České vysoké učení technické v Praze
Fakulta jaderná a fyzikálně inženýrská

Katedra matematiky
Obor: Matematické inženýrství

Vliv ztrátové funkce na detekci anomálií

The Effect of the Loss Function on Quality of Anomaly Detection

DIPLOMOVÁ PRÁCE

Vypracoval: Bc. Viktor Bílek
Vedoucí práce: Doc. Ing. Tomáš Pevný, Ph.D.
Rok: 2021

České vysoké učení technické v Praze, Fakulta jaderná a fyzikálně inženýrská

Katedra: matematiky Akademický rok: 2020/2021

ZADÁNÍ DIPLOMOVÉ PRÁCE

Student:	Bc. Viktor Bílek
Studijní program:	Aplikace přírodních věd
Studijní obor:	Matematické inženýrství
Název práce (česky):	Vliv ztrátové funkce na detekci anomálií
Název práce (anglicky):	The Effect of the Loss Function on Quality of Anomaly Detection

Pokyny pro vypracování:

1) Nastudujte současnou literaturu o metodách maximální věrohodnosti při detekci anomálií, o ztrátové funkci kalibrovaných anomálií a ztrátovou funkci OC-SPN.
2) Vypište podmínky na rozhodovací funkce vyžadované jednotlivými ztrátovými funkcemi.
3) Naimplementujte jednotlivé ztrátové funkce.
4) Naimplementujte rozhodovací funkce použitelné se ztrátovými funkcemi.
5) Porovnejte jednotlivé ztrátové funkce za použití evaluačního nástroje vyvíjeného na katedře počítačů fakulty elektrotechnické.

Doporučená literatura:

1) B. Schölkopf, J. Platt, J. Shawe-Taylor, A. Smola, R. Williamson, Estimating the support of a high-dimensional distribution. Neural computation 13(7), 2001, 1443-1471.

2) A. Menon, R. Williamson, A loss framework for calibrated anomaly detection. Proceedings of the 32nd International Conference on Neural Information Processing Systems, 2018, 1494–1504.

3) J. Wang, S. Sun, Y. Yu, Multivariate Triangular Quantile Maps for Novelty Detection. Advances in Neural Information Processing Systems 32, 2019, 5060-5071.

4) T. Pevny, V. Smidl, M. Trapp, O. Polacek, T. Oberhuber, Sum-Product-Transform Networks: Exploiting Symmetries using Invertible Transformations. arXiv preprint arXiv:2005.01297, 2020.

5) W. Grathwohl, R. T. Chen, J. Betterncourt, I. Sutskever, D. Duvenaud, Ffjord: Free-form continuous dynamics for scalable reversible generative models. arXiv preprint arXiv:1810.01367, 2019.

6) I. Steinwart, D. Hush, C. Scovel, A Classification Framework for Anomaly Detection. Journal of Machine Learning Research 6(1), 2005, 211-232.

Jméno a pracoviště vedoucího diplomové práce:

Doc. Ing. Tomáš Pevný, Ph.D.
Katedra počítačů, Fakulta elektrotechnická, České vysoké učení technické v Praze, Karlovo náměstí 13, 121 35 Praha 2

Jméno a pracoviště konzultanta:

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

Datum zadání diplomové práce: 31.10.2020

Datum odevzdání diplomové práce: 3.5.2021

Doba platnosti zadání je dva roky od data zadání.

V Praze dne 27. října 2020

.....
garant oboru

.....
vedoucí katedry

.....
děkan

Prohlášení

Prohlašuji, že jsem svoji diplomovou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v přiloženém seznamu.

V Praze dne .....

.....
Bc. Viktor Bílek

Poděkování

Chtěl bych poděkovat svému školiteli Doc. Ing. Tomáši Pevnému, Ph.D. za ochotu, vstřícnost a precizní vedení mé diplomové práce i v krizové době s omezenou komunikací.

Bc. Viktor Bílek

Název práce:
Vliv ztrátové funkce na detekci anomálií

Autor: Bc. Viktor Bílek

Studijní program: Aplikace přírodních věd
Obor: Matematické inženýrství
Druh práce: Diplomová práce

Vedoucí práce: Doc. Ing. Tomáš Pevný, Ph.D.
Katedra počítačů, Fakulta elektrotechnická,
České vysoké učení technické v Praze
Konzultant: Ing. Tomáš Oberhuber, Ph.D.
Katedra matematiky, Fakulta jaderná a fyzikálně inženýrská,
České vysoké učení technické v Praze

Abstrakt: Detekce anomálií zaznamenává uplatnění v mnohých oborech moderní datové analýzy. Diplomová práce pojednává o několika metodách strojového učení právě pro tuto detekci. V práci jsou podrobně rozebrány modely odhadující hustotu pravděpodobnosti - konkrétně modely využívající transformaci náhodné veličiny a modely s grafovou reprezentací. Poté následuje popis jejich využití při klasifikaci anomálií. Dále je představena modifikace samotného procesu učení jednotlivých metod pomocí úpravy tzv. ztrátové funkce. Zda-li tato úprava přinesla pozitivní výsledky je vyhodnoceno v rozsáhlé výpočetní studii, kde jsme modifikaci provedli pro několik představitelů zmíněných modelů odhadujících hustotu pravděpodobnosti.
Klíčová slova: Detekce anomálií, strojové učení, hustota pravděpodobnosti.

Title:
The Effect of the Loss Function on Quality of Anomaly Detection

Author: Bc. Viktor Bílek

Abstract: Anomaly detection finds use in many fields of the modern data analysis. Master thesis deals with several machine learning methods in this regard. In the first place, we summarize several methods for probability density estimation in detail - specifically models using the transformation of a random variable and the probabilistic graphical models. Afterwards, we describe their use in anomaly classification. Furthermore, we present the modification of the learning process of each method by modifying the so-called loss function. Whether this adjustment brought positive results is evaluated in an extensive computational study, where we performed the modification for several representatives of the mentioned models for probability density estimation.
Key words: Anomaly detection, machine learning, probability density.

Obsah

Úvod

Detekce anomálií skýtá využití v mnohých odvětvích, od analýzy medicínských dat přes vyhledávání transakcí z ukradených bankovních účtů po kontrolu funkčnosti motorů letadel. V této práci se budeme zabývat detekcí anomálií pomocí metod strojového učení. Představené modely umělé inteligence se naučí normální chování poskytnutých dat. Budoucí data, která se budou od tohoto normálního chování vzdalovat, poté model vyhodnotí jako anomální. Výstupem diplomové práce bude studie našeho návrhu změny samotného procesu učení jednotlivých modelů, a to pomocí modifikace tzv. ztrátové funkce.

V první kapitole se seznámíme s matematickou definicí pojmu anomálie a pro budoucí účely nastíníme princip detekce anomálií pomocí modelů odhadujících hustotu pravděpodobnosti. Dále si zmíníme tzv. klasifikátory jedné třídy a rekonstrukční modely, které se principiálně od modelů odhadujících hustotu pravděpodobnosti liší. Jejich stručný popis nám zároveň umožní chápat pojem anomálie z lehce jiného úhlu pohledu.

