmenovateli nejsme v obecném případě schopni spočítat. Například i pro jednoduchou volbu apriorní hustoty z Gaussova rozdělení je výraz analyticky neřešitelný. Dokonce i numerická integrace z důvodu vysoce dimenzionálního souboru dat x i případného velkého počtu latentních proměnných z není vhodná. Proto se v takovém případě snažíme skutečný tvar $p_\theta(z|x)$ alespoň co nejlépe aproximovat. K tomu nám poslouží hustota pravděpodobnosti $q(z)$ zvolená z nějaké vhodné rodiny hustot. Obvykle pak taková hustota závisí na parametrech ϕ, jejichž prostřednictvím se snažíme co nejvíce přiblížit k aposteriornímu rozdělení. Ovšem pro to, abychom mohli určit, jak moc se od sebe dané hustoty liší, potřebujeme nějaké měřítko. Zde by se například mohla zdát vhodnou volbou Euklidovská vzdálenost rozdílu hustot

$$L_2(p_\theta(z|x) - q_\phi(z)) = \left(\int_z (p_\theta(z|x) - q_\phi(z))^2 dz \right)^{1/2}.$$

Ta ovšem není vhodná hned ze dvou důvodů. Ten první je, že v obecném případě nejsme schopni $p_\theta(z|x)$ spočítat. Tím pádem nejsme schopni spočítat ani Euklidovskou vzdálenost těchto funkcí. Druhý problém, který zde nastává, je, že obě funkce jsou hustoty pravděpodobnosti. Obě jsou normované tak, že při integraci přes celý definiční obor dostaneme hodnotu 1. To znamená, že i při obrovském rozdílu mezi jednotlivými hustotami budeme dostávat malé hodnoty ztrátové funkce. Při hledání jejího minima pak narazíme na problém. Spočtené gradienty budou téměř rovny nule nebo dokonce úplně nulové. V takovém případě je nalezení minima ztrátové funkce za pomoci námi představených optimalizačních metod prakticky nemožné. Z toho důvodu budeme muset zavést nějaký vhodnější ukazatel rozdílnosti mezi aposteriorním rozdělením $p_\theta(z|x)$ a aproximativním hustotou $q_\phi(z)$.

4.2 Kullback-Leibler divergence a ELBO (OK)

KL divergence byla představena roku 1951 dvěma Americkými matematiky, Solomonem Kullbackem a Richardem Leiblerem. Jde o určité měřítko pro odlišnost dvou pravděpodobnostních rozdělení. Zde se nabízí zmínit, že na rozdíl od L_2 rozdílu hustot, který splňuje definici metriky, se zde o metriku nejedná, jelikož její obecný tvar

$$D_{KL}(p(x)||q(x)) = \int_x p(x) \log \left(\frac{p(x)}{q(x)} \right) dx, \tag{4.2}$$

ani tvar pro náš případ volby hustot pravděpodobnosti

$$D_{KL}(p_\theta(x|z)||q_\phi(z)) = \int_z p_\theta(x|z)\log\left(\frac{p_\theta(x|z)}{q_\phi(z)}\right)dz \tag{4.3}$$

není symetrický. Tedy $D_{KL}(p_\theta(x|z)||q_\phi(z))$ a $D_{KL}(q_\phi(z)||p_\theta(x|z))$ se obecně nerovnají. Platí však, že D_{KL} je vždy nezáporná. Nule se rovná v případě, že pravděpodobnostní rozdělení q a p jsou shodné.

Při její aplikaci ale opět vzniká problém, protože neznáme aposteriorní rozdělení $p_\theta(z|x)$, kde x jsou naměřená obvykle vysoce dimenzionální data a z je neznámá náhodná veličina. Proto se v praxi využívá metoda *Evidence lower bound* (ELBO), která je definovaná jako

$$L(\theta,\phi) = H(q_\phi(z); p_\theta(x,z)) - H(q_\phi(z)) = \int_z q_\phi(z)\log p_\theta(z,x) - \int_z q_\phi(z)\log q_\phi(z), \tag{4.4}$$

kde $H(q_\phi(z))$ je klasická entropie a $H(q_\phi(z); p_\theta(x,z))$ je takzvaná křížová entropie (*cross entropy*). Zde jde vidět, že požadavky na znalost aposteriorního rozdělení už nejsou aktuální. Pojďme se nyní podívat, jakým způsobem spolu KL divergence a metoda ELBO souvisí.

$$D_{KL}(q_\phi(z)||p_\theta(z|x)) = \int_z q_\phi(z)\log\left(\frac{p_\theta(x)q_\phi(z)}{p_\theta(z,x)}\right)dz$$

$$D_{KL}(q_\phi(z)||p_\theta(z|x)) = \int_z q_\phi(z)\log q_\phi(z)dz - \int_Z q_\phi(z)\log p_\theta(z,x) + \log p_\theta(x)dz$$

$$L(\theta,\phi) = \int_Z q_\phi(z)\log p_\theta(z,x)dz - \int_z q_\phi(z)\log q_\phi(z)dz = \log p_\theta(x) - D_{KL}(q_\phi(z)||p_\theta(z|x)).$$

Jelikož je $D_{KL}(q||p) \geq 0$ pro jakékoli volby p, q a $\log p(x)$ je při naměřených datech konstantní, vidíme, že v případě maximalizace $L(\theta,\phi)$ minimalizujeme KL divergenci. Což je velice vítaná vlastnost, protože můžeme minimalizovat KL divergenci bez znalosti aposteriorního rozdělení. Obvykle pak budeme minimalizovat záporně vzatou funkci $L(\theta,\phi)$ přes parametry θ a ϕ. Praxe ukazuje, že na takto zvolenou ztrátovou funkci lze bez obtíží aplikovat dříve představené optimalizační metody.

Pro dosažení aproximace je potřeba zvolit si třídy distribučních funkcí $q_\phi(z)$. Tato volba je prakticky neomezená, dokonce se v případě složitosti nemusíme obávat overfittingu [**?**] jako třeba u klasické lineární regrese. Jedinou penalizací za příliš složitou volbu $q_\phi(z)$ bude delší doba minimalizace KL divergence.

Jeden z nejčastějších přístupů volby $q_\phi(z)$ je rozdělit jednotlivé prvky z do disjunktních skupin a předpokládat faktorizaci. Výsledná volba pak vypadá následovně

$$q_\phi(z) = \prod_{i=1}^{M} q_i(z_i|\phi_i).$$

4.3 Reparametrizační trik

Máme tedy představenou metodu ELBO, která nám popisuje způsob, jak najít minimální hodnotu KL divergence bez znalosti aposteriorního rozdělení. V praxi pak obvykle argumentem sdružené hustoty není pouze latentní proměnná z, ale nějaká funkce latentní proměnné $f(z)$. ELBO pak vypadá následovně:

$$\begin{aligned} L(\theta,\phi) &= \int_Z q_\phi(z)(\log p_\theta(x, f(z)) - \log q_\phi(z))dz \\ &= \mathbb{E}_{z\sim q_\phi(z)}[\log p_\theta(x, f(z)) - \log q_\phi(z)]. \end{aligned} \tag{4.5}$$

V některých jednodušších případech lze $L(\theta, \phi)$ spočítat analyticky. Například pokud je $f(z)$ lineární. Spočtení gradientů vůči parametrům $\nabla_\theta L(\theta, \phi)$ a $\nabla_\phi L(\theta, \phi)$ je v takovém případě jednoduché.

Pro názornost si uveďme příklad. Mějme soubor naměřených dat $x = [x_1, \ldots, x_n]$, u kterých předpokládáme neznámou Gaussovskou chybu. Tyto data reprezentují například nějakou výchylku vzdálenosti. My se pak budeme snažit tuto výchylku odhadnout proměnnou m. Zvolme si věrohodnost a apriorní hustoty

$$\begin{aligned} p(x|m, \omega) &= \prod_i^n N(x_i|m, \omega) \\ p(m) &= N(m|0, \tau^{-1}) \\ p(\omega) &= \text{Inv-Gamma}(\omega|\alpha, \beta). \end{aligned}$$

Volba věrohodnosti vyplývá ze zadání, jelikož předpokládáme Gaussovskou chybu. Volba prvního apriorního členu vyjadřuje, že upřednostňujeme nulovou hodnotu výchylky. Tedy pokud se naměřené data vlezou do šumu, budeme výchylku považovat za nulovou. Volba druhého apriorního členu pak vyplývá z toho, že příliš velký i příliš malý rozptyl předpokládáme za málo pravděpodobný. V tomto případě je pak latentní proměnná $z = [m, \omega]$ a parametr $\theta = [\tau, \alpha, \beta]$. Obecně by místo latentní proměnné uvnitř věrohodnosti mohla být funkce latentní proměnné. V našem případě se však jedná o identitu $f(m) = m$ a $f(\omega) = \omega$, tedy o lineární funkci. Nyní si zvolíme aproximační hustotu $q(m, \omega) = q(m)q(\omega)$, kde

$$\begin{aligned} q(m) &= N(m|\widehat{m}, \widehat{s}) \\ q(\omega) &= \text{Inv-Gamma}(\omega|\widehat{a}, \widehat{b}). \end{aligned}$$

Praxe ukazuje, že je vhodné volit hustoty podobné těm, které model popisují. Zde je parametr $\phi = [\widehat{m}, \widehat{s}, \widehat{a}, \widehat{b}]$. Nyní si můžeme vyjádřit ELBO

$$\begin{aligned} L(\theta, \phi) &= \iint q(m, \omega) \log \frac{p(x, m, \omega)}{q(m, \omega)} dm d\omega \\ &= \mathbb{E}_{q(m)q(\omega)}[\log p(x|m, \omega)] + \mathbb{E}_{q(m)}[\log p(m)] + \mathbb{E}_{q(\omega)}[\log p(\omega)] - \mathbb{E}_{q(m)}[\log q(m)] - \mathbb{E}_{q(\omega)}[\log q(\omega)], \end{aligned}$$

kde jednotlivé členy mají následující tvar

$$\begin{aligned} \mathbb{E}_{q(m)q(\omega)}[\log p(x|m, \omega)] &= -\frac{n}{2}(\log(2\pi) + \mathbb{E}_{q(\omega)}[\log\omega]) - \frac{1}{2}\mathbb{E}_{q(\omega)}[\omega^{-1}] \sum_{i=1}^{n} \left(x_i^2 - 2x_i\mathbb{E}_{q(m)}[m] + \mathbb{E}_{q(m)}[m^2]\right) \\ \mathbb{E}_{q(m)}[\log p(m)] &= \frac{1}{2}\log\left(\frac{\tau}{2\pi}\right) - \frac{\tau}{2}\mathbb{E}_{q(m)}[m^2] \\ \mathbb{E}_{q(\omega)}[\log p(\omega)] &= \alpha\log(\beta) - \log(\Gamma(\alpha)) - (\alpha + 1)\mathbb{E}_{q(\omega)}[\log\omega] - \beta\mathbb{E}_{q(\omega)}[\omega] \\ \mathbb{E}_{q(m)}[\log q(m)] &= -\frac{1}{2}\log(2\pi\widehat{s}) - \frac{1}{2\widehat{s}}(\widehat{m}^2 - 2\widehat{m}\mathbb{E}_{q(m)}[m] + \mathbb{E}_{q(m)}[m^2]) \\ \mathbb{E}_{q(\omega)}[\log q(\omega)] &= \widehat{a}\log(\widehat{b}) - \log(\Gamma(\widehat{a})) - (\widehat{a} + 1)\mathbb{E}_{q(\omega)}[\log\omega] - \widehat{b}\mathbb{E}_{q(\omega)}[\omega]. \end{aligned}$$

Pro námi zvolené aproximační hustoty pravděpodobnosti platí, že

$$\begin{aligned} \mathbb{E}_{q(m)}[m] &= \widehat{m} \qquad & \mathbb{E}_{q(\omega)}[\omega^{-1}] &= \frac{\widehat{a}}{\widehat{b}} \\ \mathbb{E}_{q(m)}[m^2] &= \widehat{m}^2 + \widehat{s} \qquad & \mathbb{E}_{q(\omega)}[\log\omega] &= \Psi(\widehat{a}) - \log(\widehat{b}). \end{aligned}$$

Tyto hodnoty pak můžeme dosadit a dostat výsledný tvar

$$\mathbb{E}_{q(m)q(\omega)}[\log p(x|m,\omega)] = -\frac{n}{2}(\log(2\pi) + \Psi(\widehat{a}) - \log(\widehat{b})) - \frac{\widehat{a}}{2\widehat{b}}\sum_{i=1}^{n}\left(x_i^2 - 2x_i\widehat{m} + \widehat{m}^2 + \widehat{s}\right)$$

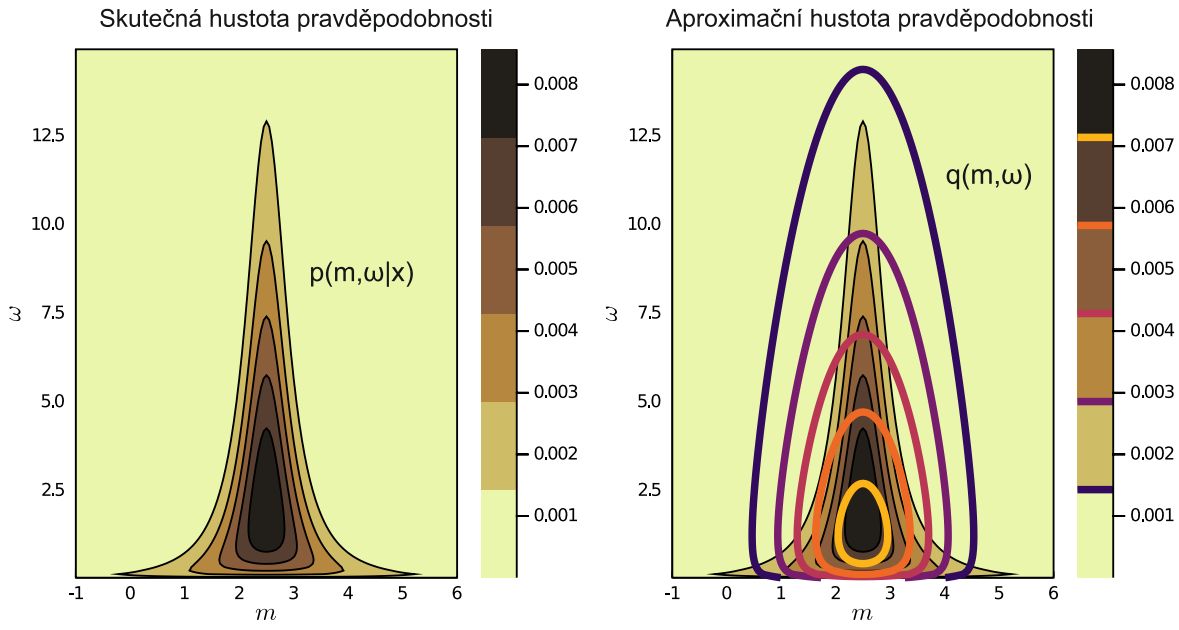
$$\mathbb{E}_{q(m)}[\log p(m)] = \frac{1}{2}\log\left(\frac{\tau}{2\pi}\right) - \frac{\tau(\widehat{m}^2 + \widehat{s})}{2}$$

$$\mathbb{E}_{q(\omega)}[\log p(\omega)] = \alpha\log(\beta) - \log(\Gamma(\alpha)) - (\alpha+1)(\Psi(\widehat{a}) - \log(\widehat{b})) - \beta\frac{\widehat{a}}{\widehat{b}}$$

$$\mathbb{E}_{q(m)}[\log q(m)] = -\frac{1}{2}\log(2\pi\widehat{s} + 1)$$

$$\mathbb{E}_{q(\omega)}[\log q(\omega)] = -\widehat{a} - \log(\widehat{b}\Gamma(\widehat{a})) + (1+\widehat{a})\Psi(\widehat{a}).$$

Tuto ztrátovou funkci už lze poměrně snadno optimalizovat. Pojďme si nyní ukázat jak takový příklad dopadne.



Obrázek 4.1: Demonstrační graf aposteriorní hustoty a výsledné aproximační hustoty pro experimentální data $x_1 = 3.0$ a $x_2 = 2.0$.

Všimněme si, že integrál šlo spočítat pouze díky znalosti středních hodnot latentní proměnné. To ovšem pro obecnou funkci latentní proměnné už nebude možné.

Vraťme se tedy k případu (4.5) s obecnou funkcí $f(z)$. Nejprve si ukážeme, jakým způsobem lze získat $\nabla_\theta L(\theta, \phi)$. Pro začátek předpokládejme, že ϕ je známé a fixní. Pro výpočet gradientu vzhledem k θ je pro nás důležitá pouze první část integrálu, tedy

$$\mathbb{E}_{z\sim q_\phi(z)}[\log p_\theta(x, f(z))] = \int_Z q_\phi(z)(\log p_\theta(x, f(z)))dz.$$

Tento integrál ovšem nejsme v úplné obecnosti schopni spočítat. Proto se v praxi využívá Monte Carlo odhad

$$\mathbb{E}_{z\sim q_\phi(z)}[\log p_\theta(x, f(z))] \approx \frac{1}{s}\sum_{i=1}^{s} \log p_\theta(x, f(z_i)) \quad \text{kde} \quad z_i \sim q_\phi(z) \text{ pro } \forall i = 1, \ldots, s.$$

Gradient pak nalezneme ve tvaru

$$\nabla_\theta \mathbb{E}_{z\sim q_\phi(z)}[\log p_\theta(x, f(z))] = \nabla_\theta \int_Z q_\phi(z)(\log p_\theta(x, f(z)))dz \approx$$
$$\approx \nabla_\theta \frac{1}{s}\sum_{i=1}^{s} \log p_\theta(x, f(z_i)) = \frac{1}{s}\sum_{i=1}^{s} \nabla_\theta \log p_\theta(x, f(z_i)).$$

Tímto způsobem tedy dostaneme aproximaci $\nabla_\theta L(\theta, \phi)$.