V následující kapitole budeme pokračovat podrobnějším popisem metod odhadujících hustotu pravděpodobnosti. V první části této kapitoly se budeme zabývat modely, které k odhadu hustoty využívají transformaci náhodné veličiny. Konkrétně si podrobně popíšeme spojité normalizační modely a metodu FFJORD. Ty k transformaci náhodné veličiny využívají tzv. diferenciální neuronové sítě. Dále je stručně shrnut princip metod MAF a RealNVP. V druhé části se seznámíme s modelem Součtových-produktových sítí a Součtových-produktových transformačních sítí. Tyto sítě spadají to třídy pravděpodobnostních modelů s grafovou reprezentací, kam také patří např. známe Bayesovské sítě.

V další kapitole se budeme zabývat tím, jak využít popsané modely odhadující hustotu pravděpodobnosti při detekci anomálií. Nejprve si shrnutím teoretických poznatků objasníme ideu toho, že detekci anomálií můžeme chápat jako klasifikační problém. Dále si představíme druhy anomálií vyskytující se v poskytnutých datových souborech a základní pojmy pro evaluaci jednotlivých metod. Zavedené pojmy nám umožní ohodnotit, jak byla metoda při detekci anomálií úspěšná, resp. nám dovolí jednotlivé metody porovnat.

V poslední kapitole teoretické části této práce se budeme zabývat samotnou modifikací ztrátové funkce. Stručně si popíšeme základní teoretické principy klasifikátorů jedné třídy. Pomocí těchto principů okomentujeme naši motivaci pro zmíněnou modifikaci ztrátové funkce. Zároveň zadefinujeme tzv. kvantilový interval a učení na kvantilovém intervalu.

Práci zakončíme výpočetní studií, kde nejprve na ukázkových datech vizuálně předvedeme vliv provedené modifikace. Dále pomocí evaluační knihovny provedeme rozsáhlou studii na reálných datech. Reálná data budou vybrána z různorodých oborů, od medicínských dat po data z chemického rozboru vín. V poslední řadě jednotlivé výsledky rozebereme v diskuzi.

Kapitola 1

Úvod do detekce anomálií pomocí metod strojového učení

Tato kapitola bude sloužit jako první přiblížení čtenáři k tomu, co je samotná detekce anomálií a stručný popis modelů strojového učení, které k této detekci mohou být využity.

1.1 Základní pojmy detekce anomálií

Nejprve si přiblížíme to, jak chápat samotný pojem anomálie. Pokud bychom chtěli uvést obecnou definici pojmu anomálie, můžeme ji vyslovit takto:

Definice 1. Anomálie je pozorování, které se výrazně liší od nějakého konceptu normality.

Tato definice je na první pohled velice obecná. Ovšem tato obecnost nám poskytuje chtěnou flexibilitu tohoto pojmu. Například různé modely strojového učení pro detekci anomálií mohou na tento pojem nahlížet z trochu jiného úhlu. V této práci však budeme převážně na anomálii nahlížet takto:

Definice 2. Nechť $\mathcal{X} \subseteq \mathbb{R}^D$ je prostor se σ-algebrou $\mathcal{A}$ a s absolutně spojitou (vzhledem k Lebesguově míře) pravděpodobnostní mírou $\mathbb{P}^+$, tuto míru nazýváme **koncept normality**. Nechť p_x^+ je její hustota pravděpodobnosti. **Množinu anomálií** poté definujeme jako

$$\mathcal{A} = \{\boldsymbol{x} \in \mathcal{X},\ p_x^+(\boldsymbol{x}) \leq \tau\},\ \tau \geq 0,$$

kde $\tau \geq 0$ je **práh normality**.

Množina anomálií tedy závisí na prahu $\tau \geq 0$. Pokud by např. nosič hustoty p_x^+ pokrýval celý prostor $\mathcal{X}$, pak by pro $\tau = 0$ byla $\mathcal{A}$ prázdnou množinou. Nyní nastává otázka, jak vhodně zvolit práh $\tau \geq 0$.

Abychom na otázku mohli zodpovědět, musíme dodat tzv. **předpoklad koncentrace dat**. Tento předpoklad říká to, že normální data, tedy prvky množiny $\mathcal{X} \setminus \mathcal{A}$, se

budou shlukovat v nějakém malém a omezeném objemu. V rámci Lebesguovy míry μ předpokládáme, že pro $\mathcal{X} \setminus \mathcal{A}$ neprázdnou množinu bude platit, že hodnota

$$\mu\left(\mathcal{X} \setminus \mathcal{A}\right) = \mu\left(\{\boldsymbol{x} \in \mathcal{X},\ p_x^+(\boldsymbol{x}) > \tau\}\right) \tag{1.1}$$

bude relativně malá pro vhodně zvolené τ.

Poznámka. Nechť $\mathcal{X}$ je v definici 2 kompaktní a nechť $\mathbb{P}^+$ odpovídá rovnoměrnému rozdělení. Intuitivně bychom v rovnoměrně rozdělených datech těžko hledali anomálie. Zároveň by pro žádné τ nebyl splněn předpoklad koncentrace míry (buď by množina $\mathcal{X} \setminus \mathcal{A}$ byla prázdná, nebo by $\mu\left(\mathcal{X} \setminus \mathcal{A}\right) = \mu(\mathcal{X})$).

Poznámka. Všimněme si, že v definici množiny anomálií je $\mathbb{P}^+$ absolutně spojitá vůči Lebesguově míře μ. V předpokladu koncentrace dat jsme objem vztahovali také k míře μ. Pokud by $\mathbb{P}^+$ byla absolutně spojitá vůči jiné míře ν, mohli bychom definici 2 a předpoklad (1.1) vyslovit s mírou ν. V této práci budeme však vždy uvažovat Lebesguovu míru.

Nyní předpokládejme, že data $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathcal{X}$ jsou generována pravděpodobnostní distribucí s absolutně spojitou mírou $\mathbb{P}$ na prostoru $\mathcal{X} \subseteq \mathbb{R}^D$ s hustotou pravděpodobnosti p_x. Uvažujme, že tato míra je rovna konceptu normality, tedy $\mathbb{P} \equiv \mathbb{P}^+$ a $p_x \equiv p_x^+$.

Poznámka. V praxi tato rovnost reálně nenastává, jelikož generovaná data podléhají šumu, atd.

Dále předpokládejme, že v takovémto vzorku dat bývá ze zkušenosti až 5% dat považováno za anomálie. Např. banka ví, že měsíčně až 5% provedených transakcí přes internet je z ukradeného účtu. Označme množinu normálních dat vzhledem k prahu $\tau \geq 0$ jako

$$C^\tau = \{\boldsymbol{x} \in \mathcal{X},\ p_x(\boldsymbol{x}) > \tau\}.$$

Z předchozího příkladu je patrné, že je vhodné požadovat, aby $\mathbb{P}(C^\tau) \geq 0,95$. Neboli, že pravděpodobnost toho, že bankovní transakce nebyla provedena z ukradeného účtu, je nejméně 95%. Ovšem z předpokladu koncentrace dat očekáváme, že se normální data budou shlukovat v nějakém malém objemu. Ideální volba C^τ tedy bude

$$\begin{aligned} C_\alpha &= \operatorname*{arginf}_{C^\tau}\{\mu(C^\tau),\ \mathbb{P}(C^\tau) \geq 1 - \alpha\} \\ &= \{\boldsymbol{x} \in \mathcal{X},\ p_x(\boldsymbol{x}) > \tau_\alpha\}, \end{aligned}$$

kde μ je Lebesguova míra (a kde v našem příkladu je α rovno $0,05$). Dále $\tau_\alpha \geq 0$ je voleno tak, aby $C_\alpha \equiv C^{\tau_\alpha}$. Množinu C_α můžeme nazvat **množinu hladiny normality $\boldsymbol{\alpha}$** vzhledem k hustotě p_x. Pokud bychom znali přesný tvar množiny C_α, mohli bychom zavést klasifikátor pro detekci anomálií $c_\alpha : \mathcal{X} \to \{\pm 1\}$ tvaru

$$c_\alpha(\boldsymbol{x}) = \begin{cases} +1 & \text{pro } \boldsymbol{x} \notin C_\alpha, \\ -1 & \text{pro } \boldsymbol{x} \in C_\alpha. \end{cases}$$

Poznámka. Všimněme si, že v tomto případě anomálie detekujeme pomocí čísla +1, tedy pozitivně. Normální detekujeme jako negativní. Tato konvence je v detekci anomálií běžná (anomálie - pozitivní, normální - negativní).

Zavedením množiny C_α jsme teoreticky zodpověděli na to, jak správně určit práh $\tau_\alpha \geq 0$. Prakticky určení prahu τ_α závisí na tvaru samotného modelu pro detekci anomálií. V této práci se budeme převážně zabývat modely, které odhadují hustotu pravděpodobnosti. V případě takovýchto modelů můžeme práh odhadnout empiricky.

Mějme data $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathcal{X}$, kde $\mathcal{X}$ je prostor s absolutně spojitou mírou s hustotou pravděpodobnosti p_x. Nechť náš model odhadne neznámou hustotu p_x hustotou $\hat{p}_x$. Pak práh $\tau_\alpha \geq 0$ můžeme empiricky odhadnout hodnotou

$$\hat{\tau}_\alpha = \sup_\tau \left\{ \tau \geq 0,\ \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{[0,\hat{p}_x(\boldsymbol{x}_i))}(\tau) \geq 1 - \alpha \right\},$$

kde $\mathbb{1}_{(a,b)}$ značí charakteristickou funkci intervalu (a, b). Zmíněným tvarem empirického odhadu prahu normality $\hat{\tau}_\alpha$ se budeme podrobně zabývat v kapitole 4.

Tímto jsme zmínili i první typ metod pro detekci anomálií, tedy pomocí modelů pro odhad hustoty pravděpodobnosti. Těmto modelům však věnujeme vlastní kapitolu. V následujících sekcích této kapitoly si nastíníme princip jiných tříd modelů pro detekci anomálií.

1.2 Klasifikátory jedné třídy

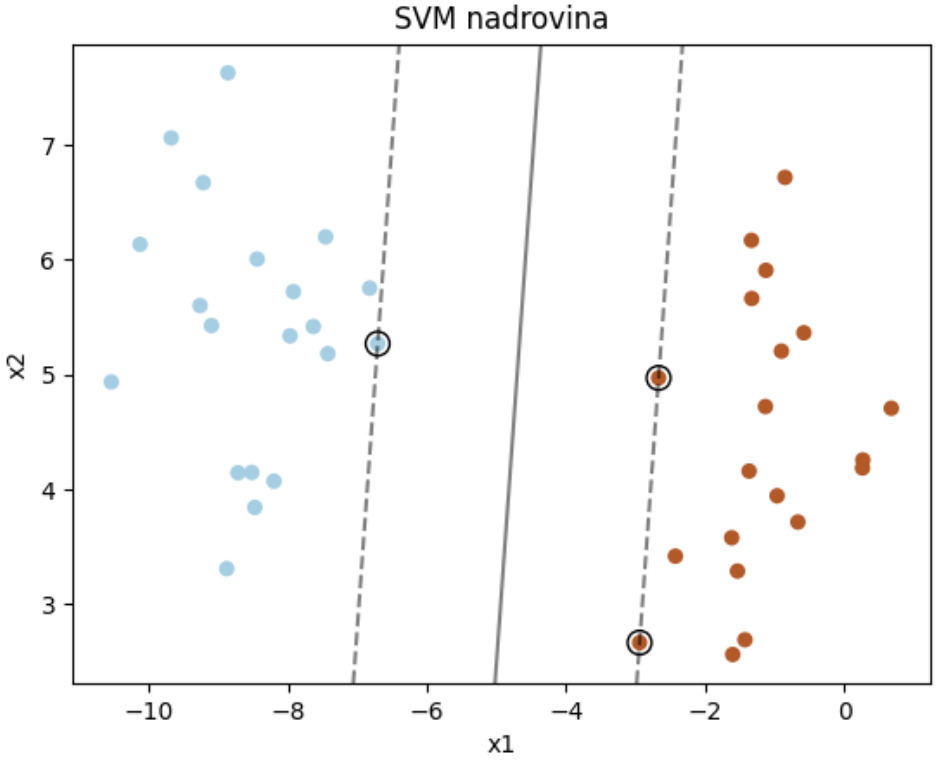
Velkou část výzkumu ve strojovém učení tvoří právě problém klasifikace. Uvažujme problém, kdy chceme určit, zda-li se na fotografii nachází automobil. Cílem je sestrojit funkci takovou, která vrátí číslo +1, pokud se automobil na fotografii nachází, resp. vrátí číslo −1 v opačném případě. Tuto funkci se snažíme nalézt na základě historických dat, tedy předpokládejme, že máme k dispozici soubor dat fotografií $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathcal{X} \subseteq \mathbb{R}^D$. Zároveň mějme ke každé fotografii informaci o tom, zda-li se na fotografii automobil nachází. Tuto zkutečnost vyjádříme pomocí čísla $y \in \{\pm 1\}$, které přiřadíme ke každé fotografii, z čehož utvoříme soubor dat $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n) \in \mathcal{X} \times \{\pm 1\}$. Řekneme, že tento soubor dat náleží dvěma třídám.

Jednou z metod, jak nalézt zmíněnou funkci je tzv. **metoda podpůrných vektorů** (anglicky *Support vector machines*, v textu tuto metodu budeme označovat zkratkou **SVM**). Přesný popis a odvození této metody je nad rámec této práce, proto si popíšeme pouze její základní principy. To nám napomůže k pochopení toho, jak SVM využít pro detekci anomálií.

Základem SVM je lineární separátor. Mějme soubor dat $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n) \in \mathcal{X} \times \{\pm 1\}$, kde $\mathcal{X} \subseteq \mathbb{R}^D$. Předpokládejme, že tato data lze na základě hodnot y_i, $i = 1, \ldots, n$ oddělit nadrovinou danou rovnicí

$$\boldsymbol{w} \cdot \boldsymbol{x} + b = 0, \tag{1.2}$$

kde $\boldsymbol{w} \in \mathbb{R}^D,\ b \in \mathbb{R}$. Neboli řekneme, že $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)$ jsou **lineárně separovatelné**. Pokud existuje jedna taková nadrovina, tak bude pravděpodobně existovat více nadrovin s touto vlastností. Metoda SVM se v tomto případě snaží najít takovou nadrovinu, která bude maximalizovat svojí vzdálenost od vůči ní nejbližšího bodu z první třídy a zároveň vzdálenost od nejbližšího bodu z druhé třídy. Příklad této nadroviny můžeme vidět ve dvourozměrném případě na obrázku 1.1.