Nyní dalším úkolem bude najít $\nabla_\phi L(\theta, \phi)$. Tentokrát pro změnu předpokládejme, že θ je známé a fixní. V tomto případě musíme gradient počítat vzhledem k oběma členům integrálu

$$\mathbb{E}_{z\sim q_\phi(z)}[\log p_\theta(x, f(z)) - \log q_\phi(z)] = \int_Z q_\phi(z)(\log p_\theta(x, f(z)) - \log q_\phi(z))dz.$$

To z důvodu, že ačkoli člen $p_\theta(x, f(z))$ explicitně nezávisí na parametru ϕ, závisí na latentní proměnné z, která je určená rozdělením $q_\phi(z)$, které už na parametru ϕ závisí. Pojďme si nyní celý výraz v integrálu substituovat $h_\phi(z) = \log p_\theta(x, f(z)) - \log q_\phi(z)$. Zde nová funkce $h_\phi(z)$ nezávisí na θ, jelikož ho považujeme za fixní. Na rozdíl od předchozího případu už nemůžeme celý výraz nahradit Monte Carlo odhadem. To z toho důvodu, že potřebujeme znát gradient vzhledem k parametrům rozdělení, z kterého vzorky děláme. V takovém případě pak totiž vždy dostaneme výsledek nula, jelikož derivujeme konstantu. Tento problém ovšem můžeme obejít pomocí takzvaného **reparametrizačního triku**, jehož hlavní myšlenka spočívá v tom, že velké množství distribucí $q_\phi(z)$ lze přepsat jako deterministická transformace $T(\epsilon, \phi)$ nějaké základní distribuce $b(\epsilon)$, která už na ϕ nezávisí. Příklady pravděpodobnostních rozdělení, u kterých lze takovou transformaci provést, jsou uvedeny v literatuře [5]. Integrál za pomoci reparametrizačního triku pak můžeme upravit následovně

$$\begin{aligned}\mathbb{E}_{z\sim q_\phi(z)}[h_\phi(z)] &= \int_Z q_\phi(z) h_\phi(z) dz \\ &= \int_\epsilon b(\epsilon) h_\phi(T(\epsilon, \phi)) d\epsilon \\ &= \mathbb{E}_{\epsilon\sim b(\epsilon)}[h_\phi(T(\epsilon, \phi))].\end{aligned}$$

Tím jsme celý problém převedli na předchozí případ. Výsledný tvar pak bude vypadat následovně

$$\nabla_\theta \mathbb{E}_{\epsilon\sim b(\epsilon)}[h_\phi(T(\epsilon, \phi))] = \nabla_\phi \int_\epsilon b(\epsilon)(h_\phi(T(\epsilon, \phi))d\epsilon \approx$$
$$\approx \nabla_\phi \frac{1}{s}\sum_{i=1}^{s} h_\phi(T(\epsilon, \phi)) = \frac{1}{s}\sum_{i=1}^{s} \nabla_\phi h_\phi(T(\epsilon, \phi)).$$

Zde pro odhad integrálu opět použijeme metodu Monte Carlo.

Tímto je celý problém vyřešen. Na hledání minima pak už stačí jen použít některý vhodný optimalizační algoritmus.

Pojďme si celý postup opět demonstrovat na příkladu. Mějme soubor dat $x = [x_1, \ldots, x_n]$, u kterých předpokládáme, že se chovají podle nějaké obecné funkce $f(m)$. V kapitole 6 pak tato funkce

bude numerický řešič diferenciálních rovnic závislý na parametrech rovnice. Naším úkolem bude najít m tak, aby funkce co nejlépe korespondovala s daty. U dat dále pro ulehčení úlohy, předpokládáme Gaussovskou chybu se známým rozptylem ω_0. Zvolíme si věrohodnost a apriorní hustotu

$$p(x|f(m)) = \prod_i^n N(x_i|f(m), \omega_0)$$
$$p(m) = N(m|0, \tau^{-1}).$$

Volba věrohodnosti přímo vyplývá ze zadání úlohy. Volba apriorní hustota nám opět upřednostňuje nulovou hodnotu m.

Pro námi takto zadanou úlohu je latentní proměnná $z = m$ a parametr $\theta = \tau$. Nyní si zvolíme aproximační hustotu

$$q(m) = N(m|\widehat{m}, \widehat{s}).$$

Tuto hustotu si volíme proto, jelikož sdružená hustota pravděpodobnosti je popsána kombinací Gaussovských rozdělení, proto se zdá být vhodné ji aproximovat jiným Gaussovským rozdělením. Zde je parametr $\phi = [\widehat{m}, \widehat{s}]$. Nyní si můžeme vyjádřit

$$\begin{aligned} L(\theta, \phi) = L(\tau, \widehat{m}, \widehat{s}) &= \int q(m) \log \frac{p(x, f(m))}{q(m)} dm \\ &= \mathbb{E}_{q(m)}[\log p(x|f(m))] + \mathbb{E}_{q(m)}[\log p(m)] - \mathbb{E}_{q(m)}[\log q(m)], \end{aligned}$$

kde jednotlivé členy mají následující tvar

$$\mathbb{E}_{q(m)}[\log p(x|f(m))] = -\frac{n}{2}\log(2\pi\omega_0) - \frac{1}{2\omega_0}\sum_{i=1}^{n}\left(x_i^2 - 2x_i\mathbb{E}_{q(m)}[f(m)] + \mathbb{E}_{q(m)}[f^2(m)]\right)$$
$$\mathbb{E}_{q(m)}[\log p(m)] = \frac{1}{2}\log\left(\frac{\tau}{2\pi}\right) - \frac{\tau}{2}\mathbb{E}_{q(m)}[m^2]$$
$$\mathbb{E}_{q(m)}[\log q(m)] = -\frac{1}{2}(\log(2\pi\widehat{s}) - \frac{1}{2\widehat{s}}(\widehat{m}^2 - 2\widehat{m}\mathbb{E}_{q(m)}[m] + \mathbb{E}_{q(m)}[m^2]).$$

V tomto případě už nejsme schopni se střední hodnoty latentní proměnné zbavit tak snadno jako v předchozím případě, jelikož neznáme $\mathbb{E}_{q(m)}[f(m)]$ ani $\mathbb{E}_{q(m)}[f^2(m)]$. Vypomůžeme si tedy reparametrizačním trikem. Ten říká, že následující výrazy jsou ekvivalentní

$$m \sim N(\widehat{m}, \widehat{s})$$
$$m = \widehat{m} + \sqrt{\widehat{s}} \cdot \epsilon \quad \text{kde} \quad \epsilon \sim N(0, 1).$$

Tuto transformaci dosadíme do jednotlivých členů a použijeme Monte Carlo odhad pro $s = 1$

$$\mathbb{E}_{q(m)}[\log p(x|f(m))] = -\frac{n}{2}\log(2\pi\omega_0) - \frac{1}{2\omega_0}\sum_{i=1}^{n}\left(x_i^2 - 2x_i f(\widehat{m} + \sqrt{\widehat{s}} \cdot \epsilon) + f^2(\widehat{m} + \sqrt{\widehat{s}} \cdot \epsilon)\right)$$
$$\mathbb{E}_{q(m)}[\log p(m)] = \frac{1}{2}\log\left(\frac{\tau}{2\pi}\right) - \frac{\tau}{2}(\widehat{m} + \sqrt{\widehat{s}} \cdot \epsilon)^2$$
$$\begin{aligned} \mathbb{E}_{q(m)}[\log q(m)] &= -\frac{1}{2}(\log(2\pi\widehat{s}) - \frac{1}{2\widehat{s}}(\widehat{m}^2 - 2\widehat{m}(\widehat{m} + \sqrt{\widehat{s}} \cdot \epsilon) + (\widehat{m} + \sqrt{\widehat{s}} \cdot \epsilon)^2) \\ &= -\frac{1}{2}(\log(2\pi\widehat{s}) - \frac{(1 + 2\widehat{m})\sqrt{\widehat{s}}\epsilon + \widehat{s}\epsilon^2}{2\widehat{s}}. \end{aligned}$$

Z tohoto výrazu už jsme schopni spočítat $\nabla_{\tau,\widehat{m},\widehat{s}} L(\tau, \widehat{m}, \widehat{s})$. Tím pádem můžeme použít nějakou z optimalizačních metod. Aproximace integrálu za pomoci pouze jedné iterace u Monte Carlo metody je sama o sobě velmi špatný odhad. Je ovšem třeba zvážit, že těchto iterací budeme dělat opravdu velké množství. V průměru nám potom gradient vyjde ve směru největšího růstu tak, jako kdybychom celý výraz spočítali exaktně. Můžeme si také povšimnout, že se zde vlastně jedná o určitou modifikaci statistického gradientu, který místo dat používá náhodné vzorky normálního rozdělení.

Kapitola 5

Lineární regrese

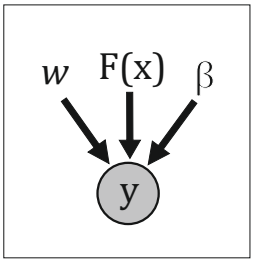
Tato kapitola vychází převážně z článku [7].

5.1 Klasická lineární regrese

Předpokládejme, že máme dvě fyzikální veličiny x a y, mezi kterými existuje určitá lineární závislost

$$\widehat{y} = w_0 + w_1 f_1(x) + \cdots + w_k f_k(x),$$

kde funkce $f_i(x)$, $i \in \{1, \ldots, k\}$, které budeme nazývat bazickými, i váhy w_j, $j \in \{0, \ldots, k\}$ jsou volené tak, aby model co nejlépe korespondoval s naměřenými daty. V praxi pak postupujeme způsobem, u kterého si nejdříve vybereme funkce, kterými chceme data aproximovat a následně hledáme váhy w tak, aby rozdíl mezi vypočtenými hodnotami a naměřenými daty byl co nejmenší.



Obrázek 5.1: Schéma klasické regrese

Pojďme si tedy nyní vzít soubor $x = [x_1,...,x_n]^T$ a k tomu odpovídající naměřená data $y = [y_1,...,y_n]^T$. U souboru x předpokládáme, že dané hodnoty jsou přesné. V modelu se pak x považuje za tzv. *vysvětlující (nezávislou) proměnou*. Zatímco v souboru y předpokládáme u dat normální nezávislou chybu a v modelu je považována za *vysvětlovanou (závislou) proměnou*. Naše data pak předpokládáme ve tvaru

$$y_i = y(x_i, w) = w_0 + w_1 f_1(x_i) + \cdots + w_k f_k(x_i) + e_i,$$

kde $e_i \sim N(0, \beta^{-1})$, $i \in \{1, \ldots, n\}$ je náhodná chyba s Gaussovým rozdělením se střední hodnotou 0 a rozptylem β^{-1}. Výraz výše si můžeme přepsat do maticové podoby

$$y = Fw + e \quad \text{kde}$$

$$F = F(x) = \begin{bmatrix} 1 & f_1(x_1) & \cdots & f_k(x_1) \\ 1 & f_1(x_2) & \cdots & f_k(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ 1 & f_1(x_n) & \cdots & f_k(x_n) \end{bmatrix}, \qquad w = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_k \end{bmatrix}, \qquad e = \begin{bmatrix} e_0 \\ e_1 \\ \vdots \\ e_n \end{bmatrix}.$$

Z předpokladu normality chyby vyplývá, že pravděpodobnostní rozdělení naměřených dat má tvar $p(y_i|x,w,\beta) = N(y_i|F(x_i)w,\beta^{-1})$ a jelikož předpokládáme nezávislost jednotlivých chyb, můžeme výraz přepsat následovně

$$p(y|x,w,\beta) = \prod_{i=1}^{n} N(y_i|F(x_i)w,\beta^{-1}) = N_n(y|F(x)w,\beta^{-1}\mathbb{I}).$$

Tento tvar nás vede na základní model regresní aproximace. Na obrázku 5.1 můžeme vidět jeho schéma. Váhy w, parametr β i funkce $F(x)$ jsou zde brány jako deterministické proměnné. Oproti tomu y je náhodná veličina a její popis je realizován hustotou pravděpodobnosti.

Nyní se můžeme pokusit najít odhad váhy w metodou maximální věrohodnosti. Věrohodnostní funkci a její logaritmus, který se budeme snažit maximalizovat, mají tvar

$$L(w,\beta) = p(y|x,w,\beta) = \left(\sqrt{\frac{\beta}{2\pi}}\right)^n exp\left(-\frac{\beta}{2}\sum_{i=1}^{n}(F(x_i)w - y_i)^2\right)$$

$$l(w,\beta) = log\, p(y|x,w,\beta) = -\frac{n}{2}log(2\pi) + \frac{n}{2}log(\beta) - \frac{\beta}{2}\sum_{i=1}^{n}(F(x_i)w - y_i)^2.$$

Maximální $l(w,\beta)$ pro konstantní β pak najdeme, pokud minimalizujeme sumu

$$S(w) = \sum_{i=1}^{n}(F(x_i)w - y_i)^2\ , \tag{5.1}$$

což vede k metodě nejmenších čtverců. Pokud $l(w,\beta)$ zderivujeme parciálně podle jednotlivých w_i, položíme rovno nule a přepíšeme do maticové podoby, dostaneme

$$F^T(y - F)w = 0 \quad \text{neboli} \quad F^T y = F^T F w.$$

Tento výraz se za předpokladu, že je matice $(F^T F)$ regulární, který je splněn pokud jsou sloupce matice F nezávislé, dá přepsat do tvaru

$$w_{ML} = (F^T F)^{-1} F^T y. \tag{5.2}$$

Tímto způsobem dostaneme explicitní vyjádření pro maximálně věrohodný odhad (MLE) vektoru w. Stejným způsobem pak můžeme odhadnout i parametr β, jeho odhad je roven

$$\frac{1}{\beta_{ML}} = \frac{1}{n}\sum_{i=1}^{n}(y_i - F(x_i)w_{ML})^2.$$

Spolu se zvolenými bazickými funkcemi a maximálně věrohodnými odhady w_{ML} máme hotový celý model. S jeho pomocí pak můžeme predikovat hodnoty $y_{\text{pred}}(x)$ následujícím způsobem

$$y_{\text{pred}}(x) = F(x)w_{ML} = w_{ML0} + w_{ML1}f_1(x) + \cdots + w_{MLk}f_k(x).$$

Ty pak můžeme porovnat s naměřenými hodnotami y. Jako jeden z ukazatelů jak dobře sestavený model odpovídá naměřeným datům se dá použít R^2 statistika, která porovnává variabilitu dat nevysvětlenou modelem a rozptyl dat samotných. Její tvar

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - F(x_i)w_{ML}))^2}{\sum_{i=1}^{n}(y_i - \overline{y})^2},$$

kde $\overline{y}$ je rovno aritmetickému průměru dat, nám říká, že čím větší je R^2, tím lépe náš model aproximuje data. Pokud $R^2 = 1$, znamená to, že všechny data leží na námi modelované křivce.

Tento model by velice dobře fungoval na soubory dat, které nejsou příliš velké, v opačném případě by ale nastal problém. A to především s hledáním inverzní matice $F^T F$. Proto se pro hledání odhadů vah w využívají optimalizační algoritmy. V případě, že je n řádově 10^6 a vyšší se využívá metoda *stochastického gradientu* (3.3). Pro optimalizační metody je ovšem nutné zavést si ztrátovou funkci, kterou chceme minimalizovat. My vyjdeme z mírně upraveného výrazu (5.1)

$$E_D(w) = \frac{1}{2} \sum_{i=1}^{n} (y_i - F(x_i)w)^2, \tag{5.3}$$

kde RSS $= 2E_D(w)$ se občas nazývá reziduální součet čtverců. Při námi zvoleném značení má l-tá iterace tvar

$$w_i^{(l+1)} = w_i^{(l)} - h\nabla E_D(w_i^{(l)}) = w_i^{(l)} + h\left(y_i - F(x_i)w_i^{(l)}\right) F(x_i), \tag{5.4}$$

kde h je učící krok a $i \in \{0, \ldots, k\}$.

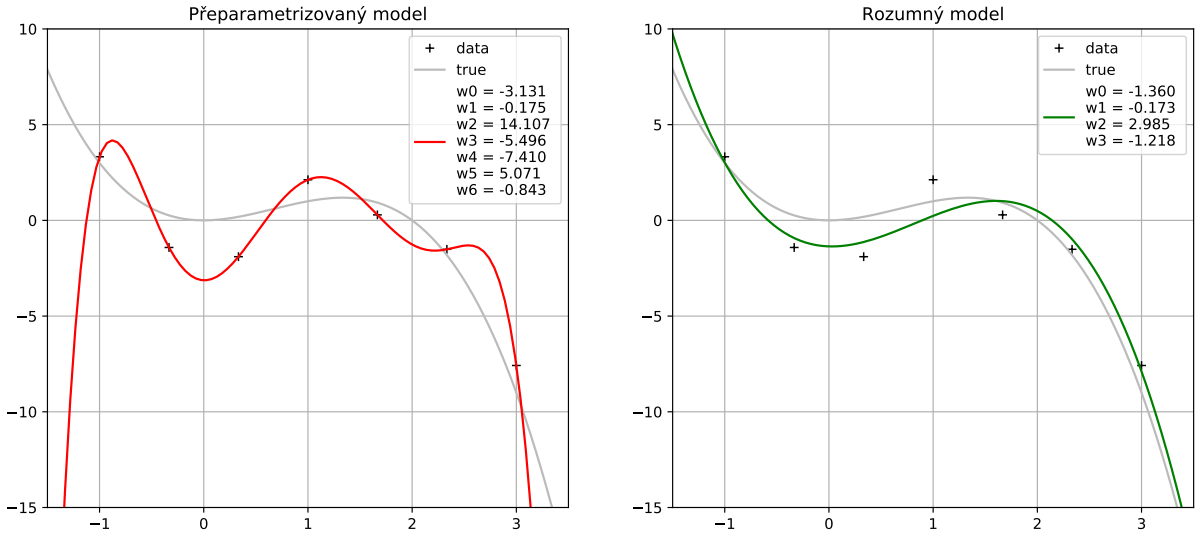
Celý tento model byl založen na předpokladu, že chybu obsahující data jsou Gaussovské. Jednou z jeho nevýhod může být velké ovlivnění naší křivky kvůli odlehlým hodnotám naměřených dat. Tím je myšleno, že jedna hodnota s velkou chybou může značně ovlivnit výsledný tvar vah w.