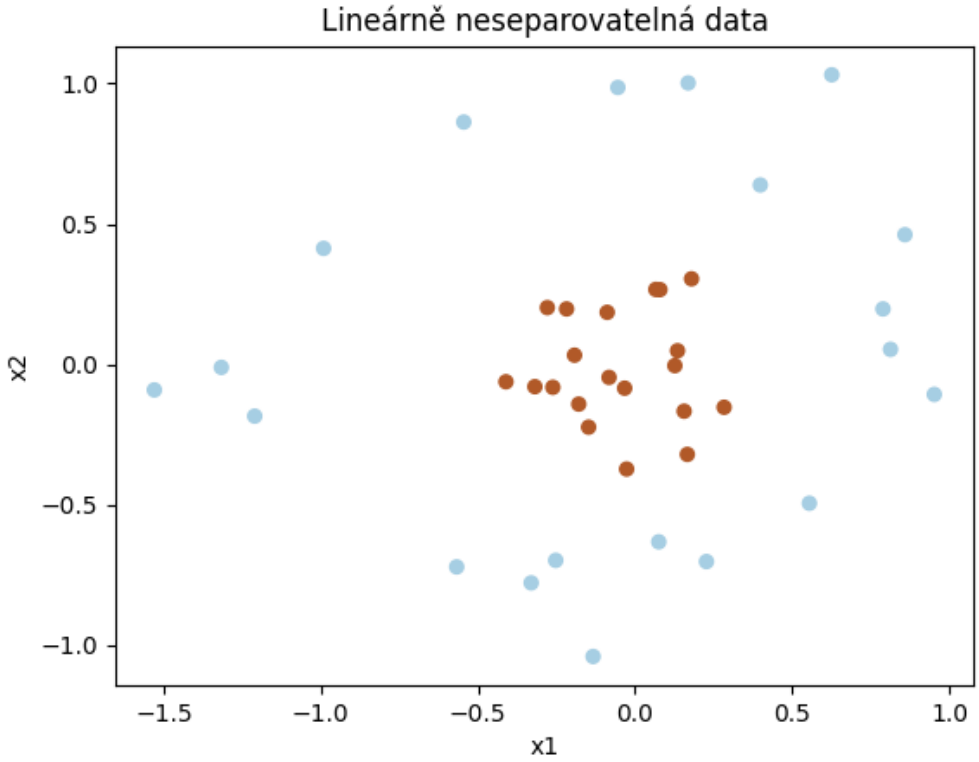
Obrázek 1.1: Příklad nalezené nadroviny separující dvě třídy bodů pomocí metody SVM.

Nejbližší body k nadrovině z každé třídy nazýváme právě **podpůrnými vektory**.

Do této chvíli jsme měli na soubor dat velmi silný předpoklad - data jsou lineárně separovatelná. Co když data lineárně separovatelná nejsou? Příklad takovýchto dat je vyobrazen na obrázku 1.2.

SVM tuto situaci řeší tak, že data nejprve přetransformujeme do Hilbertova prostoru $\boldsymbol{\mathcal{H}}$ se skalárním součinem $(\cdot,\cdot)_{\boldsymbol{\mathcal{H}}}$. Tuto transformaci označíme jako $\Phi : \mathcal{X} \to \boldsymbol{\mathcal{H}}$. Předpokládejme, že existuje zobrazení $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_0^+$ splňující $k(\boldsymbol{x}_1, \boldsymbol{x}_2) = (\Phi(\boldsymbol{x}_1), \Phi(\boldsymbol{x}_2))_{\boldsymbol{\mathcal{H}}}$. Zobrazení k nazýváme **kernel**. Dále předpokládáme, že po této transformaci budou data v novém prostoru $\boldsymbol{\mathcal{H}}$ již lineárně separovatelná. V prostoru $\boldsymbol{\mathcal{H}}$ dále nalezneme lineární separátor s největší vzdáleností od nejbližších bodů z obou tříd.

Poznámka. Dívali bychom se na nalezený lineární separátor v prostoru $\boldsymbol{\mathcal{H}}$ z původního prostoru $\mathcal{X}$ se standardním skalárním součinem, nemusel by se nám separátor



Obrázek 1.2: Příklad dvourozměrným lineárně neseparovatelných dat.

jevit jako rovná nadrovina.

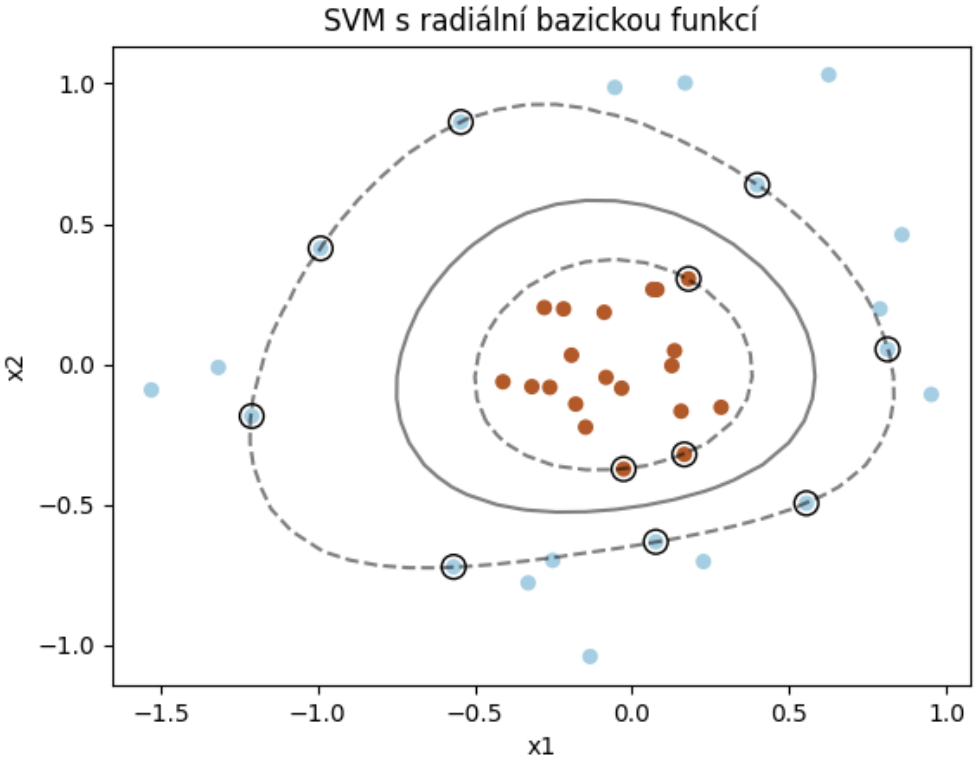
Při hledání tohoto separátoru však nemusíme znát tvar přetransformovaných dat, ale pouze jejich vzájemné hodnoty zvoleného kernelu, tedy hodnoty $k(\boldsymbol{x}_i, \boldsymbol{x}_j)$, kde $i, j = 1, \ldots, n$. To nám umožňuje uvažovat transformaci do Hilbertova prostoru s libovolnou dimenzí, dokonce i nekonečnou! Tvar separátoru nám tedy definuje pouze kernel k a poskytnutá data. Příklad používaného kernelu může být tzv. radiální bázová funkce tvaru

$$k(\boldsymbol{x}_1, \boldsymbol{x}_2) = \exp\left(\frac{-\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2^2}{2\sigma^2}\right),$$

kde $\sigma > 0$. Separátor zkonstruovaný pomocí tohoto kernelu na datech na obrázku 1.2 můžeme vidět na obrázku 1.3.

Pokud jsme nalezli vhodný separátor, můžeme v budoucnu klasifikovat nová data do tříd podle toho, na jaké straně separátoru leží. Nalezení separátoru metodou SVM se zvoleným kernelem je úlohou kvadratického programování.

Nyní nastává otázka, jak tento model použít či modifikovat pro detekci anomálií. Nejčastější situací v detekci anomálií je ta, kdy máme soubor dat $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathcal{X} \subseteq \mathbb{R}^D$. O tomto souboru apriorně víme, resp. se domníváme, že část dat bude anomální. V sekci 1.1 jsme poměr anomálních a normálních dat vyjádřili číslem $\alpha \in (0, 1)$. Máme zde ovšem pouze jednu třídu dat.



Obrázek 1.3: Příklad separace dat pomocí SVM. Použitý kernel byla radiální bázová funkce.

Modifikace SVM pro detekci anomálií byla poprvé publikována v [18]. Idea oddělení normálních dat od anomálních spočívá opět v lineárním separátoru v Hilbertově prostor $\boldsymbol{\mathcal{H}}$ s příslušným kernelem. Od tohoto separátoru požadujeme následující vlastnosti:

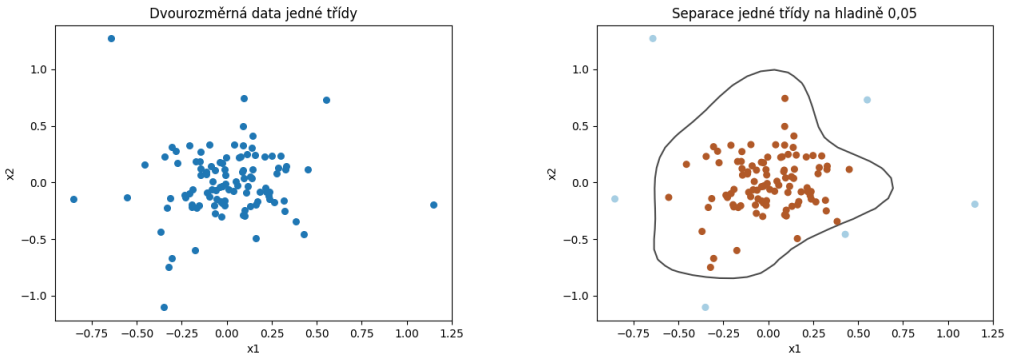
1. Lineární separátor odděluje v prostoru $\boldsymbol{\mathcal{H}}$ $\lceil \alpha \cdot n \rceil$ (funkce $\lceil \cdot \rceil$ zde značí horní celou část) přetransformovaných dat (anomálních) od zbytku (normálních).

2. Vzdálenost lineárního separátoru od nejbližšího normálního bodu a zároveň od nulového vektoru v prostoru $\boldsymbol{\mathcal{H}}$ bude co největší.

Použili jsme tedy SVM pro klasifikaci jedné třídy na hladině $\alpha = 0,05$. Na obrázku 1.4 můžeme vidět nalezení separátoru touto metodou na dvourozměrných datech s $\alpha = 0,05$.

Nalezení tohoto separátoru můžeme opět vyjádřit jako úlohu kvadratického programování.

Tato metoda je zároveň historicky jedním z prvních přístupů strojového učení pro řešení problému detekce anomálií.



(a) Dvourozměrná data

(b) Separace pro $\alpha = 0.05$

Obrázek 1.4: Na obrázku (a) můžeme vidět 106 dvourozměrných dat. Na obrázku (b) se nachází separace jedné třídy na hladině $\alpha = 0.05$ pomocí SVM. Použitý kernel byla radiální bázová funkce.

1.3 Rekonstrukční modely

Popis následujících dvou příkladů modelů vychází se shrnujícího článku pro detekci anomálií [17].

Mějme opět soubor dat $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathcal{X} \subseteq \mathbb{R}^D$. Cílem rekonstrukčních modelů je nalézt zobrazení $\phi_\theta : \mathcal{X} \times \Theta \to \mathcal{X}$, kde Θ značí prostor parametrů a θ hodnotu těchto parametrů. Po tomto zobrazení požadujeme, aby bylo blízké identitě pro data ze souboru $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$, která považujeme za normální. Zobrazení ϕ_θ zavedeme složením dvou jiných zobrazení

1. $\phi_e : \mathcal{X} \to \mathcal{Z}$, které nazýváme *enkodér*,
2. $\phi_d : \mathcal{Z} \to \mathcal{X}$, které nazýváme *dekodér*,

pro která platí $\phi_\theta = (\phi_d \circ \phi_e)_\theta$. Nově vyskytující prostor $\mathcal{Z}$ nazveme latentním prostorem. Typicky pro něj platí, že $\dim(\mathcal{Z}) < \dim(\mathcal{X})$. Pro $\boldsymbol{x} \in \mathcal{X}$, které považujeme za normální, požadujeme, aby $\phi_\theta(\boldsymbol{x}) = \phi_d(\phi_e(\boldsymbol{x})) = \hat{\boldsymbol{x}} \approx \boldsymbol{x}$. Pokud se nám podaří nalézt tato zobrazení s požadovanými vlastnostmi, můžeme zavést klasifikátor anomálií $c_\alpha : \mathcal{X} \to \{\pm 1\}$ tvaru

$$c_\alpha(\boldsymbol{x}) = \begin{cases} +1 & \text{pro} \quad \|\boldsymbol{x} - \phi_d(\phi_e(\boldsymbol{x}))\|^2 \geq \tau_\alpha, \\ -1 & \text{pro} \quad \|\boldsymbol{x} - \phi_d(\phi_e(\boldsymbol{x}))\|^2 < \tau_\alpha, \end{cases}$$

kde τ_α je opět vhodně zvolený práh normality. Vyskytující se normu volíme dle uvážení, typicky euklidovskou.

Abychom zajistili vhodný tvar těchto zobrazení a prostoru $\mathcal{Z}$, zavedeme tzv. **předpoklad koncentrace dat na varietě**. Tento nový předpoklad je intuitivně podobný předpokladu koncentrace dat v sekci 1.1. Nechť $\mathcal{X} \subseteq \mathbb{R}^D$ je prostor s pravděpodobnostní mírou $\mathbb{P}^+$. Nechť v tomto prostoru existuje varieta $\mathcal{M} \subset \mathcal{X}$ s $\dim(\mathcal{M}) <$

$\dim(\mathcal{X})$ splňující

$$\mathbb{P}^+(\mathcal{M}) \geq 1 - \alpha, \tag{1.3}$$

kde $\alpha \in (0, 1)$ je opět předem určená hladina (např. $\alpha = 0.05$). Vhodně zvolené zobrazení ϕ_θ by poté mělo splňovat $\phi_\theta(\mathcal{X}) \approx \mathcal{M} \subset \mathcal{X}$.