5.2 Ridge regrese

Jedním z problémů klasické lineární regrese bývá, že model je zbytečně složitý. Když si například vezmeme n počet měřených dat a pro regresi si zvolíme bazické n lineárně nezávislých bazických funkcí, vždy dokážeme najít takové w, aby statistika $R^2 = 1$. Což znamená, že naše modelovaná křivka protne všechna naměřené data. To by se na první pohled mohlo zdát vhodné, není tomu ovšem tak. Pro ilustraci si pojďme vygenerovat hodnoty

$$y_i = 2x_i^2 - x_i^3 + e_i \quad \text{kde} \quad e \sim N(0, 1) \quad \text{a} \quad i = 1, ..., 7.$$

Na obou grafech je znázorněno použití metody nejmenších čtverců (5.2). U prvního grafu je 7 bazických funkci, polynomy 0.–6. stupně. U druhého grafu pouze 4 bazické funkce, polynomy 0.–3. stupně. Ačkoli druhý graf neprotíná všechny generované hodnoty, skutečným váhám $w = [0, 0, 2, -1, 0, 0, 0]$, s kterými jsme data generovali, je blíže než první.



Obrázek 5.2: Dvě regresní křivky vygenerované klasickou lineární regresí. První s přebytečným počtem bazických funkci - 7. Druhá s přiměřeným počtem bazických funkcí - 4. Data generujeme z $y_i = 2x_i^2 - x_i^3 + e_i$.

Můžeme si všimnout, že velikost vah u prvního modelu je o dost vyšší než u druhého, což bývá jedním z ukazatelů, že model není zcela v pořádku.

V předchozím modelu klasické lineární regrese jsme předpokládali, že pravděpodobnostní rozdělení naměřených dat je rovno $p(y|x, w, \beta) = N_n(y|F(x)w, \beta^{-1}\mathbb{I})$. Na váhy w a parametr β nebyly kladeny žádné preference, což nás může přivést ke zbytečné složitosti modelu naznačeného na obrázku výše. Proto pro následují model přidáme preferenci nulové hodnoty pro váhy w. To zrealizujeme přidáním apriorní hustoty pravděpodobnosti

$$p(w) = N_k(w|0, \lambda^{-1}\mathbb{I}) \quad \text{kde} \quad w \in \mathbb{R}^{k+1}$$

do modelu. Tímto způsobem uděláme z deterministické váhy novou náhodnou veličinu. Váhy se nyní budou chovat podle Gaussova rozdělení s nulovou střední hodnotou. Tím do modelu přidáváme informaci, která nám upřednostňuje hodnotu každé jednotlivé váhy za nulovou.

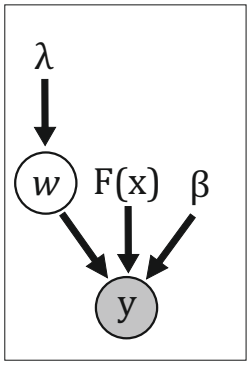
Nyní bude sdružená hustota pravděpodobnostní pro naměřená data vypadat následovně.

$$p(y, w) = p(y, w|x, \beta, \lambda) = p(y|x, w, \beta)\, p(w|\lambda) = N_n(\, y \,|\, F(x)w \,,\, \beta^{-1}\mathbb{I})\, N_{k+1}(\, w \,|\, 0 \,,\, \lambda^{-1}\mathbb{I})$$

Na obrázku 5.3 lze vidět schéma našeho nového modelu. Parametry λ a β, stejně tak jako funkce $F(x)$, jsou stále deterministické proměnné. Mají tedy přesně danou hodnotu. Oproti tomu váhy w i výstupní funkce y jsou považovány za náhodné veličiny. Jejich hodnota pak není dána exaktně, ale pouze prostřednictvím pravděpodobnostního rozdělení.

Nyní, když máme připravený model, můžeme odhady w získat metodou maximální věrohodnosti. Tedy nalézt maximum sdružené hustoty pravděpodobnosti $p(y, w)$ popřípadě $log\, p(y, w)$ s ohledem na váhy w. To najdeme za pomoci parciálních derivací podle vah w. Výsledný výraz následně položíme roven nule a vyřešíme. Tímto způsobem získáme nový explicitní vztah pro výpočet maximálně věrohodných odhadů vah

$$w_{ML} = (\lambda\mathbb{I} + F^T F)^{-1} F^T y.$$



Obrázek 5.3: Schéma ridge regrese

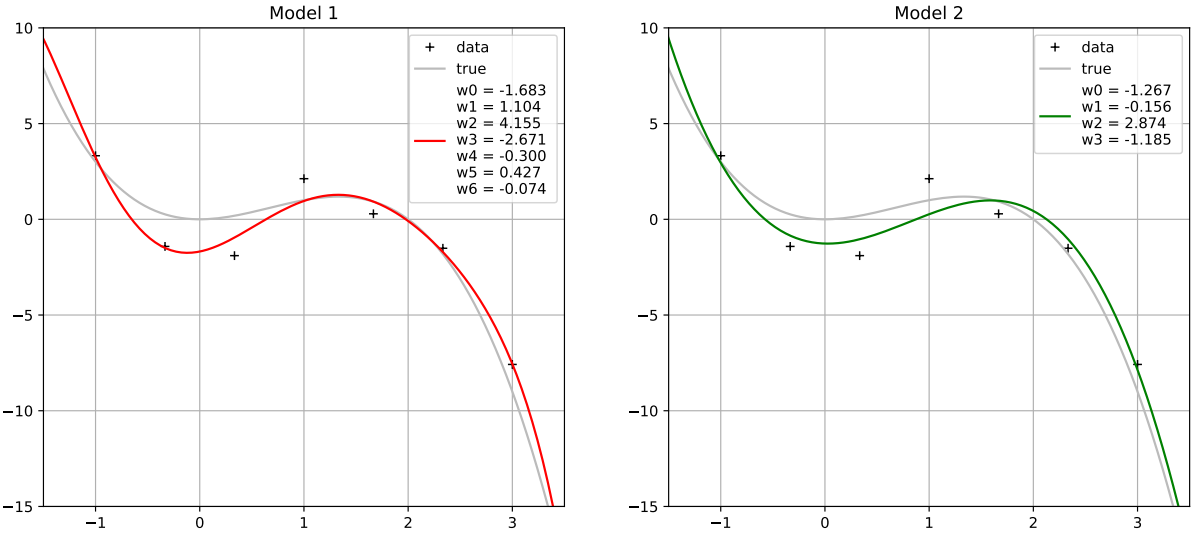
To ovšem za předpokladu že matice $(\lambda\mathbb{I} + F^T F)$ je regulární. V případě, že je výpočet inverzní matice příliš složitý, můžeme opět zavést ztrátovou funkci

$$E(w) = E_D(w) + \lambda E_W(w) = \frac{1}{2}\sum_{i=1}^{n}(y_i - F(x_i)w)^2 + \lambda\frac{1}{2}w^T w,$$

která je ekvivalentní s předchozím výrazem a počítat váhy iterativně.

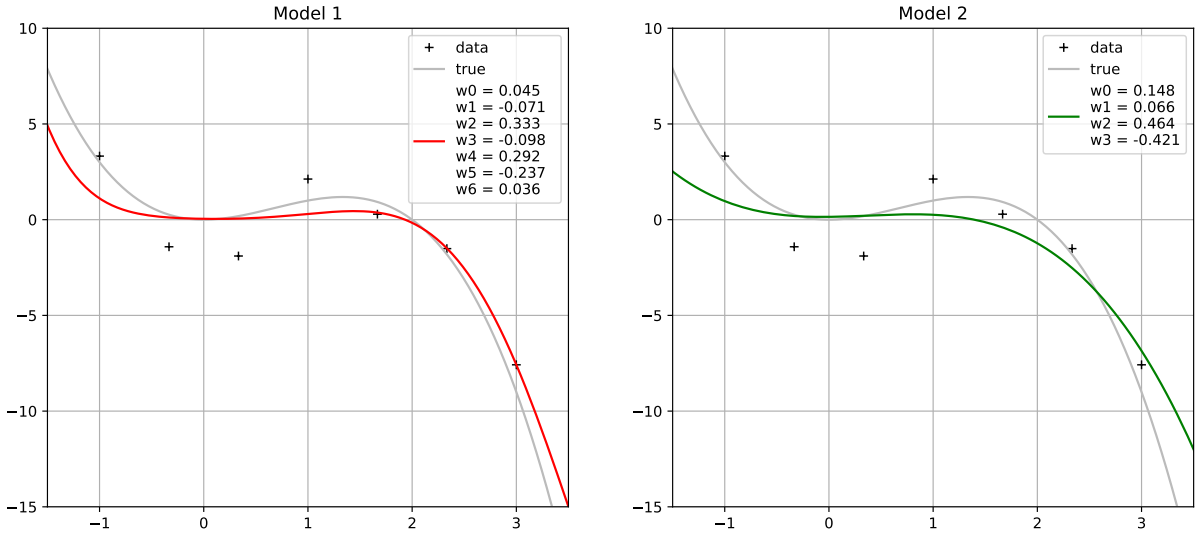
Parametr λ je v takovém případě volen nezáporný. Čím větší se zvolí, tím více tlačíme hodnotu vah v absolutní hodnotě k nule. V praxi se pak obvykle volí menší než jedna. Je však vhodné si volbu parametru otestovat.

Pojďme si nyní ukázat, co se stane, pokud použijeme novou metodu na data z obrázku 5.2.



Obrázek 5.4: Dvě regresní křivky vygenerované za pomoci ridge regrese při $\lambda = 0.05$. První s přebytečným počtem bazických funkci - 7. Druhá s přiměřeným počtem bazických funkcí - 4.Data generujeme z $y_i = 2x_i^2 - x_i^3 + e_i$.

Je zřejmé, že nová regresní křivka aproximuje skutečnou křivku lépe než předchozí metoda i pro nadbytečný počet vah. Dále si můžeme povšimnout, že absolutní hodnota jednotlivých vah v průměru klesla. Z obrázku by se dále mohlo zdát, že díky novému přístupu klesají absolutní hodnoty nevýznamných vah (vah odpovídajících skutečným vahám s nulovou hodnotou) rychleji než významné váhy. To ale bohužel obecně není splněno. Ukážeme si, jak by tato regrese dopadla při volbě parametru $\lambda = 10$.



Obrázek 5.5: Dvě regresní křivky vygenerované za pomoci ridge regrese při $\lambda = 10$. První s přebytečným počtem vah (7). Druhá s přiměřeným počtem vah (4).Data generujeme z $y_i = 2x_i^2 - x_i^3 + e_i$.

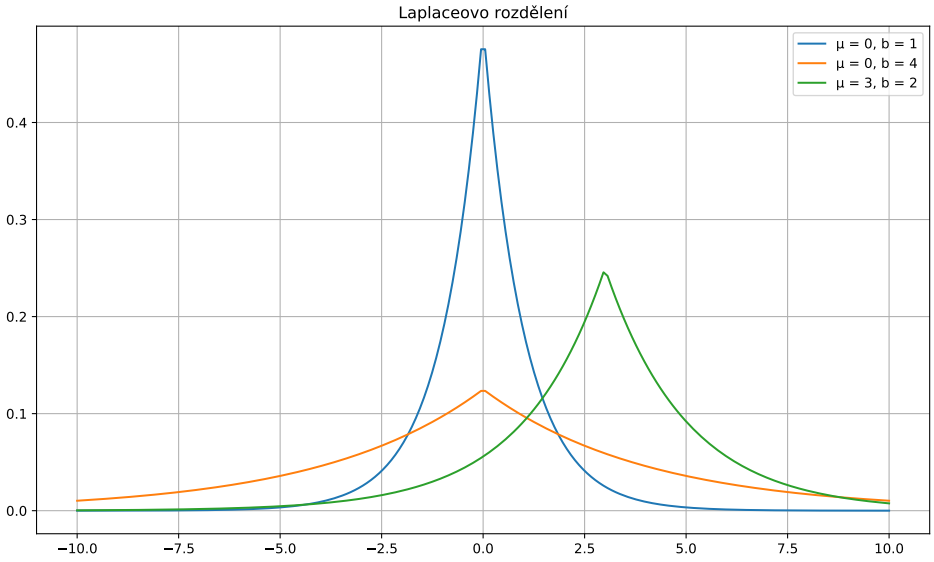
U modelu 1 je například váha w_3 relativně malá, ve srovnání s w_4 nebo w_5. Obě tyto váhy by však v našem modelu neměly být, jelikož jsou přebytečné. Můžeme si zde také povšimnout, jak s rostoucím λ klesají hodnoty absolutní hodnoty vah.

5.3 Lasso regrese

Nyní si pojďme představit další z regresních modelů. Jedná o velice podobný případ jako ridge regrese. Rozdíl bude pouze v tvaru apriorního rozdělení, kde nyní místo Gaussovy hustoty pravděpodobnosti použijeme Laplaceovu hustotu pravděpodobnosti. Ta má tvar

$$p(x|\mu, b) = \frac{1}{2b} exp\left(-\frac{|x-\mu|}{b}\right), \tag{5.5}$$

kde μ je reálný parametr a $b > 0$. Hustota pak vypadá následovně.



Obrázek 5.6: Příklady různých Laplaceových distribučních funkcí

Ačkoli jsou obě regrese velice podobné, je mezi nimi jeden důležitý rozdíl. V případě rigde regrese penalizujeme především hodnoty velmi vzdálené od naší predikované regresní funkce. To je zapříčiněno volbou Gaussovy apriorní hustoty. Ta totiž vzdálenosti mezi body a funkční hodnotou predikované regresní funkce mocní na druhou. To znamená, že hodně odchýlené hodnoty nám budou výrazně ovlivňovat ztrátovou funkci, oproti tomu odchylky blízké nule se ještě zmenší a ztrátovou funkci budou ovlivňovat minimálně. Lasso regrese na duhou stranu odchylky nijak nemění. Tedy bere rozdíly mezi predikovanou funkční hodnotou a daty beze změny. To nám ve výsledku zapříčiní to, že lasso regrese bude v modelu schopna lépe nulovat nevýznamné váhy.

Pojďme si tedy sestavit model. Začneme tvarem sdružené hustoty pravděpodobnosti.

$$p(y,w) = p(y,w|x,\beta,\lambda) = p(y|x,w,\beta)\,p(w|\lambda) = N_n(\,y\,|\,F(x)w\,,\,\beta^{-1}\mathbb{I})\,\mathrm{Lap}_{k+1}(\,w\,|\,0\,,\,\lambda^{-1}\mathbb{I})$$

Zde $\mathrm{Lap}_{k+1}(\,w\,|\,0\,,\,\lambda^{-1}\mathbb{I})$ představuje sdruženou Laplaceovu hustotu pravděpodobnosti.

Nyní můžeme opět aplikovat metodu maximální věrohodnosti.

$$L(y,w) = \prod_{i=1}^{n} p(y_i|x_i,w,\beta) \prod_{i=0}^{k} p(w|\lambda)$$

$$l(y,w) = \log L(y,w) = \frac{n}{2}\log(\frac{\beta}{2\pi}) - \frac{\beta}{2}\sum_{i=1}^{n}(y_i - F(x_i)w) + n\log(\frac{\lambda}{2}) - \lambda\sum_{i=0}^{k}|w|$$

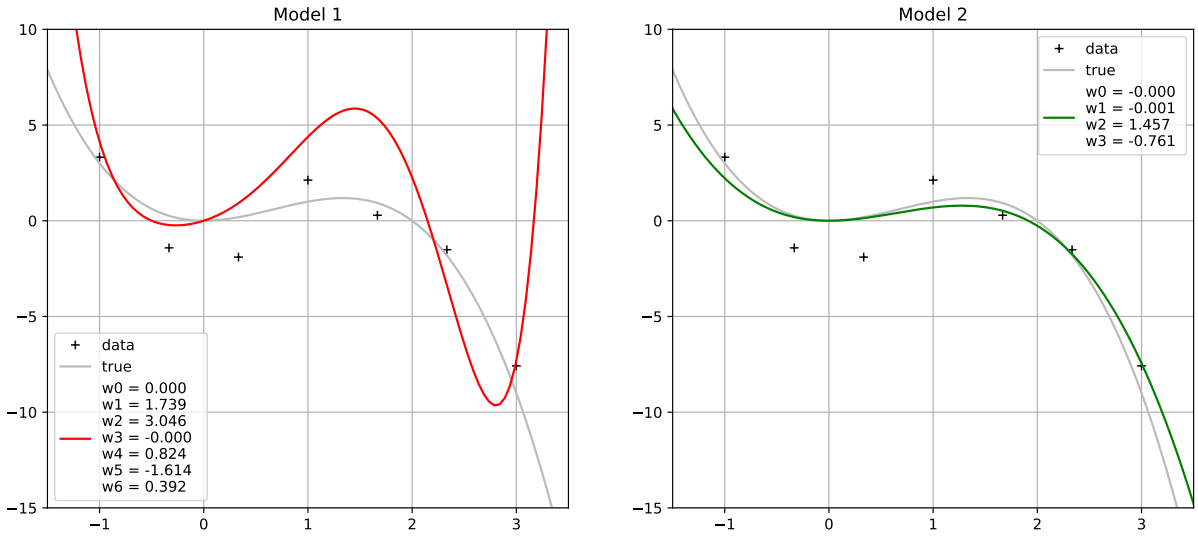
Pro nalezení extrému funkce ji zderivujeme parciálně podle vah a výsledný výraz položíme roven nule. Následně jej pak můžeme přepsat do maticové podoby.

$$F(x)^T(y - F(x)w) - \lambda\mathrm{sgn}(w) = 0$$

Z tohoto výrazu jde vidět, že v tomto případě se bohužel už nepodaří explicitní vyjádření vah w. Budeme se tedy muset spokojit se ztrátovou funkci, která při konstantním β a λ má následující tvar.

$$E(w) = \frac{\beta}{2}\sum_{i=1}^{n}(y_i - F(x_i)w)^2 + \lambda\frac{1}{2}\sum_{i=0}^{k}|w_i|.$$

Pro nalezení minima funkce lze použít některou z iterativních metod. Nový model pak opět aplikujeme na původní data.