Uvedeme si dvě metody, jak vhodně nalézt zmiňovaný enkodér a dekodér. V první variantě využijeme tzv. analýzu hlavních komponent (anglicky *Principal Component Analysis*, dále jen PCA). Mějme soubor $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathcal{X}$. Z tohoto souboru vypočtěme kovarianční matici $\Sigma \in \mathbb{R}^{D\times D}$. Pravděpodobně tato matice nebude diagonální, což implikuje, že jednotlivé příznaky dat budou korelované. Jelikož je matice Σ pozitivně definitní a symetrická, můžeme nalézt ortonormální matici $W \in \mathbb{R}^{D\times D}$, která ji pomocí podobnostní transformace diagonalizuje. Tedy $C = W\Lambda W^T$ bude diagonální. Dále volme W tak, aby čísla na diagonále C byla seřazena od největšího po nejmenší. Provedeme-li dále transformaci nějakého prvku souboru dat $W\boldsymbol{x}_i$, $i = 1, \ldots, n$, dostaneme nový vektor, jehož příznaky korelované nebudou. Zároveň složky vektoru $W\boldsymbol{x}_i$ budou seřazeny podle jejich rozptylu vůči ostatnímu prvkům ze souboru. Tento rozptyl odpovídá příslušnému prvku na diagonále matice C. Pokud bude l posledních prvků na diagonále matice C výrazně menších než ostatní, můžeme posledních l složek ve vektorech $W\boldsymbol{x}_1, \ldots, W\boldsymbol{x}_n$ zanedbat, aniž bychom ztratili mnoho informace o souboru dat. Tato situace odpovídá tomu, že v matici W zanedbáme posledních l řádků, tedy zavedeme $\hat{W} = (W_{ij})_{i=1,\ldots,m}^{j=1,\ldots,D}$, kde $m = D-l-1$.

Z tohoto postupu vyplývá, že $\hat{W}^T\hat{W}\boldsymbol{x} \approx \boldsymbol{x}$ pro většinu prvků souboru dat. Zavedeme tedy enkodér a dekodér jako

$$\phi_e(\boldsymbol{x}) = \hat{W}\boldsymbol{x},$$
$$\phi_d(\boldsymbol{x}) = \hat{W}^T\boldsymbol{x}.$$

Druhá metoda spočívá v použití neuronových sítí (více k neuronovým sítím v [6]). Nechť $\dim(\mathcal{X}) = d$ a volme prostor $\mathcal{Z}$ s $\dim(\mathcal{Z}) = m$, přičemž $m < d$. Za enkodér ϕ_e volme dopřednou neuronovou síť se vstupní dimenzí d a výstupní dimenzí m. Naopak za dekodér volme dopřednou neuronovou síť se vstupní dimenzí m a výstupní dimenzí d. Budeme po těchto sítích požadovat, aby se jejich složení chovalo na varietě $\mathcal{M}$ jako identita. Zároveň z předpokladu koncentrace míry na varietě plyne, že většina dat ze souboru $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathcal{X}$ se bude nacházet na $\mathcal{M}$, nebo v její blízkosti. Pro trénování těchto sítí můžeme zavést ztrátovou funkci tvaru

$$\mathcal{L}(\theta|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = \sum_{i=1}^{n} \|\boldsymbol{x}_i - (\phi_d \circ \phi_e)_\theta(\boldsymbol{x}_i)\|^2 + \mathcal{R}(\phi_d, \phi_e),$$

kde θ značí souhrn parametrů obou sítí a $\mathcal{R}(\phi_d, \phi_e)$ značí zvolený faktor regularizace.

Poznámka. Pokud budou neuronové sítě realizovány jednou lineární vrstvou, pak po optimalizaci bude tato vrstva shodná s maticí metody PCA.

Zmíněné postupy samozřejmě nejsou jedinými příklady detekce anomálií rekonstrukčními modely. Zmíněné dvě metody můžeme např. kombinovat.

Poznámka. Zmíněné postupy nám mohou naznačit, že pojem **anomálie** nemusíme definovat pouze pomocí hustoty pravděpodobnost (viz. definice 2).

Kapitola 2

Modely pro odhad hustoty pravděpodobnosti

Jak již bylo zmíněno v kapitole 1, jednou z možností pro detekci anomálií je pomocí modelů odhadujících hustotu pravděpodobnosti. Těmito modely se budeme ve zbytku práce zabývat, jelikož přirozeně svou podstatou nejlépe vystihují detekování anomálií v souladu s definicí anomálie na počátku kapitoly 1.

Mějme prostor $\mathcal{X} \subseteq \mathbb{R}^D$ s σ-algebrou $\mathcal{A}$ a s absolutně spojitou pravděpodobnostní mírou $\mathbb{P}$ vzhledem k Lebesguovské míře a s hustotou pravděpodobnosti p_x. Mějme data z této distribuce $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$. V následujícím textu budeme značit, že data přísluší prostoru $\mathcal{X}$ se zmíněnou mírou zápisem

$$\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \sim (\mathcal{X}, p_x(\boldsymbol{x})).$$

Následující metody modelují hustotu pravděpodobnosti $p_x^\theta : \mathcal{X} \times \Theta \to \mathbb{R}_0^+$, kde $\Theta \subseteq \mathbb{R}^N$ je prostor parametrů. Navržený model bude robustní, pokud $\exists \theta_0 \in \Theta$ tak, že $p_x^{\theta_0} \approx p_x$, neboli vhodnou sadou parametrů dokáže dostatečně aproximovat hledanou hustotu p_x. Na otázku, jak nalézt vhodné θ_0, nám zodpoví **maximálně věrohodný odhad**. Nejprve vyslovme definice týkající se maximálně věrohodného odhadu.

Definice 3 (Věrohodnostní funkce)**.** Buďte $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathcal{X} \subseteq \mathbb{R}^D$ nezávislá pozorování absolutně spojité náhodné veličiny $\boldsymbol{X}$ s neznámou hustotou pravděpodobnosti. Nechť $\{p_x^\theta : \mathcal{X} \to \mathbb{R}_0^+,\ \theta \in \Theta \subseteq \mathbb{R}^N\}$ je množina hypotetických hustot pravděpodobností. Potom libovolnou funkci tvaru

$$L(\theta|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = c(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) \cdot \prod_{i=1}^{n} p_x^\theta(\boldsymbol{x}_i)$$

nazýváme **věrohodnostní funkcí**, a funkci tvaru

$$l(\theta|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = \ln L(\theta|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$$

nazýváme **logaritmickou věrohodnostní funkcí**, kde c je libovolná funkce (obvykle konstantní).

Poznámka. V této práci budeme vždy uvažovat $c(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = 1$.

Definice 4. Buď $\hat{\theta}_{ML}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ taková borelovsky měřitelná funkce na množině elementárních jevů náhodné veličiny $\boldsymbol{X}$, že platí

$$L(\hat{\theta}_{ML}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = \sup_{\theta \in \Theta} L(\theta|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n).$$

Pokud $\hat{\theta}_{ML}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ závisí na $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ a pokud je určena jednoznačně, potom je nazývána **maximálně věrohodným odhadem** parametru θ.

To že maximálně věrohodný odhad je dobrým odhadem neznámých parametrů θ_0 nám potvrdí následující věta.

Věta 1. Buďte $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \sim (\mathcal{X}, p_x^{\theta_0}(\boldsymbol{x}))$ nezávislá pozorování a nechť $\theta_0 \in \Theta \subseteq \mathbb{R}^N$, nechť $\operatorname{supp} p_x^\theta$ nezávisí na θ a $\mathbb{E}|\ln p_x^\theta| < \infty$ pro všechna $\theta \in \Theta$. Potom pro všechna $\theta \neq \theta_0$ platí

$$\lim_{n\to\infty} P(L(\theta_0|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) > L(\theta|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)) = 1,$$

kde $\theta \neq \theta_0$ je libovolný parametr z Θ.

Věta 1 nám tedy říká, že pokud budeme mít soubor dat $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in (\mathcal{X}, p_x(\boldsymbol{x}))$ nezávislých pozorování a předpoklad, že existuje $\theta_0 \in \Theta$ tak, že $p_x^{\theta_0} \approx p_x$, tak maximalizací funkce $L(\theta|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ (vůči parametrům θ) dosáhneme v rámci navrženého modelu nejlepšího odhadu neznáme hustoty pravděpodobnosti p_x.

V metodách strojového učení jsou často preferovány úlohy minimalizační, kde minimalizovanou funkci nazýváme **ztrátovou funkcí**. Pro modifikaci maximalizační úlohy na úlohu minimalizační stačí výraz přenásobit hodnotou -1. Dále pokud minimalizovanou funkci zlogaritmujeme, tak se minima této funkce nezmění, jelikož logaritmus je ostře rostoucí funkce. Proto v úlohách pro odhad hustoty pravděpodobnosti můžeme zavést ztrátovou funkci tvaru

$$\mathcal{L}(\theta|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = -\sum_{i=1}^{n} \log p_x^\theta(\boldsymbol{x}_i).$$

Aby se hodnota této funkce dramaticky neměnila s velikostí souboru dat, modifikujeme ztrátovou funkci na

$$\mathcal{L}(\theta|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = -\frac{1}{n}\sum_{i=1}^{n} \log p_x^\theta(\boldsymbol{x}_i). \tag{2.1}$$

Minimum této funkce můžeme hledat např. pomocí *metody nejvyššího spádu*. Nyní máme zmíněno vše potřebné k optimalizaci odhadu hustoty pravděpodobnosti. Můžeme se tedy přesunout k popisu samotných modelů.

2.1 Modely využívající transformaci náhodné veličiny

Následující třída modelů (anglicky se tato třída nazývá tzv. *flow models*) využívá větu o transformaci náhodné veličiny. Uveďme si proto znění této věty nejprve v jednorozměrném případě.

Věta 2. Nechť X je absolutně spojitá náhodná veličina, $h : \mathbb{R} \to \mathbb{R}$ je ryze monotónní funkce na množině $X(\Omega)$ a h^{-1} je diferencovatelná. Potom náhodná veličina $Y = h(X)$ má hustotu

$$p_y(y) = p_x(h^{-1}(y)) \cdot \left| \frac{dh^{-1}}{dy}(y) \right|.$$

Dále si uveďme analogii této věty ve více dimenzích.

Věta 3. Nechť náhodný vektor $X_1, \ldots, X_D$ má sdružené absolutně spojité rozdělení a $(Y_1, \ldots, Y_D) = h(X_1, \ldots, X_D)$, kde $h : \mathbb{R}^D \to \mathbb{R}^D$ je spojité bijektivní zobrazení na otevřené množině G takové, že pro hustotu pravděpodobnosti p_x platí $\int \cdots \int_G p_x dx_1 \ldots dx_D = 1$. Nechť dále inverzní zobrazení h^{-1} je spojité, diferencovatelné a na množině $h(G)$ splňuje podmínku

$$\det \frac{\partial h^{-1}}{\partial \boldsymbol{y}} = \begin{vmatrix} \frac{\partial h_1^{-1}}{\partial y_1} & \cdots & \frac{\partial h_1^{-1}}{\partial y_D} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_D^{-1}}{\partial y_1} & \cdots & \frac{\partial h_D^{-1}}{\partial y_D} \end{vmatrix} \neq 0.$$

Potom i náhodný vektor $(Y_1, \ldots, Y_D)$ má sdružené absolutně spojité rozdělení a platí

$$p_y(y_1, \ldots, y_D) = p_x(h_1^{-1}(y_1, \ldots, y_D), \ldots, h_D^{-1}(y_1, \ldots, y_D)) \cdot \left| \det \frac{\partial h^{-1}}{\partial \boldsymbol{y}} \right|.$$

Jak nám tato věta poslouží k sestavení pravděpodobnostního modelu? Uvažujme opět soubor dat $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in (\mathcal{X}, p_x(\boldsymbol{x}))$ nezávislých pozorování. Dále zaveďme pomocnou absolutně spojitou náhodnou veličinu $\boldsymbol{Z}$ na prostoru $\mathbb{R}^D$ se známou hustotou pravděpodobnosti p_z. Dále klademe na náhodnou veličinu $\boldsymbol{Z}$ požadavek, aby byla z exponenciální třídy pravděpodobnostních rozdělení.

Poznámka. V této práce budeme typicky volit $\boldsymbol{Z} \sim \mathcal{N}(0, I)$, tedy normované Gaussovo rozdělení.

Inverzní zobrazení ve větě 3 označme jako $g := h^{-1}$. Dále nechť je zobrazení $g = g(\cdot, \theta)$ závislé na parametrech θ, tedy $g : \mathbb{R}^D \times \Theta \to \mathbb{R}^D$, kde Θ je parametrický prostor. Uvažujeme transformaci náhodné veličiny $\boldsymbol{Z} = g(\boldsymbol{X}, \theta)$, kde náhodná veličina $\boldsymbol{X}$ přísluší prostoru $(\mathcal{X}, p_x^\theta(\boldsymbol{x}))$. Hustota pravděpodobnosti p_x^θ pak dle věty 3 bude nabývat tvaru

$$p_x^\theta(\boldsymbol{x}) = p_z(g(\boldsymbol{x}, \theta)) \cdot \left| \det \frac{\partial g(\boldsymbol{x}, \theta)}{\partial \boldsymbol{x}} \right|. \tag{2.2}$$

Tímto jsme vytvořili množinu hustot pravděpodobností $\{p_x^\theta : \mathcal{X} \to \mathbb{R}_0^+,\ \theta \in \Theta \subseteq \mathbb{R}^N\}$. Dále vztah (2.2) zlogaritmujeme a upravíme na

$$\log p_x^\theta(\boldsymbol{x}) = \log p_z(g(\boldsymbol{x}, \theta)) + \log \left| \det \frac{\partial g(\boldsymbol{x}, \theta)}{\partial \boldsymbol{x}} \right|. \tag{2.3}$$

Tvar logaritmické hustoty pravděpodobnosti tvaru (2.3) se bude v následujících modelech často objevovat.

Poznámka. Pod zobrazením $g = g(\cdot, \theta)$ si v této chvíli čtenář může představit např. dopřednou neuronovou síť se stejným rozměrem vstupu a výstupu.