Obrázek 5.7: Dvě regresní křivky vygenerované za pomoci lasso regrese při $\lambda = 3$. První s přebytečným počtem bazických funkci - 7. Druhá s přiměřeným počtem bazických funkcí - 4. Data generujeme z $y_i = 2x_i^2 - x_i^3 + e_i$.

Ačkoli v případě modelu 1 nedostáváme zcela ideální výsledky, u modelu 2, který má pouze dva přebytečné parametry, se podařilo nevýznamné váhy téměř vynulovat. Významné váhy pak mají o něco menší hodnotu než váhy původní. To je zapříčiněno tím, že stejně jako v předchozím případě, absolutní hodnoty vah s rostoucím λ klesají. Tato vlastnost lasso regrese je velice vítaná.

Je zřejmé, že ztrátové funkce ridge regrese a lasso regrese mají velice podobný tvar. Do ztrátové funkce můžeme přidat mocninu k apriornímu členu, kterou můžeme ovlivňovat citlivost na odchylku v datech.

$$E(w) = \frac{1}{2}\sum_{i=1}^{n}(y_i - F(x_i)w)^2 + \lambda\frac{1}{2}\sum_{i=0}^{k}|w_i|^q.$$

Čím větší q pak budeme volit, tím větší penalizace postihne data hodně vzdálené od daného modelu. Na druhou stranu, pokud budeme q zmenšovat, nevýznamné váhy budou nulovány s větším důrazem.

5.4 ARD regrese

Zkratka ARD je odvozena z anglického *automatic relevance determination*, tedy automatické stanovení relevance. To znamená, že model, který připravíme, by měl být schopen automaticky nulovat

nevýznamné váhy. Mějme opět soubor vysvětlujících proměnných $x = [x_1, \ldots, x_n]^T$ a k tomu odpovídající naměřená data $y = [y_1, \ldots, y_n]$. Stejně jako při předchozích modelech budeme u dat předpokládat Gaussovskou chybu. Znovu si zvolíme bazické funkce $[f_1(x), \ldots, f_k(x)\ ]$. V následujícím modelu budeme vycházet z ridge regrese, u které jsme předpokládali následující sdruženou hustotu pravděpodobnosti.

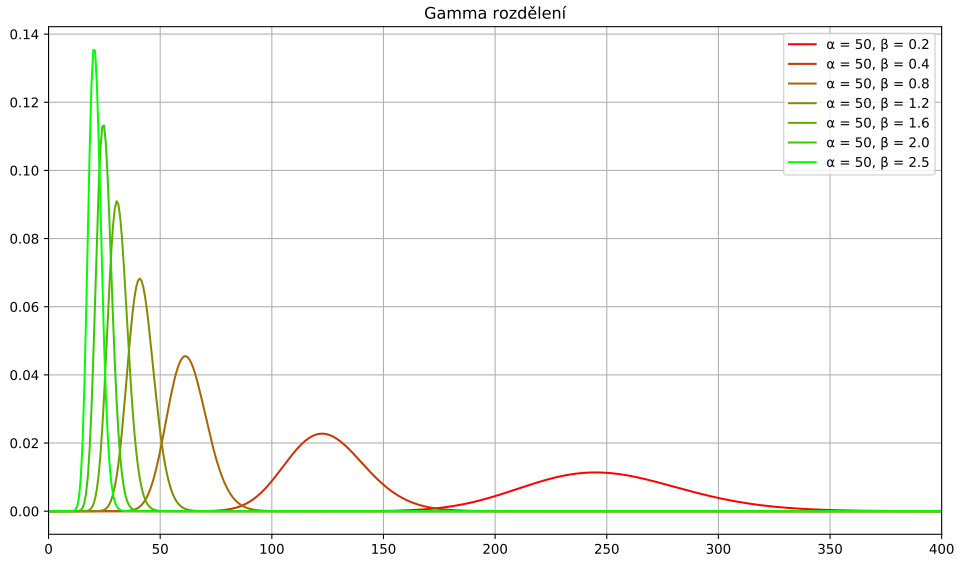
$$p(y, w) = p(y|x, w, \beta)\, p(w|\lambda) = N_n(\, y \,|\, F(x)w\,,\, \beta^{-1}\mathbb{I})\, N_{k+1}(\, w \,|\, 0\,,\, \lambda^{-1}\mathbb{I})$$

Nevýhodou pak bylo, že parametr λ, který ovlivňoval výslednou regresi, jsme museli volit sami. Další nevýhodou bylo, že parametr ovlivňoval všechny váhy stejným způsobem. To se v tomto modelu pokusíme napravit.

Uděláme to tím způsobem, že místo skaláru bude $\lambda \in \mathbb{R}^{k+1}$ a namísto deterministické proměnné bude, stejně jako váhy w, považován za náhodnou veličinu. Nyní je na nás, jaké rozdělení si pro tuto nově vzniklou náhodnou veličinu zvolíme. Jediným omezením bude požadavek na nezápornost parametru λ ve všech jeho složkách. Proto je například Gaussovo rozdělení nevhodné. My si pro model zvolíme gamma rozdělení

$$p(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} \cdot (x)^{\alpha+1} \cdot \exp(-\beta x),$$

které budeme značit $p(x) = \text{Gamma}(x \,|\, \alpha, \beta)$. Pro názornost si můžeme vykreslit hustoty pravděpodobnosti gamma rozdělení pro různé parametry α a β.



Obrázek 5.8: Příklady různých hustot pravděpodobnosti popisující gamma rozdělení.

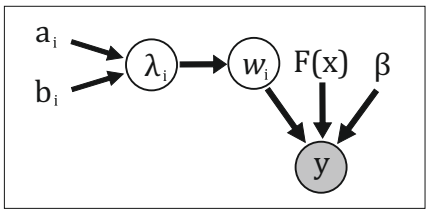
Náš model tedy bude mít následnou volbu věrohodnosti a apriorních hustot

$$p(y \mid w) = N_n(\, y \mid F(x)w \,,\, \beta^{-1}\mathbb{I})$$

$$p(w \mid \lambda) = \prod_{i=0}^{k} N(\, w_i \mid 0 \,,\, \lambda_i^{-1})$$

$$p(\lambda) = \prod_{i=0}^{k} \mathrm{Gamma}(\, \lambda_i \mid \alpha_i \,,\, \beta_i).$$

Závislost jednotlivých proměnných si můžeme znázornit na následujícím schématu. Proměnné v kroužku jsou ty, které považujeme za náhodné, oproti tomu samostatně stojící proměnné jsou deterministické.



Obrázek 5.9: Schema ARD regrese

Zde se na místě si položit otázku, jestli chceme na parametry α a β klást nějaké požadavky. Ty totiž, pokud bychom je nechali pouze jako další parametry ztrátové funkce, by nijak nepřispívaly k našemu kýženému výsledku. Je proto potřeba jejich hodnoty nějak provázat s hodnotami jednotlivých vah. Našim záměrem je vynulovat nevýznamné váhy. To jsme doposud realizovali přes parametr λ. Ten funguje tak, že čím je větší jeho hodnota, tím více tlačí jednotlivé váhy k nule. Z obrázku (5.4) můžeme na druhou stranu vidět, že s rostoucími hodnotami parametru β klesá argument maximální hodnoty gamma hustoty. Toho můžeme u maximálně věrohodných odhadů vah w využít. Hodnotu parametru nastavíme

$$\beta_i = \beta_0 + \frac{1}{2}w_i^2 \quad \text{pro} \quad \forall i \in 0, \ldots, k,$$

tím pak docílíme toho, že čím menší hodnota váhy bude, tím více ji gamma rozdělení bude tlačit do nuly. Hodnoty parametrů α_i obvykle volíme všechny stejné. Výslednou ztrátovou funkci $E(w, \lambda)$ můžeme získat zlogaritmováním sdružené hustoty pravděpodobnosti. Její jednotlivé členy pak mají tvar

$$\log p(y|w) = -\frac{n}{2}\log\frac{\beta}{2\pi} - \frac{\beta}{2}\left(\sum_{i}^{n}(F_i w - y_i)^2\right)$$

$$\log p(w|\lambda) = \frac{1}{2}\sum_{i=0}^{k}\log 2\pi\lambda_i - \frac{1}{2}\sum_{i=0}^{k}\lambda_i w_i^2$$

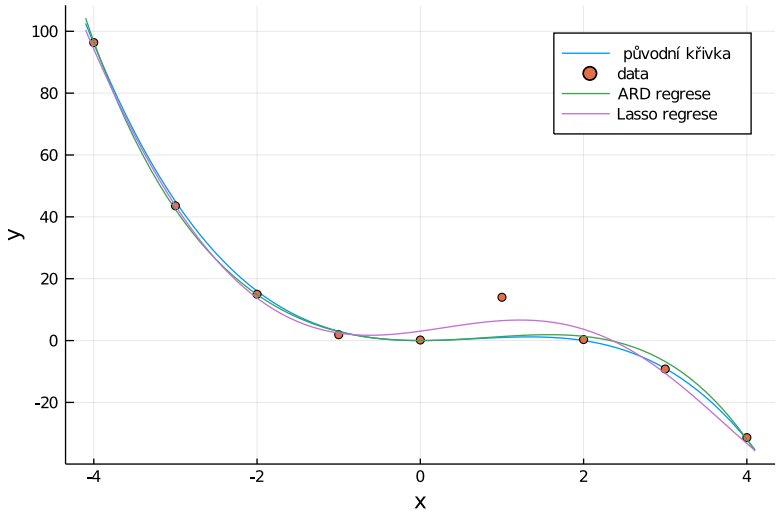
$$\log p(\lambda) = \sum_{i=0}^{k} a_0 \log(b_0 + \frac{1}{2}w_i^2) + \sum_{i=0}^{k}(a_0 - 1)\log\lambda_i - \sum_{i=0}^{k}(b_0 + \frac{1}{2}w_i^2)\lambda_i - \sum_{i=0}^{k}\log\Gamma(a_0).$$

Zde členy, které nezávisí na λ nebo w můžeme vynechat. Výsledná ztrátová funkce bude mít tvar

$$E(w,\lambda) = -\frac{\beta}{2}\left(\sum_{i}^{n}(F_i w - y_i)^2\right) + \frac{1}{2}\sum_{i=0}^{k}(\log(2\pi\lambda_i) - \lambda_i w_i^2) \\ + \sum_{i=0}^{k}(a_0\log(b_0 + \frac{1}{2}w_i^2) + (a_0 - 1)\log\lambda_i - (b_0 + \frac{1}{2}w_i^2)\lambda_i). \tag{5.6}$$

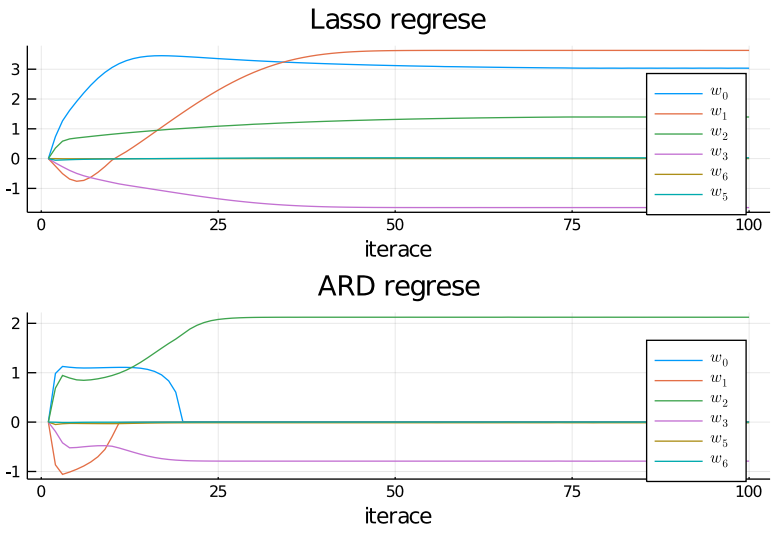
Tuto funkci potřebujeme maximalizovat přes parametry w a λ. K tomu nám poslouží nějaký vhodně zvolený optimalizační algoritmus.

Pojďme si tuto metodu demonstrovat na příkladu. Mějme podobný soubor dat s normální chybou, jako u předchozích regresních modelů. Nyní však přidáme nějakou hodně odchýlenou hodnotu. Na následujícím grafu pak můžeme vidět, jak ARD regrese dopadne.



Obrázek 5.10: ARD regresní křivka při volbě $b_0 = 0.007$, $a_0 = 30$. Lasso regresní křivka při volbě $\lambda = 5$. Obě pro bazické funkce polynomy 0.–6. stupně.

Lépe lze efekt ARD regrese vidět z průběhu učení. Z grafu níže můžeme vidět, že pouze dvě váhy nekonvergují k nule. Z toho vyplývá, že jsme našli model, kterému pro popis dat stačí pouze dvě bazické funkce. Oproti tomu Lasso regrese považuje čtyři bazické funkce za významné.



Obrázek 5.11: Průběh učení jednotlivých metod u ARD regrese (5.6) ve srovnání s Lasso regresí (5.3).

Kapitola 6

Hledání struktury diferenciálních rovnic

V této kapitole se budeme snažit generovat diferenciální rovnice z experimentálních dat při minimální znalosti fyzikálního systému, z kterého byly data získány. Připravíme si skupiny diferenciálních rovnic, pokrývající velké množství fyzikálních systémů. Naším úkolem pak bude najít ten, který data popisuje nejlépe.

Při tvorbě těchto modelů bylo nahlíženo do [3].

6.1 Nejmenší čtverce pro ODE

V následující kapitole se budeme zabývat aproximací experimentálních dat z různých fyzikálních systémů za pomoci soustav diferenciálních rovnic. Jejich hodnoty obvykle obsahují náhodné chyby, které se zde mohou objevit například z důvodu nepřesnosti měření. My v této kapitole budeme předpokládat, že chyba experimentálních dat má Gaussovské rozdělení.

Mějme tedy soubor experimentálních dat $D = [d_0, ..., d_n]$ z nějakého fyzikálního systému, který lze popsat soustavou diferenciálních rovnic. Předpokládejme dále, že soustava diferenciálních rovnic, která systém popisuje, může být zapsána v následujícím tvaru

$$\frac{\partial \xi(t)}{\partial t} = \widehat{W} \cdot f(t, \xi(t)) \quad \text{spolu s p. p.} \quad \xi(0) = \xi_0(\widehat{\theta}), \tag{6.1}$$

kde $\widehat{W} \in \mathbb{R}^{k \times l}$, $\xi(t) \in \mathbb{R}^k$ a $f(t, \xi(t)) \in \mathbb{R}^l$ pro $l, k \in \mathbb{N}$. Funkce $f(t, \xi(t))$ zde plní podobnou roli jako bazické funkce v kapitole lineární regrese. Matice $\widehat{W}$ pak obvykle bývá řídká, což zapříčiní vybírání pouze některých funkcí pro odpovídající diferenciální člen $\frac{\partial \xi_i(t)}{\partial t}$.

Pro řešení ODE si vybereme nějakou vhodnou numerickou metodu. Její výstup si pak označíme následovně

$$y_i(\widehat{W}) = y(t_i, \widehat{W}) = S_{\text{diff}}[h, n, t_0, \xi_0, \widehat{W} \cdot f(t, \xi(t))]_i \quad \text{kde} \quad i \in 0, \ldots, n,$$

kde funkce S_{diff} je tedy nějaký řešič obyčejné diferenciální rovnice. V našem případě se pak obvykle bude jednat o Rungovy-Kuttovy metody 4. nebo 5. řádu. Parametr t_0 pak vyjadřuje počáteční bod řešení ODE. Parametr n vyjadřuje počet bodů, ve kterých chceme rovnici řešit a parametr h představuje vzorkovací krok. Celou diferenciální rovnici řešíme v bodech $t_i = t_0 + i \cdot h$. U jednotlivých experimentálních dat předpokládáme následující tvar

$$d_i = y_i(\widehat{W}) + e_i \quad \text{kde} \quad i \in 0, \ldots, n, \tag{6.2}$$

kde $e_i \sim N(0, \omega)$. Jelikož u dat předpokládáme Gaussovskou chybu, můžeme jejich pravděpodobnostní rozdělení popsat hustotou pravděpodobnosti

$$p(d_i) = N(d_i|y_i(\widehat{W}), \omega).$$

Pro nalezení odhadu W skutečných hodnot váhové matice $\widehat{W}$, můžeme využít metodu maximální věrohodnosti

$$W_{ML} = \arg\sup_W p(D, W) = \arg\sup_W \left(\prod_{i=1}^{n} \frac{1}{2\pi\omega} \exp\left(\frac{(d_i - y_i(W))^2}{2\omega} \right) \right).$$

Tato úloha je pak ekvivalentní k minimalizování rozdílu čtverců dat a řešení diferenciální rovnice. Úlohu tedy můžeme převést na minimalizování ztrátové funkce

$$L(W) = \|D - Y(W)\|_2^2 = \sum_{i=1}^{n} (d_i(\widehat{W}) - y_i(W))^2 \tag{6.3}$$

za pomocí nějaké z optimalizačních metod. U těch zpravidla požadujeme, aby funkce $y(W)$ byla diferencovatelná a mi tak mohli spočítat její gradient. To je zajištěno z linearity soustavy diferenciálních rovnic vůči váhové matici W.

Pro metodu s takovouto ztrátovou funkcí při použití obecného řešiče ODE budeme dále používat zkratku **LS-ODE**.

Tato metoda je vhodná zejména pro úlohy, u kterých známe přesný tvar ODE. Známe tedy funkce $f(t, \xi(t))$, které jsou pro popsání dané úlohy nutné. Rovněž také víme, které členy matice $\widehat{W}$ jsou nulové. V takovém případě můžeme metodu modifikovat a použít optimalizační algoritmy pouze vzhledem k nenulovým w_{ij}. V opačném případě, kdy přesný tvar ODE znát nebudeme, bude velmi pravděpodobné, že výsledná váhová matice W, aproximující skutečnou matici $\widehat{W}$, bude převážně nenulová. Pro vysvětlení dat tedy budeme nacházet zbytečně složité modely.

6.2 Lasso model pro ODE

V předchozí sekci jsme si představili metodu pro hledání neznámých parametrů v soustavách diferenciálních rovnic vysvětlujících experimentální data. Tento přístup ovšem v obecném případě nachází zbytečně složité modely. To se pokusíme následující metodou napravit.

Mějme totožné zadání úlohy jako v předchozí sekci (6.1). U té jsme díky předpokládanému tvaru dat popsali jejich pravděpodobnostní rozdělení za pomoci hustoty

$$p(d_i) = N(d_i|y_i(W), \omega).$$

Pokud budeme W považovat také za náhodnou veličinu, můžeme do modelu přidat apriorní člen $p(W)$. Jelikož chceme, aby náš model, vysvětlující data, byl co možná nejjednodušší, budeme apriorní člen volit tak, abychom upřednostnili nulovou hodnotu jednotlivých w_{ij}. Vhodnou volbou se může zdát například Laplaceova hustota pravděpodobnosti

$$p(w_{ij}) = \mathrm{Lap}(w_{ij}|0, \lambda^{-1}) \quad \text{pro} \quad \forall i \in 1, \ldots, k \ , \ \forall j \in 1, \ldots, l.$$

Z těchto výrazů pak můžeme získat maximálně věrohodný odhad váhové matice $\widehat{W}$.

$$W_{ML} = \arg\sup_W p(D, W) = \arg\sup_W \left(\prod_{i=1}^{n} \frac{1}{2\pi\omega} \exp\left(\frac{(d_i - y_i(W))^2}{2\omega} \right) \prod_{i=1}^{k} \prod_{j=1}^{l} \frac{\lambda}{2} \exp(\lambda \cdot |w_{ij}|) \right).$$

Zde se opět nepodaří získat explicitní vyjádření W_{ML}. Můžeme se ale pokusit tento odhad najít za pomoci některé z optimalizačních metod. Pro ty je vhodnější výraz výše zlogaritmovat. Při následném odstranění členů neobsahujících W získáme nový tvar ztrátové funkce

$$L(W) = \sum_{i=1}^{n} (d_i(\widehat{W}) - y_i(W))^2 + \lambda \sum_{i=1}^{k} \sum_{j=1}^{l} |w_{ij}|., \tag{6.4}$$

Zde se parametr λ opět volí jako nějaké kladné číslo. Čím větší bude jeho zvolená hodnota, tím více budou všechny členy matice W tlačeny k nule.

Pro metodu s takovouto ztrátovou funkcí při použití nějakého obecného řešiče ODE budeme dále používat zkratku Lasso-ODE.

6.3 Normal model pro ODE

Předchozí přístup má mimo volby λ i druhou nepříjemnou vlastnost. Parametr tlačí všechny váhy k nule stejnou silou. To budeme chtít v následujícím modelu napravit.

Začneme tím, že budeme předpokládat stejný tvar diferenciální rovnice jako u předchozích dvou metod. Na rozdíl od nich, kde jsme předpokládali nějaký tvar sdružené hustoty pravděpodobnosti a následně hledali deterministický MLE odhad $y_i(\widehat{W})$, se nyní budeme snažit najít aposteriorní rozdělení $y_i(\widehat{W})$.

$$p(y(\widehat{W})|D) = \frac{p(D|y(\widehat{W})) \cdot p(\widehat{W})}{\int_{\widehat{W}} p(D|y(\widehat{W})) \cdot p(\widehat{W}) d\widehat{W}}$$

To ovšem nejsme obecně schopni spočítat kvůli výrazu ve jmenovateli. Proto se pokusíme aposteriorní rozdělení alespoň co nejlépe aproximovat. K tomu nám poslouží hustota pravděpodobnosti $q(\widehat{W})$ a Kullback-Leibler divergence.

Z kapitoly o Kullback-Leibler divergenci víme, že její minimalizace je ekvivalentní s maximalizací ELBO.

$$KL(q_\phi(\widehat{W}) || p_\theta(y(\widehat{W})|D)) \rightarrow \min_{\phi,\theta}$$

$$\Longleftrightarrow$$

$$L(\phi, \theta) = \int q_\phi(\widehat{W}) \log \frac{p_\theta(D, y(\widehat{W}))}{q_\phi(\widehat{W})} d\widehat{W} = \int q_\phi(\widehat{W}) \log \frac{p_\theta(D|y(\widehat{W})) \cdot p_\theta(\widehat{W})}{q_\phi(\widehat{W})} d\widehat{W} \rightarrow \max_{\phi,\theta}$$

Pro jednodušší zápis budeme matice $W, \widehat{W} \in \mathbb{R}^{k \times l}$ občas považovat za vektory. To uděláme tím způsobem, že jednotlivé složky přepíšeme do sloupce. Výsledné vektory jsou $W, \widehat{W} \in \mathbb{R}^m$, kde $m = k \times l$. Z kontextu by pak mělo být jasné, o který objekt se jedná.

Zvolme si nyní věrohodnost a apriorní člen

$$p_\theta(d_i|y_i(\widehat{W})) = N(d_i|y_i(\widehat{W}), \omega_0) \quad \text{pro} \quad i \in 0, \ldots, n$$
$$p_\theta(\widehat{W}) = N_m(\widehat{W}|0, \tau^{-1}\mathbb{I}),$$

kde $\tau^{-1} \in \mathbb{R}^m$ a rozptyl ω_0 předpokládáme za známý. Volba věrohodnostního členu opět plyne z předpokládaného tvaru experimentálních dat. Apriorní člen pak vyjadřuje upřednostnění co nejjednoduššího modelu. Připomeňme si předpokládaný tvar dat

$$d_i(\widehat{W}) = y_i(\widehat{W}) + e_i = y(t_i, \widehat{W}) + e_i = S_{\text{diff}}[h, n, t_0, \xi_0, \widehat{W} \cdot f(t, \xi(t))]_i + e_i \quad \text{kde } e_i \sim N(0, \omega_0). \tag{6.5}$$

Aproximační hustotu si zvolíme z Gaussovského rozdělení

$$q_\phi(\widehat{W}) = \prod_{i=1}^{m} N(\widehat{W}_i|W_i, \sigma_i).$$

Tvar $L(\phi, \theta) = L(W, \sigma, \tau)$ můžeme upravit následovně.

$$L(W, \sigma, \tau) = \mathbb{E}_{q_\phi(\widehat{W})}[\log p_\theta(D|y(\widehat{W}))] + \mathbb{E}_{q_\phi(\widehat{W})}[\log p_\theta(\widehat{W})] - \mathbb{E}_{q_\phi(\widehat{W})}[\log q_\phi(\widehat{W})].$$

Jednotlivé členy si můžeme rozepsat

$$\begin{aligned}
\mathbb{E}_{q_\phi(\widehat{W})}[\log p_\theta(D|y(\widehat{W}))] &= -\frac{n}{2}\log(2\pi\omega_0) - \frac{1}{2\omega_0}\left(\sum_{i=1}^{n} d_i^2 - 2\sum_{i=1}^{n} d_i \mathbb{E}_{q(\widehat{W})}[y_i(\widehat{W})] + \sum_{i=1}^{n} \mathbb{E}_{q(\widehat{W})}[(y_i(\widehat{W}))^2]\right) \\
\mathbb{E}_{q_\phi(\widehat{W})}[\log p_\theta(\widehat{W})] &= -\frac{m}{2}\log 2\pi + \frac{1}{2}\sum_{i=1}^{m} \log\tau_i - \frac{1}{2}\sum_{i=1}^{m} \tau_i \mathbb{E}_{q(\widehat{W})}[\widehat{W}^2] \\
\mathbb{E}_{q_\phi(\widehat{W})}[\log q_\phi(\widehat{W})] &= -\frac{1}{2}\sum_{i=1}^{m} \log 2\pi\sigma_i - \frac{m}{2}.
\end{aligned} \tag{6.6}$$

Zde bohužel výraz $\mathbb{E}_{q(\widehat{W})}[y_i(\widehat{W})]$ ani $\mathbb{E}_{q(\widehat{W})}[(y_i(\widehat{W}))^2]$ neumíme spočítat. Proto budeme muset použít reparametrizační trik (4.3). Pro náš případ bude nahrazení $\widehat{W}$ vypadat následovně.

$$\widehat{W} = W + \sigma \odot e \quad \text{kde} \quad e \sim N_m(0, 1)$$

Po nahrazení dostaneme následující tvar jednotlivých členů.

$$\begin{aligned}
\mathbb{E}_{q_\phi(\widehat{W})}[\log p_\theta(D|y(\widehat{W}))] &= -\frac{n}{2}\log(2\pi\omega_0) - \frac{1}{2\omega_0}\left(\sum_{i=1}^{n} d_i^2 - 2\sum_{i=1}^{n} d_i y_i(W + \sigma \odot e) + \sum_{i=1}^{n} (y_i(W + \sigma \odot e)^2\right) \\
\mathbb{E}_{q_\phi(\widehat{W})}[\log p_\theta(\widehat{W})] &= -\frac{m}{2}\log 2\pi + \frac{1}{2}\sum_{i=1}^{m} \log\tau_i - \frac{1}{2}\sum_{i=1}^{m} \tau_i (W_i + \sigma_i \odot e_i))^2 \\
\mathbb{E}_{q_\phi(\widehat{W})}[\log q_\phi(\widehat{W})] &= -\frac{1}{2}\sum_{i=1}^{m} \log 2\pi\sigma_i - \frac{m}{2}
\end{aligned} \tag{6.7}$$

Tyto výrazy už lze spočítat. Máme tedy nový tvar ztrátové funkce $-L(W, \sigma, \tau)$, kterou chceme minimalizovat. Její hodnota ovšem také závisí na náhodné proměnné e. Z toho vyplývá, že i hodnota ztrátové funkce i její gradient jsou náhodné. Při její optimalizaci pak není zajištěn monotónní sestup. To však nevadí, jelikož v průměru dostáváme správný gradient. To nás postupně, při použití optimalizačního algoritmu s adaptivní délkou kroku (např. ADAM), dostane až k minimu ztrátové funkce.

Pro danou volbu optimalizačního algoritmu potřebujeme výchozí hodnoty. Ty teoreticky můžeme volit jakkoli. Praxe však ukazuje, že výsledek metody může na počátečním stavu záviset. To je zapříčiněno tím, že v obecném případě může mít funkce $L(\phi, \theta)$ velké množství lokálních extrémů. V takovém případě se pak může stát, že globálního minima, při určité volbě počátečního bodu, nemusíme dosáhnout.

Pro metodu s takovouto ztrátovou funkcí při použití nějakého obecného řešiče ODE budeme dále používat zkratku **ELBO-N-ODE**.

6.4 Normal-iGamma model pro ODE

V této sekci si představíme určitou modifikaci předchozího modelu za pomoci inverzního gamma rozdělení. Model ELBO-N-ODE obsahoval skupinu parametrů τ_i, které tlačili odpovídající váhy w_i k nule různou silou. Na parametr ovšem nebyly kladeny žádné nároky. To se dá volně interpretovat jako stav, kdy byla nulová hodnota preferována, nikoli však vyžadována. Tento nedostatek se pokusíme v následujícím modelu napravit.

Opět si zvolíme věrohodnostní a apriorní členy

$$\begin{aligned} p(d_i|y_i(\widehat{W})) &= N(d_i|y_i(\widehat{W}), \omega_0) \\ p(\widehat{W}_j|\tau_j) &= N(\widehat{W}_j|0, \tau_j) \\ p(\tau_j) &= \tau_j^{-1/2}, \end{aligned}$$

kde $i \in \{0, \dots, n\}$ a $j \in \{0, \dots, m\}$. Zde volba věrohodnostního členu opět vyplývá z předpokládaného tvaru dat. První apriorní člen vyjadřuje preferenci nulové hodnoty. Druhý, nově přidaný, apriorní člen zde funguje jako jistá penalizace za příliš nízké hodnoty τ_i. Za zmínku stojí, že se vlastně nejedná o hustotu pravděpodobnosti.

Aproximační hustotu si zvolíme nově ve tvaru $q_\phi(\widehat{W}, \tau) = q_\phi(\widehat{W}) \cdot q_\phi(\tau)$, kde

$$\begin{aligned} q_\phi(\widehat{W}) &= \prod_{i=1}^{m} N(\widehat{W}_i|W_i, \sigma_i) \\ q_\phi(\tau) &= \prod_{i=1}^{m} \text{Inv-Gamma}(\tau_i|a_i, b_i). \end{aligned}$$

Tato volba aproximačních hustot a apriorních členů vychází z [8]. Je možné nalézt jistou podobnost s ARD regresí 5.4, která podobně jako tento model nuluje členy jednotlivě.

Nyní budeme opět minimalizovat KL divergenci za pomocí ELBO

$$\begin{aligned} L(W, \sigma, a, b) =& \mathbb{E}_{q_\phi(\widehat{W})}[\log p(D|y(\widehat{W}))] + \mathbb{E}_{q_\phi(\widehat{W},\tau)}[\log p(\widehat{W}|\tau)] + \mathbb{E}_{q_\phi(\tau)}[\log p(\tau)] \\ & - \mathbb{E}_{q_\phi(\widehat{W})}[\log q_\phi(\widehat{W})] - \mathbb{E}_{q_\phi(\tau)}[\log q_\phi(\tau)]. \end{aligned}$$

Jednotlivé členy pak mají tvar

$$\begin{aligned} \mathbb{E}_{q_\phi(\widehat{W})}[\log p_\theta(D|y(\widehat{W}))] &= -\frac{n}{2}\log(2\pi\omega_0) - \frac{1}{2\omega_0}\left(\sum_{i=1}^{n} d_i^2 - 2\sum_{i=1}^{n} d_i \mathbb{E}_{q(\widehat{W})}[y_i(\widehat{W})] + \sum_{i=1}^{n} \mathbb{E}_{q(\widehat{W})}[(y_i(\widehat{W}))^2]\right) \\ \mathbb{E}_{q_\phi(\widehat{W},\tau)}[\log p(\widehat{W}, \tau)] &= -\frac{m}{2}\log 2\pi - \frac{1}{2}\sum_{i=1}^{m} \mathbb{E}_{q_\phi(\tau)}[\log(\tau_i)] - \frac{1}{2}\sum_{i=1}^{m} \mathbb{E}_{q_\phi(\tau_i)}[1/\tau_i]\mathbb{E}_{q(\widehat{W})}[\widehat{W}^2] \\ \mathbb{E}_{q_\phi(\tau)}[\log p(\tau)] &= -\frac{1}{2}\sum_{i=1}^{m} \mathbb{E}_{q_\phi(\tau)}[\log(\tau_i)] \\ \mathbb{E}_{q_\phi(\widehat{W})}[\log q_\phi(\widehat{W})] &= -\frac{1}{2}\sum_{i=1}^{m} \log 2\pi\sigma_i - \frac{m}{2} \\ \mathbb{E}_{q_\phi(\tau)}[\log q_\phi(\tau)] &= \sum_{i=1}^{m} \Big(a_i \log(b_i) - \log(\Gamma(a_i)) - (a_i + 1)\mathbb{E}_{q_\phi(\tau)}[\log(\tau_i)] - b_i \mathbb{E}_{q_\phi(\tau_i)}[1/\tau_i]\Big). \end{aligned} \tag{6.8}$$

Nyní opět použijeme reparametrizační trik, jelikož výrazy $\mathbb{E}_{q(\widehat{W})}[y_i(\widehat{W})]$ a $\mathbb{E}_{q(\widehat{W})}[(y_i(\widehat{W}))^2]$ neumíme spočítat.

$$\widehat{W} = W + \sigma \odot e \quad \text{kde} \quad e \sim N_m(0,1)$$

Dále si nahradíme střední hodnoty inverzního gamma rozdělení za pomoci následujících výrazů

$$\mathbb{E}_{q_\phi(\tau_i)}[1/\tau_i] = \frac{a_i}{b_i} \qquad \mathbb{E}_{q_\phi(\tau)}[\log(\tau_i)] = \log(b_i) - \Psi(a_i).$$

Při dosazení dostaneme následující tvar jednotlivých členů

$$\begin{aligned}
\mathbb{E}_{q_\phi(\widehat{W})}[\log p_\theta(D|y(\widehat{W}))] &= -\frac{n}{2}\log(2\pi\omega_0) - \frac{1}{2\omega_0}\left(\sum_{i=1}^{n} d_i^2 - 2\sum_{i=1}^{n} d_i y_i(W+\sigma\odot e) + \sum_{i=1}^{n}(y_i(W+\sigma\odot e)^2\right) \\
\mathbb{E}_{q_\phi(\widehat{W},\tau)}[\log p(\widehat{W},\tau)] &= -\frac{m}{2}\log 2\pi - \frac{1}{2}\sum_{i=1}^{m}(log(b_i) - \Psi(a_i)) - \frac{1}{2}\sum_{i=1}^{m}\frac{a_i}{b_i}(W+\sigma\odot e)^2 \\
\mathbb{E}_{q_\phi(\tau)}[\log p(\tau)] &= -\frac{1}{2}\sum_{i=1}^{m}(log(b_i) - \Psi(a_i)) \\
\mathbb{E}_{q_\phi(\widehat{W})}[\log q_\phi(\widehat{W})] &= -\frac{1}{2}\sum_{i=1}^{m}\log 2\pi\sigma_i - \frac{m}{2} \\
\mathbb{E}_{q_\phi(\tau)}[\log q_\phi(\tau)] &= \sum_{i=1}^{m}\left(a_i\log(b_i) - \log(\Gamma(a_i)) - (a_i+1)(log(b_i) - \Psi(a_i)) - a_i\right).
\end{aligned} \tag{6.9}$$

Tímto docílíme nového tvaru ztrátové funkce $-L(W,\sigma,a,b)$. Tuto funkci pak můžeme minimalizovat vůči jejím parametrům za použití některé z optimalizačních metod. Jelikož se zde ale opět jedná o určitou formu stochastického gradientu, je vhodné volit optimalizační metodu s adaptivní délkou kroku.

Pro metodu s takovouto ztrátovou funkcí při použití obecného řešiče ODE, budeme dále používat zkratku **ELBO-NiG-ODE**.

Kapitola 7

Výpočetní studie

V této kapitole budeme porovnávat jednotlivé metody hledání struktury diferenciálních rovnic na konkrétních úlohách.

V modelech budeme hodnotit přibližný počet iterací nutných k dosažení ustáleného stavu aproximace. Dále budeme hodnotit, jak dobře model aproximuje experimentální data. Pro jejich porovnání využijeme

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i(\widehat{W}) - y_i(W))^2}{\sum_{i=1}^{n}(y_i(\widehat{W}) - \frac{1}{n}\sum_{i=1}^{n} y_i(\widehat{W}))^2}$$

statistiku. Ta udává, jak moc daný model prostupuje daty. Čím blíže se dostaneme k hodnotě 1, tím lépe bude model data aproximovat.

Dalším ukazatelem nám bude počet nenulových w_i, který vyjadřuje složitost modelu. Čím menší tohle číslo bude, tím jednodušším modelem se nám podařilo data vysvětlit. U všech výše zmíněných modelů se téměř nikdy nepodaří w_{ij} zcela vynulovat. Proto ho budeme považovat za nulové v případě, že $|w_{ij}| < 0.05$.

U všech metod pro hledáni struktury diferenciálních rovnic budeme v této kapitole používat optimalizační algoritmus ADAM (3.2).

7.1 Lotka-Volterra ODE

První úloha, kterou budeme řešit má následující tvar

$$\frac{dx}{dt} = \widehat{\alpha}x - \widehat{\beta}xy$$
$$\frac{dy}{dt} = \widehat{\delta}xy - \widehat{\gamma}y$$

s počátečními podmínkami $[x(0), y(0)] = [x_0, y_0]$. Všimněme si, že tato soustava diferenciálních rovnic splňuje námi požadovaný tvar (6.1)

$$\frac{\partial\xi(t)}{\partial t} = \begin{bmatrix}\dot{x}(t)\\ \dot{y}(t)\end{bmatrix}, \qquad \widehat{W} = \begin{bmatrix}\widehat{w}_{11} & \widehat{w}_{12} & \widehat{w}_{13}\\ \widehat{w}_{21} & \widehat{w}_{22} & \widehat{w}_{23}\end{bmatrix} = \begin{bmatrix}\widehat{\alpha} & 0 & -\widehat{\beta}\\ 0 & -\widehat{\delta} & \widehat{\gamma}\end{bmatrix}, \qquad f(t, \xi(t)) = \begin{bmatrix}x(t)\\ y(t)\\ x(t)y(t)\end{bmatrix}.$$

Úkolem bude nalézt hodnoty členů váhové matice tak, aby se co možná nejvíc blížily skutečným hodnotám $\widehat{W}$. Tu si pro tuto úlohu zvolíme následovně:

$$\widehat{W} = \begin{bmatrix}1.5 & 0.0 & -1.0\\ 0.0 & -1.0 & 0.5\end{bmatrix}$$

Rovnici pak budeme řešit na intervalu $[0, 10]$ s počátečními podmínkami $[x(0), y(0)] = [1, 1]$. V bodech $t_i = ih$ pro $i \in \{0, ..., 20\}$ a krok $h = 0.5$. Ke zkoumaným bodům přidáme náhodnou chybu $e_i \sim N(0, \sigma^2)$, kde $\sigma = 0.2$. Počáteční stav aproximační matice W byl pro všechny metody volen jako nulová matice.

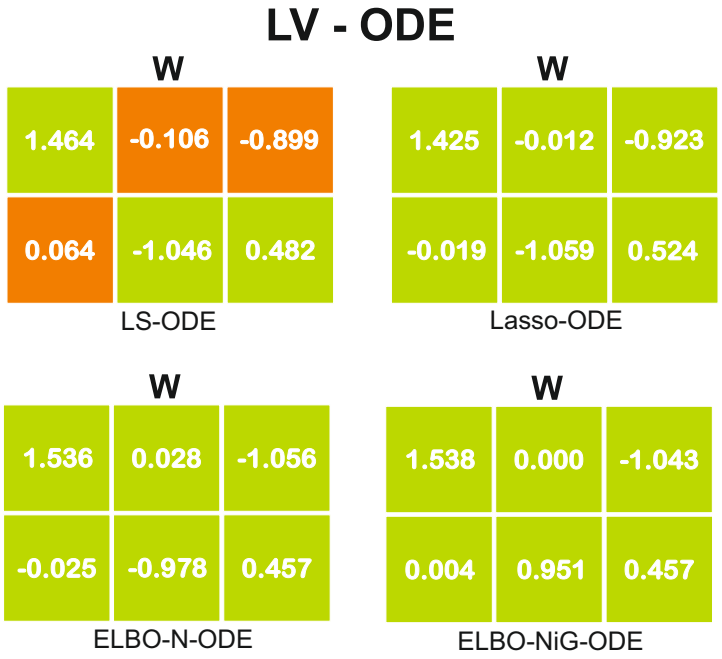
Pro jednotlivé metody dostaneme následující výsledky.

Lotkova-Volterrova ODE			
Použitá metoda	# iterací	R^2	# nenulových w_{ij}
LS-ODE	10000	0.95865	6
Lasso-ODE	10000	0.95785	4
ELBO-N-ODE	5000	0.95223	4
ELBO-NiG-ODE	30000	0.95251	4

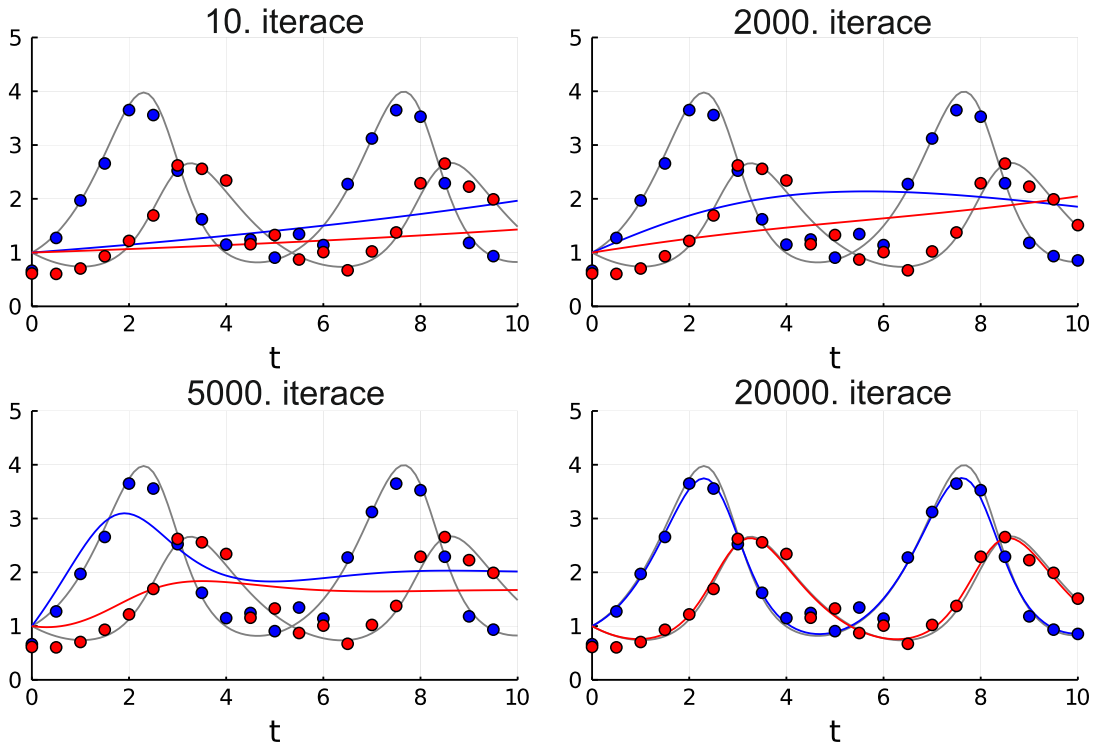
Přestože metoda **LS-ODE** dosáhla nejlepších výsledků vzhledem k R^2 statistice, celkově se ukázala jako nejméně vhodná. To především z toho důvodu, že nedokázala správně určit nulové členy váhové matice. Výsledný model má proto zbytečně vysokou složitost. Výsledky zde publikované jsou ty, kterých bylo dosaženo při volbě kroku $k = 10^3$.

Metoda **Lasso-ODE** se pro tuto úlohu ukázala jako vhodná. Nulové členy váhové matice byly správně určeny. Rovněž podle R^2 statistiky model vysvětluje data dobře. Výsledky zde zveřejněné jsou ty, kterých bylo dosaženo při volbě parametru $\lambda = 1$ a kroku $k = 10^{-3}$.

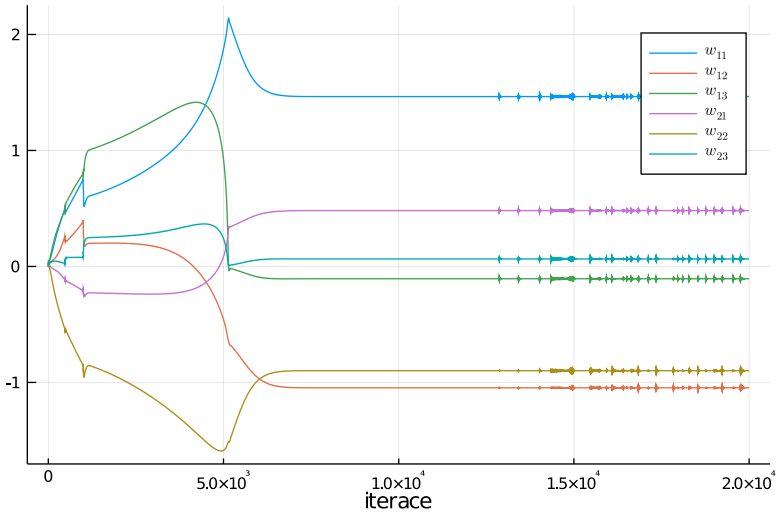
Metoda **ELBO-N-ODE** se pro tuto úlohu také ukázala jako vhodná. Členy váhové matice, které měly mít nulovou hodnotu, jsou pro tuto úlohu sice vyšší, nicméně stále splňují naše požadované kritérium $|w_{ij}| < 0.05$. R^2 statistika má vysokou hodnotu, takže model popisuje data dobře. Výsledky zde zveřejněné jsou ty, kterých bylo dosaženo při počáteční volbě parametrů $\sigma_i = 0.007$, $\tau_i = 0.1$ a kroku $k = 10^{-3}$. Metoda **ELBO-N-ODE** se pro tuto úlohu rovněž ukázala jako vhodná. Nulové členy byly v tomto případě určeny s největší přesností. Rovněž podle R^2 statistiky model vysvětluje data dobře. Výsledky zde publikované jsou ty, kterých bylo dosaženo při počáteční volbě parametrů $\sigma_i = 0.05$, $a_i = 10$, $b_i = 0.05$ a kroku $k = 10^{-3}$.



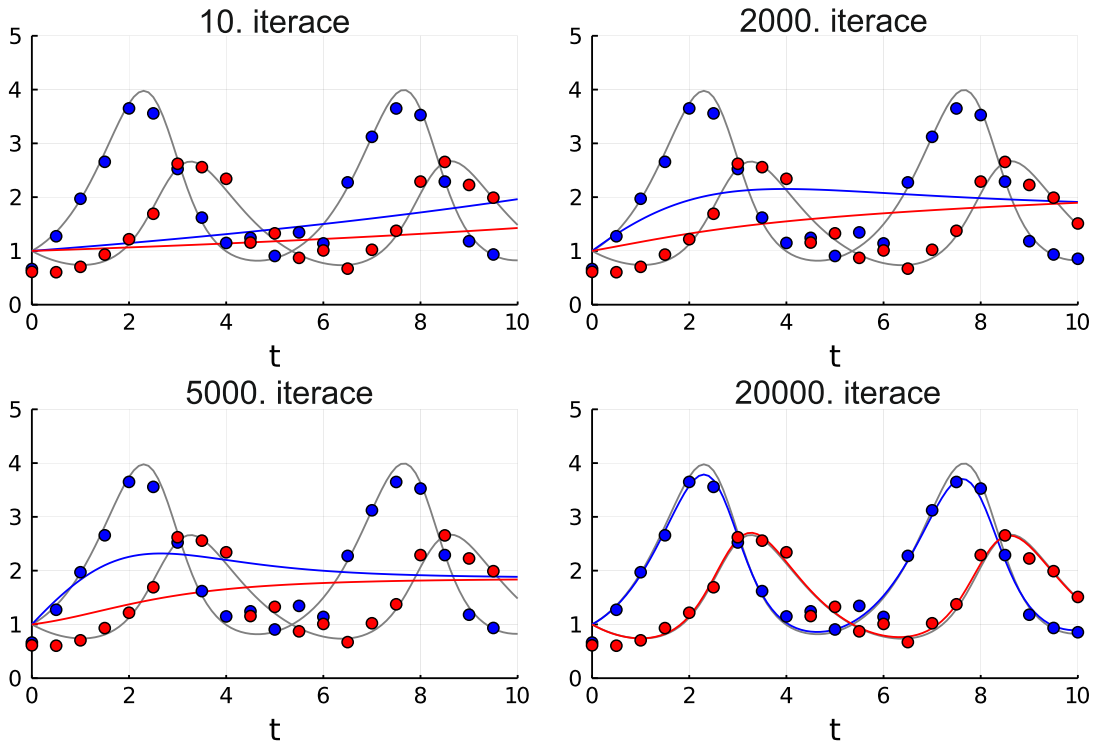
Obrázek 7.1: Výsledný stav váhové matice W jednotlivých metod pro Lotkovy-Volterovy diferenciální rovnice. Zelená barva označuje správně určené parametry, červená špatně určené parametry a oranžová parametry blížící se ke správné hodnotě. U stochastických metod (ELBO-N-ODE a ELBO-NiG-ODE) je výsledná hodnota napočítána jako průměr posledních 100 iterací. U zbylých je výsledná hodnota rovna váhové matici v poslední iteraci.



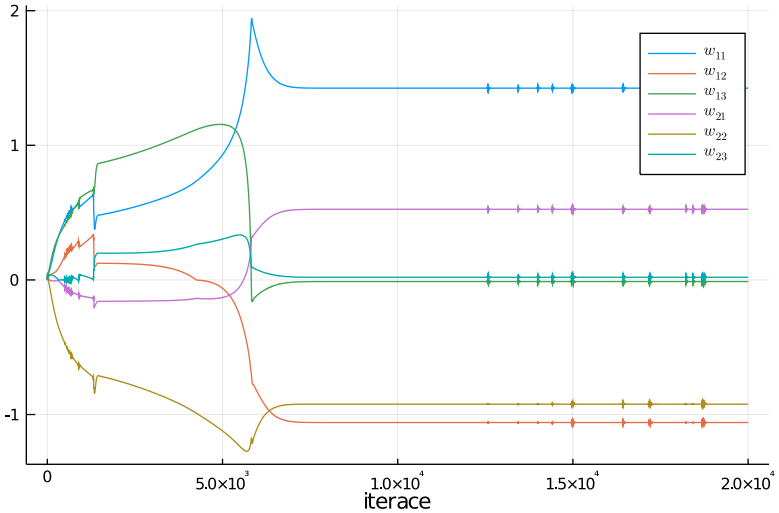
Obrázek 7.2: Průběh učení soustavy Lotkových-Volterových diferenciálních rovnic popsaných v 7.2 za použití metody **LS-ODE** 6.1 a Rungova-Kuttova řešiče 5. řádu. Šedá křivka znázorňuje správné řešení. Barevné body značí experimentální data D. Barevné křivky pak značí řešení s váhovou maticí W v k-té iteraci.



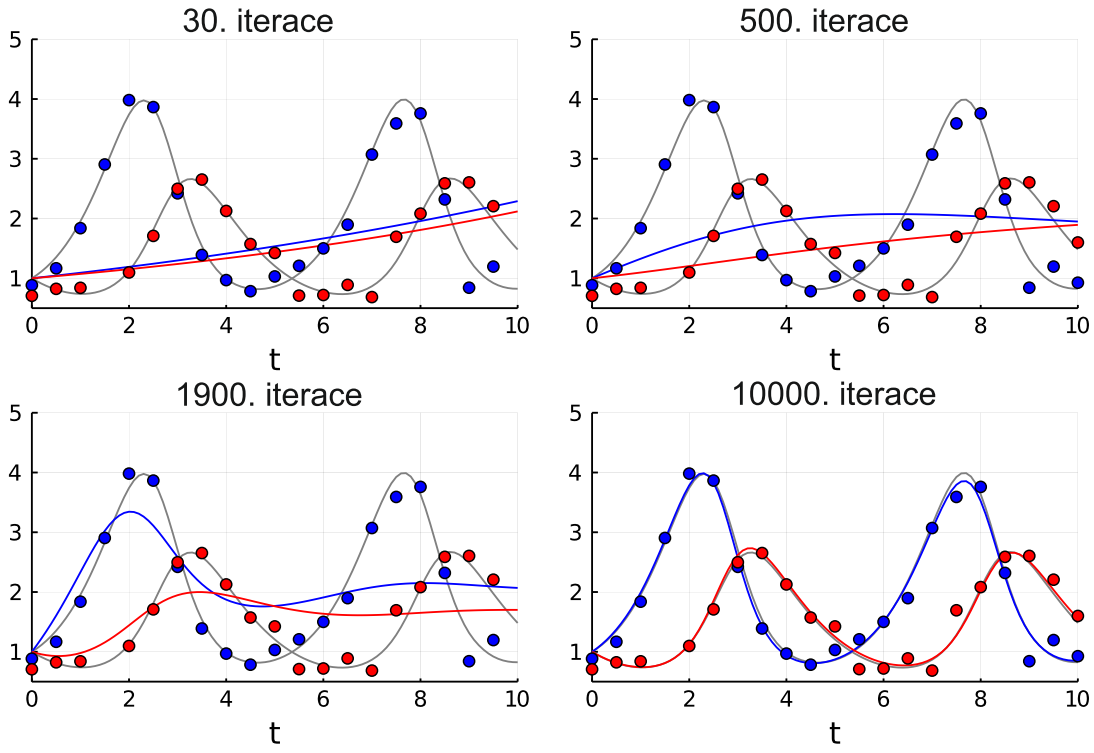
Obrázek 7.3: Průběh učení jednotlivých vah w_{ij} pro Lotkovu-Volterovu diferenciální rovnici popsanou v 7.2 za použití metody **LS-ODE** 6.1 a Rungova-Kuttova řešiče 5. řádu.



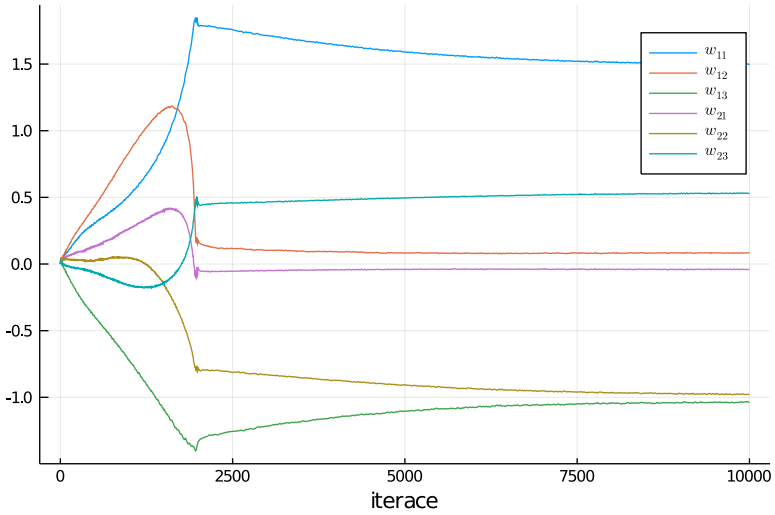
Obrázek 7.4: Průběh učení soustavy Lotkových-Volterových diferenciálních rovnic popsaných v 7.2 za použití metody **Lasso-ODE** 6.2 a Rungova-Kuttova řešiče 5. řádu. Šedá křivka znázorňuje správné řešení. Barevné body značí experimentální data D. Barevné křivky pak značí řešení s váhovou maticí W v k-té iteraci.



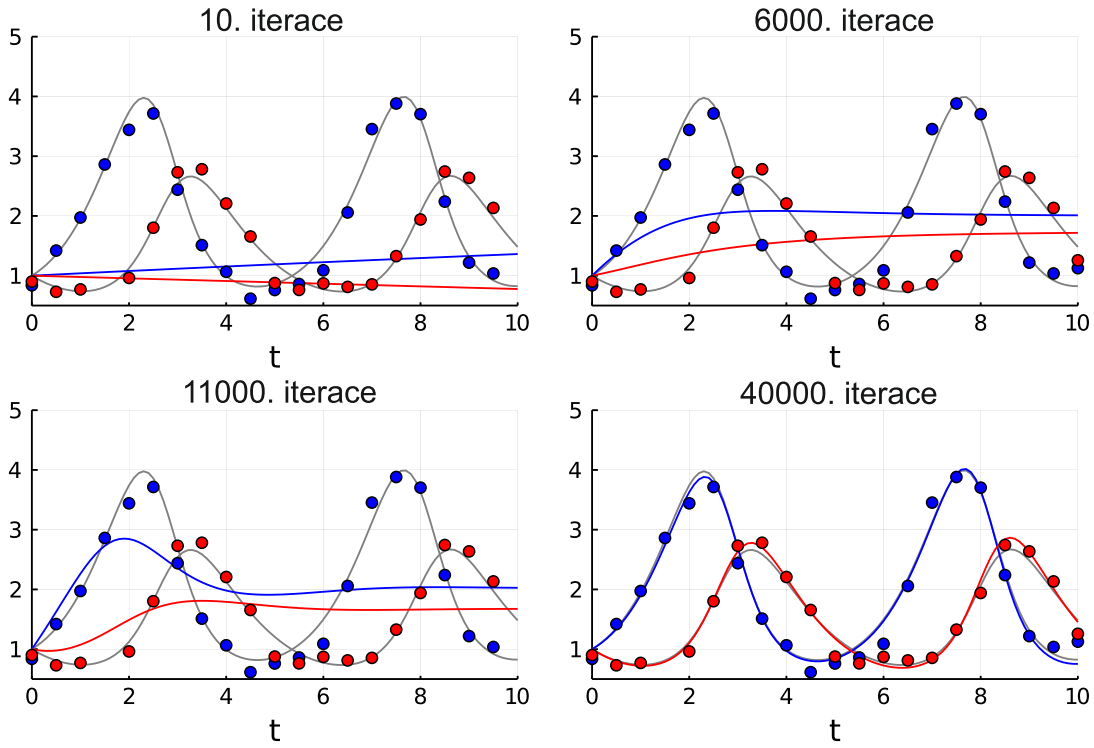
Obrázek 7.5: Průběh učení jednotlivých vah w_{ij} pro Lotkovu-Volterovu diferenciální rovnici popsanou v 7.2 za použití metody **Lasso-ODE** 6.2 a Rungova-Kuttova řešiče 5. řádu.



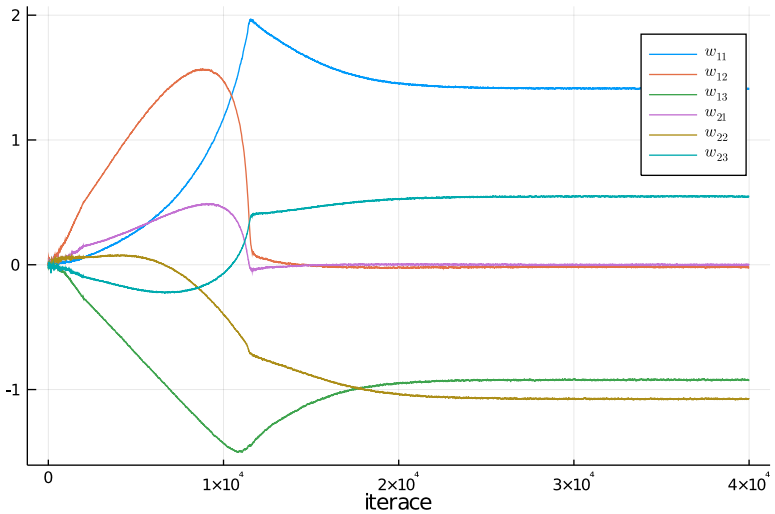
Obrázek 7.6: Průběh učení soustavy Lotkových-Volterových diferenciálních rovnic popsaných v 7.2 za použití metody **ELBO-N-ODE** 6.3 a Rungova-Kuttova řešiče 5. řádu. Šedá křivka znázorňuje správné řešení. Barevné body značí experimentální data D. Barevné křivky pak značí řešení s váhovou maticí W v k-té iteraci.



Obrázek 7.7: Průběh učení jednotlivých vah w_{ij} pro Lotkovu-Volterovu diferenciální rovnici popsanou v 7.2 za použití metody **ELBO-N-ODE** 6.3 a Rungova-Kuttova řešiče 5. řádu.



Obrázek 7.8: Průběh učení Lotkových-Volterových diferenciálních rovnic popsaných v 7.2 za použití metody **ELBO-NiG-ODE** 6.4 a Rungova-Kuttova řešiče 5. řádu. Šedá křivka znázorňuje správné řešení. Barevné body značí experimentální data D. Barevné křivky pak značí řešení s váhovou maticí W v k-té iteraci.



Obrázek 7.9: Průběh učení jednotlivých vah w_{ij} pro Lotkovu-Volterovu diferenciální rovnici popsanou v 7.2 za použití metody **ELBO-NiG-ODE** 6.4 a Rungova-Kuttova řešiče 5. řádu.

7.2 Harmonický oscilátor ODE

Nyní se pokusíme metody aplikovat na fyzikální systém harmonického oscilátoru. Ten je popsán diferenciální rovnicí

$$F = m\frac{d^2x}{dt^2} = -kx - c\frac{dx}{dt}$$

s počátečními podmínkami $\frac{dx}{dt}(0) = a, x(0) = b$. Tato rovnice se dá použitím substituce $\frac{dx}{dt} = y_2$ a $x = y_1$ převést na soustavu dvou diferenciálních rovnic prvního řádu

$$\begin{aligned} y_1' &= y_2 \\ y_2' &= -\frac{k}{m}y_1 - \frac{c}{m}y_2 \end{aligned}$$

s počátečními podmínkami $y_2(0) = a, y_1(0) = b$. Do soustavy přidáme jednu přebytečnou bazickou funkci, která nám úlohu mírně zkomplikuje. Můžeme se přesvědčit, že soustava diferenciálních rovnic opět splňuje požadovaný tvar

$$\frac{\partial\xi(t)}{\partial t} = \begin{bmatrix} y_1'(t) \\ y_2'(t) \end{bmatrix}, \qquad \widehat{W} = \begin{bmatrix} \widehat{w}_{11} & \widehat{w}_{12} & \widehat{w}_{13} \\ \widehat{w}_{21} & \widehat{w}_{22} & \widehat{w}_{23} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ -k/m & -c/m & 0 \end{bmatrix}, \qquad f(t, \xi(t)) = \begin{bmatrix} y_1(t) \\ y_2(t) \\ y_1(t)y_2(t) \end{bmatrix}.$$

Úkolem bude opět nalézt hodnoty členů váhové matice tak, aby se co možná nejvíc blížily skutečným hodnotám $\widehat{W}$. Tu si pro tuto úlohu zvolíme následovně:

$$\widehat{W} = \begin{bmatrix} 0.0 & 1.0 & 0.0 \\ -1.5 & 0.0 & 0.0 \end{bmatrix}$$

Rovnici pak budeme řešit na intervalu $[0, 10]$ s počátečními podmínkami $[y_1(0), y_2(0)] = [1, 1]$. V bodech $t_i = ih$ pro $i \in \{0, ..., 20\}$ a $h = 0.5$. Ke zkoumaným bodům pak přidáme náhodnou chybu $e_i \sim N(0, \sigma^2)$, kde $\sigma = 0.1$. Počáteční stav aproximační matice byl pro všechny metody volen

$$W = \begin{bmatrix} -0.1 & -0.1 & -0.1 \\ -0.1 & -0.1 & -0.1 \end{bmatrix}.$$

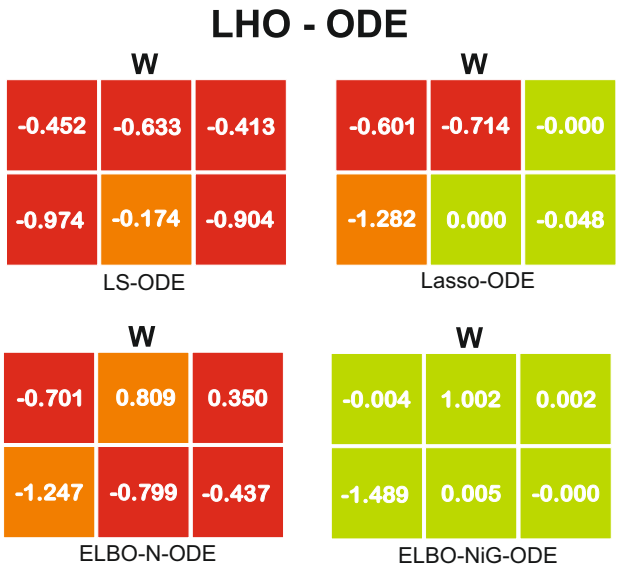
Pro jednotlivé metody dostaneme následující výsledky.

Harmonický oscilátor ODE			
Použitá metoda	# iterací	R^2	# nenulových w_i
LS-ODE	10 000	0.13207	6
Lasso-ODE	>100 000	0.09619	3
ELBO-N-ODE	>170 000	0.18490	6
ELBO-NiG-ODE	30 000	0.97942	2

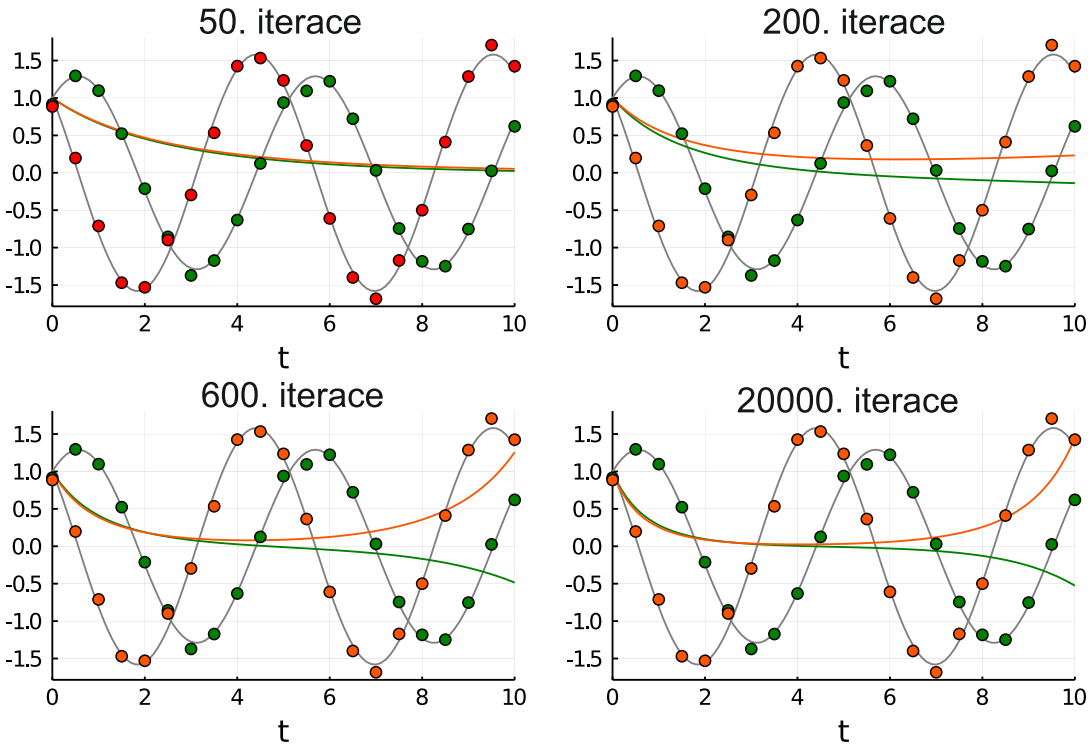
Metody založené na maximálně věrohodném odhadu (**LS-ODE** a **Lasso-ODE**) se pro tuto úlohu ukázaly jako nevhodné. Optimalizační algoritmus ADAM nebyl schopen nalézt globální minimum ztrátových funkcí pro žádnou z voleb kroku $k \in \{10^{-2}, 10^{-3}, 10^{-4}\}$. Výsledky zde publikované jsou ty, kterých bylo dosaženo při volbě $k = 10^{-3}$. Vzhledem k R^2 statistice se jednalo o ty nejlepší pro obě metody. Pro metodu Lasso-ODE byl zvolen parametr $\lambda = 1$.

Metoda **ELBO-N-ODE** se při řešení této úlohy také ukázala jako nevhodná. Výsledky zde publikované jsou získané při volbě výchozího stavu $\sigma_i = 0.05$ a $\tau_i = 0.15$ pro $\forall i = \{1, \ldots, 6\}$. Nejlepších výsledků vzhledem k R^2 statistice bylo dosaženo při volbě kroku $k = 10^{-4}$. V průběhu řešení úlohy touto metodou, bylo častým jevem, že pro některé vzorky $\epsilon \sim N(0, 1)$ řešení diferenciální rovnice na zvoleném intervalu divergovalo. V důsledku pak řešič diferenciální rovnice nebyl schopen napočítat vzorky na celém intervalu. V takových případech byl vygenerovaný nový vzorek a starý zapomenut.

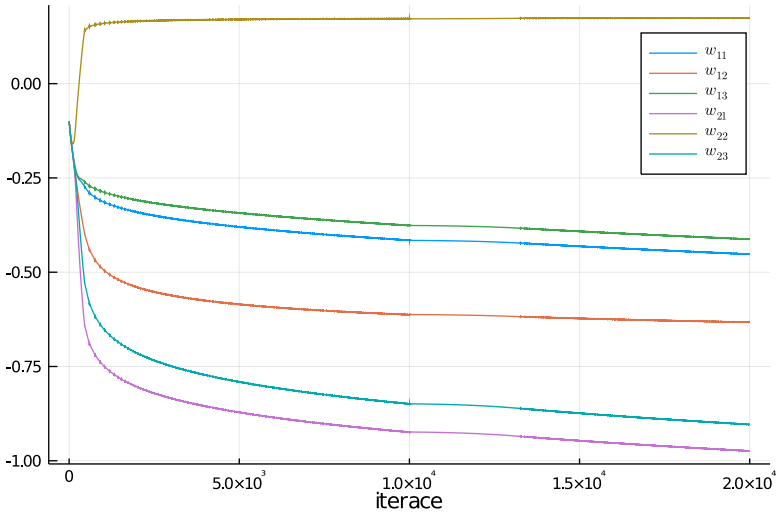
Jediná metoda, která dokázala tuto úlohu zdárně vyřešit je **ELBO-NiG-ODE**. Všechny členy váhové matice byly určeny správně s poměrně velkou přesností. Tomu zřejmě napomohl fakt, že generované body měly menší rozptyl než u předchozí úlohy. Výsledky zde publikované jsou získané při volbě $a_i = 10$, $b_i = 20$ a $\sigma_i = 0.14$ pro $\forall i = \{1, \ldots, 6\}$ a kroku $k = 10^{-3}$.



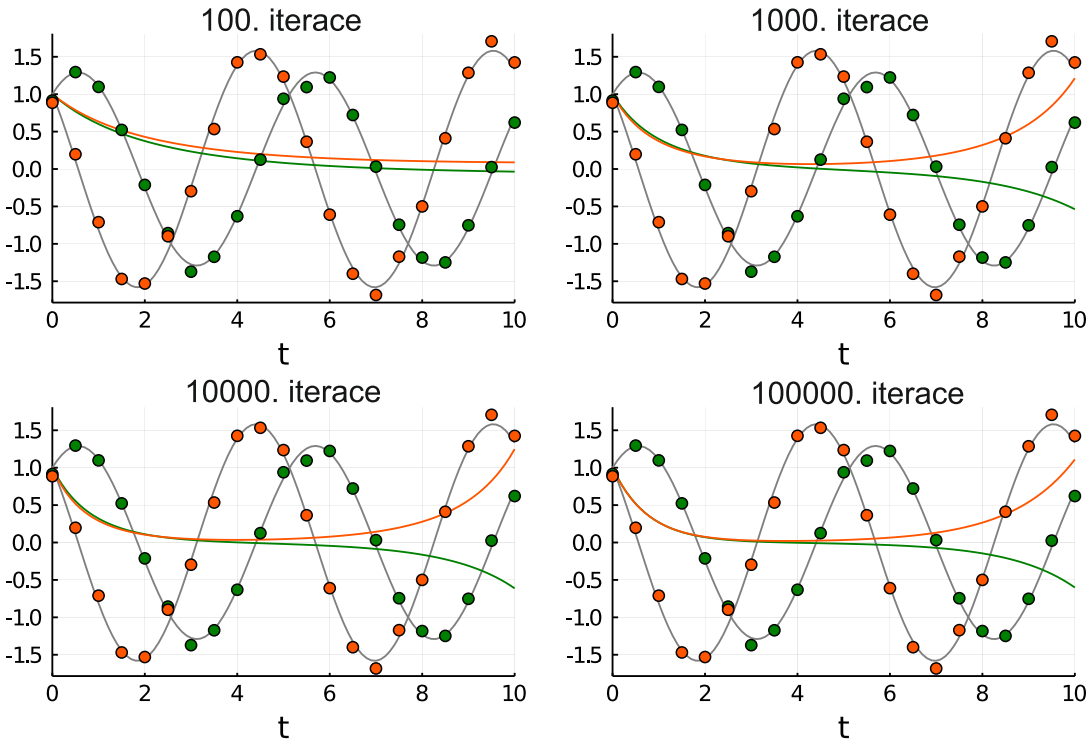
Obrázek 7.10: Výsledný stav váhové matice W jednotlivých metod pro úlohu harmonického oscilátoru. Zelená barva označuje správně určené parametry, červená špatně určené parametry a oranžová parametry blížící se ke správné hodnotě. U stochastických metod (ELBO-N-ODE a ELBO-NiG-ODE) je výsledná hodnota napočítána jako průměr posledních 100 iterací. U zbylých je výsledná hodnota rovna stavu váhové matice v poslední iteraci.



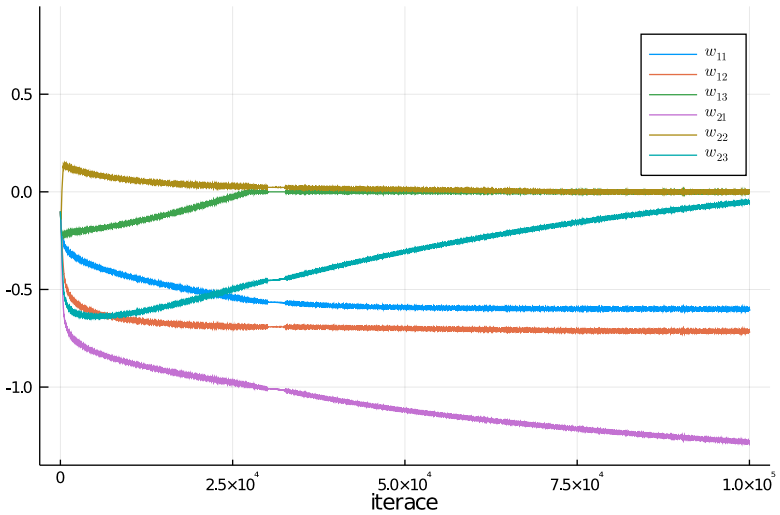
Obrázek 7.11: Průběh učení soustavy diferenciálních rovnic pro harmonický oscilátor popsaný v 7.2 za použití metody **LS-ODE** 6.1 a Rungova-Kuttova řešiče 5. řádu. Šedá křivka znázorňuje správné řešení. Barevné body značí experimentální data D. Barevné křivky pak značí řešení s váhovou maticí W v k-té iteraci.



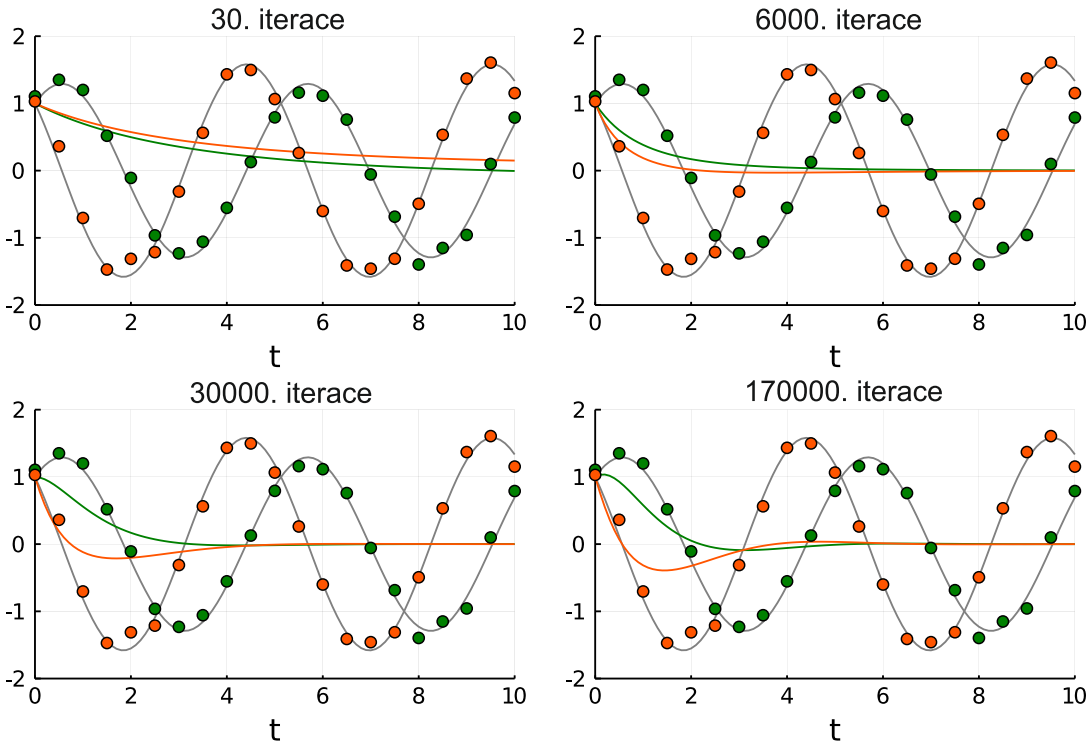
Obrázek 7.12: Průběh učení jednotlivých vah w_{ij} pro Harmonický oscilátor popsaný v 7.2 při použité metodě **LS-ODE** 6.1 a Rungova-Kuttova řešiče 5. řádu.



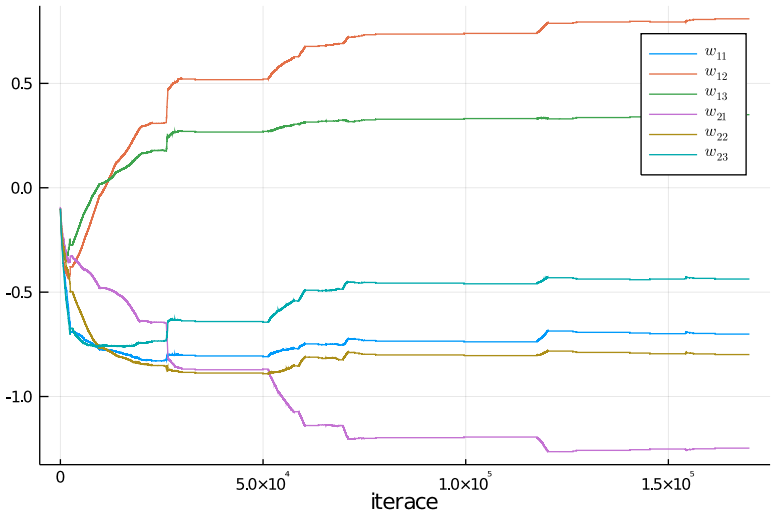
Obrázek 7.13: Průběh učení soustavy diferenciálních rovnic pro harmonický oscilátor popsaný v 7.2 za použití metody **Lasso-ODE** 6.2 a Rungova-Kuttova řešiče 5. řádu. Šedá křivka znázorňuje správné řešení. Barevné body značí experimentální data D. Barevné křivky pak značí řešení s váhovou maticí W v k-té iteraci.



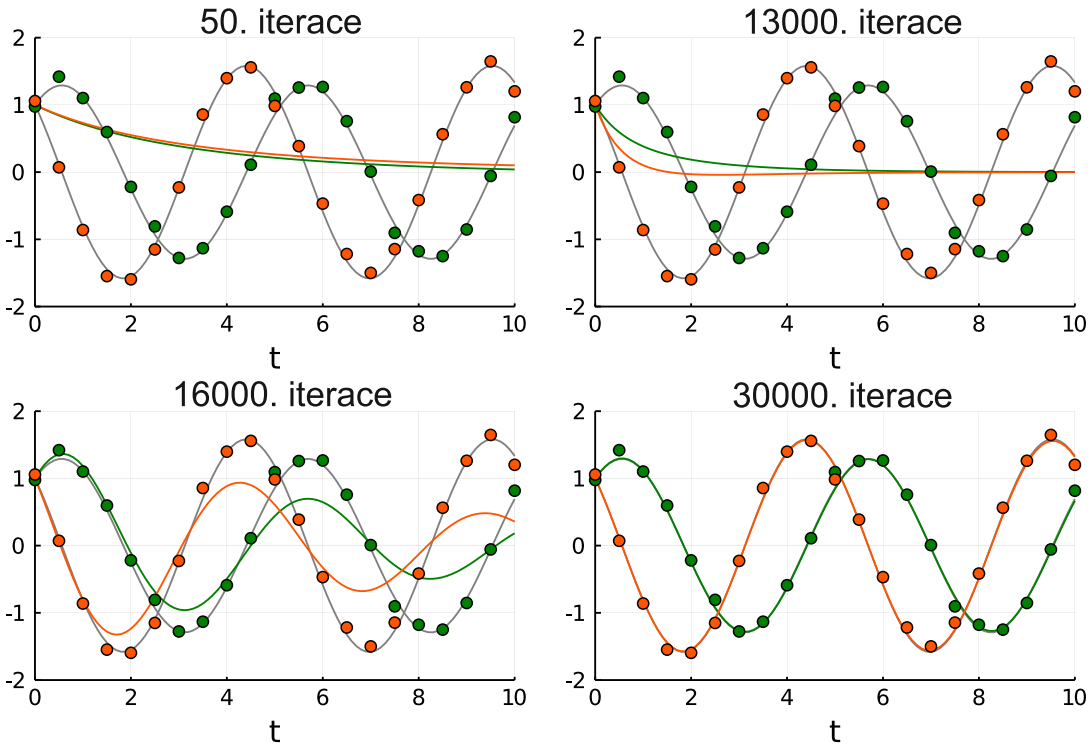
Obrázek 7.14: Průběh učení jednotlivých vah w_{ij} pro Harmonický oscilátor popsaný v 7.2 při použité metodě **Lasso-ODE** 6.2 a Rungova-Kuttova řešiče 5. řádu.



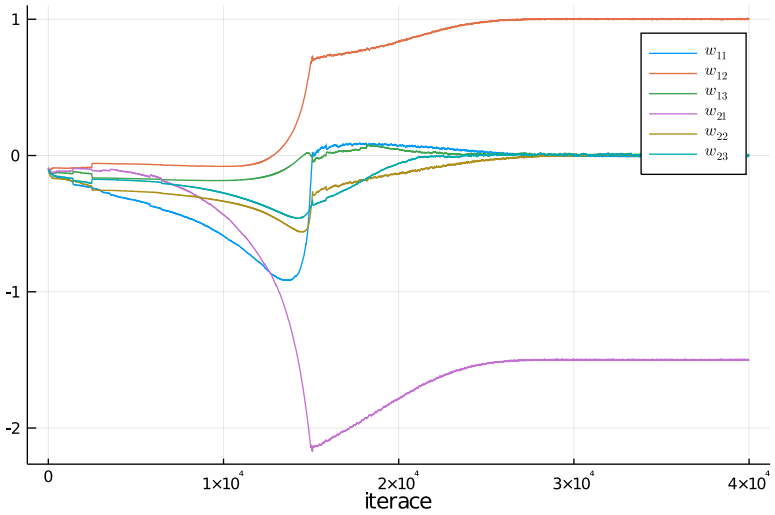
Obrázek 7.15: Průběh učení soustavy diferenciálních rovnic pro harmonický oscilátor popsaný v 7.2 za použití metody **ELBO-N-ODE** 6.3 a Rungova-Kuttova řešiče 5. řádu. Šedá křivka znázorňuje správné řešení. Barevné body značí experimentální data D. Barevné křivky pak značí řešení s váhovou maticí W v k-té iteraci.



Obrázek 7.16: Průběh učení jednotlivých vah w_{ij} pro Harmonický oscilátor popsaný v 7.2 při použité metodě **ELBO-N-ODE** 6.3 a Rungova-Kuttova řešiče 5. řádu.



Obrázek 7.17: Průběh učení soustavy diferenciálních rovnic pro harmonický oscilátor popsaný v 7.2 za použití metody **ELBO-NiG-ODE** 6.4 a Rungova-Kuttova řešiče 5. řádu. Šedá křivka znázorňuje správné řešení. Barevné body značí experimentální data D. Barevné křivky pak značí řešení s váhovou maticí W v k-té iteraci.



Obrázek 7.18: Průběh učení jednotlivých vah w_{ij} pro Harmonický oscilátor popsaný v 7.2 při použité metodě **ELBO-NiG-ODE** 6.4 a Rungova-Kuttova řešiče 5. řádu.

Závěr

Primárním úkolem této práce bylo sestavit funkční program, který dokáže z experimentálních dat zpětně odvodit tvar diferenciální rovnice, za pomoci které byly data generovány. Při řešení této úlohy jsme měli upřednostnit řešení, které dokáže fyzikální systém popsat co možná nejjednodušším způsobem.

Proto, abychom byli takovou úlohu schopni řešit, jsme si nejprve potřebovali zavést numerické metody pro řešení diferenciálních rovnic. U těchto metod jsme se omezili na řešení diferenciálních rovnic s počátečními podmínkami. Z toho důvodu jsou i finální metody pro určení struktury diferenciálních rovnic schopny řešit pouze tento typ úlohy.

Další nezbytnou částí pro řešení této úlohy jsou optimalizační algoritmy. Jedním z námi představených optimalizačních algoritmů byl algoritmus ADAM. Ten je jedním z nejpoužívanějších optimalizačních algoritmů v oblasti strojového učení.

Celkově jsme popsali čtyři různé metody pro hledání struktury diferenciálních rovnic. Dvě metody byly odvozeny za pomoci maximálně věrohodných odhadů. Dvě za pomoci teorie aproximačních Bayesových metod. Všechny tyto metody byly určeny především výsledným tvarem své ztrátové funkce. Ta se za pomoci optimalizačního algoritmu musela minimalizovat, což nás při nalezení globálního minima vedlo na správný tvar diferenciálních rovnic. Při jejich porovnávání na konkrétních úlohách v kapitole 7 se však ukázalo, že dosažení globálního minima ztrátové funkce může být pro některé metody problematické. Nejlepších výsledků pro obě řešené úlohy dosáhla metoda využívající pro aproximaci normální rozdělení a inverzní gamma rozdělení.

Z časových důvodů jsem bohužel už nebyl schopen dané metody aplikovat na reálná data. Proto všechny zde uvedené příklady obsahují pouze uměle generovaná data. Tomuto nedostatku bych se rád věnoval ve svém navazujícím studiu ve výzkumném úkolu.

Literatura

[1] ANDĚL, Jiří; Základy matematické statistiky, 2., opr. vyd. Praha: Matfyzpress, 2007.

[2] BISHOP, Christopher M.; Pattern Recognition and Machine Learning, Springer New York, 2011.

[3] HEIM, Niklas and ŠMÍDL, Václav and PEVNÝ, Tomáš; Rodent: Relevance determination in differential equations, 2020, arXiv:1912.00656v2.

[4] KINGMA, Diederik P. and BA, Jimmy; Adam: A Method for Stochastic Optimization, Proceedings of International Conference on Learning Representations, 2015, arXiv:1412.6980.

[5] KINGMA, Diederik P. and WELLING, Max; Auto-Encoding Variational Bayes, 2013, arXiv:1312.6114.

[6] KRBÁLEK, Milan; Teorie míry a Lebesgueova integrálu, 1. dotisk 1. vydání. ČVUT, 2018.

[7] KUCHIBHOTLA, Arun K. and BROWN, Lawrence D. and BUJA, Andreas and CAI, Junhui; All of Linear Regression, 2019, arXiv:1910.06386v1.

[8] MACKAY, David J. C.; Bayesian Non-linear Modelling for the Prediction Competition, pp. 14, 1994.