Dle maximálně věrohodného odhadu můžeme pro třídu modelů vycházejících ze vztahu (2.3) a pro data $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ zavést ztrátovou funkci ve tvaru

$$\begin{aligned}\mathcal{L}(\theta|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) &= -\frac{1}{n}\sum_{i=1}^{n} \log p_x^\theta(\boldsymbol{x}_i) \\ &= -\frac{1}{n}\sum_{i=1}^{n} \left(\log p_z(g(\boldsymbol{x_i}, \theta)) + \log \left|\det \frac{\partial g(\boldsymbol{x}, \theta)}{\partial \boldsymbol{x}}(\boldsymbol{x}_i)\right|\right).\end{aligned}$$

Pro takto navržené modely však obecně nastávají dva problémy. Za prvé, dle věty 3 musí být použité zobrazení $g = g(\cdot, \theta)$ invertibilní. Druhým problémem je vysoká náročnost výpočtu nacházejícího se determinantu Jacobiho matice $\left|\det \frac{\partial g(\boldsymbol{x}, \theta)}{\partial \boldsymbol{x}}(\boldsymbol{x}_i)\right|$. Náročnost tohoto výpočtu je konkrétně $\mathcal{O}(D^3)$, kde D je dimenze dat.

V následujících částech této kapitoly představíme konkrétní příklady modelů, které tyto dva problémy řeší.

2.1.1 Planární a radiální normalizační modely

Modely popsané v této sekci byly poprvé publikovány v [16]. Tato třída metod bude modelovat hustotu pravděpodobnosti ve tvaru

$$\log p_x^\theta(\boldsymbol{x}) = \log p_z(g(\boldsymbol{x}, \theta)) + \log \left|\det \frac{\partial g(\boldsymbol{x}, \theta)}{\partial \boldsymbol{x}}\right|,$$

kde g je invertibilní zobrazení a p_z známá hustota pravděpodobnosti (volme opět normované Gaussovo rozdělení). Nechť $g : \mathbb{R}^D \times \Theta \to \mathbb{R}^{\mathbb{D}}$ vznikne složením zobrazení $g_1, \ldots, g_m$, neboli $g = (g_m \circ \cdots \circ g_1)$, kde $g_i : \mathbb{R}^D \times \Theta_i \to \mathbb{R}^{\mathbb{D}}$, $\Theta_i \subseteq \Theta$ pro $i = 1, \ldots, m$, a $\bigoplus_{i=1}^m \Theta_i = \Theta$ ($\bigoplus$ značí direktní součet). Pokud budou všechna g_i invertibilní, pak i g bude invertibilní. Označme pro $\boldsymbol{x} = \boldsymbol{x}^0 \in \mathbb{R}^D$

$$\boldsymbol{x}^j = (g_j \circ \cdots \circ g_1)(\boldsymbol{x}^0), \quad j = 1, \ldots, m.$$

Logaritmus hustoty pravděpodobnosti p_x^θ můžeme pak použitím věty o derivaci složeného zobrazení a použitím vlastností logaritmu vyjádřit ve tvaru

$$\log p_x^\theta(\boldsymbol{x}) = \log p_z(g(\boldsymbol{x}, \theta)) + \sum_{j=1}^{m} \log \left|\det \frac{\partial g_j(\boldsymbol{x}^{j-1}, \theta_j)}{\partial \boldsymbol{x}^{j-1}}\right|,$$

kde $(\theta_1, \ldots, \theta_m) = \theta$. Pro soubor nezávislých pozorování $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in (\mathcal{X}, p_x(\boldsymbol{x}))$, kde $\mathcal{X} \subseteq \mathbb{R}^D$, můžeme dle maximálně věrohodného odhadu volit ztrátovou funkci ve tvaru

$$\mathcal{L}(\theta|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = -\frac{1}{n}\sum_{i=1}^{n} \left(\log p_z(g(\boldsymbol{x}_i, \theta)) + \sum_{j=1}^{m} \log \left|\det \frac{\partial g_j(\boldsymbol{x}_i^{j-1}, \theta_j)}{\partial \boldsymbol{x}_i^{j-1}}\right|\right).$$

Následující modely tedy volí všechna g_j invertibilní s dostatečným počtem parametrů θ_j.

Planární normalizační modely za zobrazení g_j, $j = 1, \ldots, m$, volí

$$g_j(\boldsymbol{x}) := \boldsymbol{x} + \boldsymbol{u}_j \cdot h_j(\boldsymbol{w}_j^T \boldsymbol{x} + b_j),$$

kde $\boldsymbol{u}_j, \boldsymbol{w}_j \in \mathbb{R}^D$, $b_j \in \mathbb{R}$ a kde $h_j : \mathbb{R} \to \mathbb{R}$ je invertibilní reálná funkce. Zároveň $(\boldsymbol{u}_j, \boldsymbol{w}_j, b_j) \in \Theta_j$ - jde tedy o parametry zobrazení g_j. Označme $\phi_j(\boldsymbol{x}) := h_j'(\boldsymbol{w}_j \boldsymbol{x} + b_j) \cdot \boldsymbol{w}_j$. Potřebný determinant Jacobiho matice jednoduše spočteme jako

$$\left| \frac{\partial g_j(\boldsymbol{x}, \theta_j)}{\partial \boldsymbol{x}} \right| = |1 + \boldsymbol{u}_j^T \phi_j(\boldsymbol{x}_j)|.$$

Radiální normalizační modely za zobrazení g_j, $j = 1, \ldots, m$, volí

$$\begin{aligned} g_j(\boldsymbol{x}) &:= \boldsymbol{x} + \beta_j \cdot h(\alpha_j, r)(\boldsymbol{x} - \boldsymbol{c}_j), \\ h(\alpha_j, r) &= \frac{1}{\alpha_j + r}, \\ r &= \|\boldsymbol{x} - \boldsymbol{x}_j\|_2, \end{aligned}$$

kde $\boldsymbol{c}_j \in \mathbb{R}^D$, $\beta_j \in \mathbb{R}$, $\alpha_j > 0$ a $(\boldsymbol{c}_j, \beta_j, \alpha_j) \in \Theta_j$ - opět jde o parametry zobrazení g_j. Determinant Jacobiho matice spočteme jako

$$\left| \frac{\partial g_j(\boldsymbol{x}, \theta_j)}{\partial \boldsymbol{x}} \right| = (1 + \beta_j h(\alpha_j, r))^{D-1}(1 + \beta_j h(\alpha_j, r) + \beta_j h'(\alpha_j, r) \cdot r).$$

Náročnost výpočtu determinantu je v obou případech $\mathcal{O}(D)$ - jde tedy o značnou redukci z $\mathcal{O}(D^3)$. Invertibilita zobrazení g je zaručena kladením dodatečných podmínek na jednotlivé parametry. Tyto podmínky lze však jednoduše algoritmicky zařídit (více k těmto podmínkám v dodatku článku [16]).

Takto navržené modely však nejsou příliš flexibilní. Jinými slovy nalezněme optimální parametr θ_0, pro který hustota $p_x^{\theta_0}$ maximalizuje věrohodnost na datech z prostoru $(\mathcal{X}, p_x)$. Pokud by např. hledaná hustota p_x vznikla složitou nelineární transformací z normalizovaného Gaussova rozdělení, potřebovali bychom velký počet jednotlivých zobrazení $g_1, \ldots, g_m$, abychom tuto nelineární transformaci aproximovali. Následující třídu modelů si můžeme představit jako spojitou analogii normalizačních modelů. Ty díky transformacím pomocí diferenciálních rovnic dokážou lépe aproximovat hypotetickou nelineární transformaci normalizovaného Gaussova rozdělení.

2.1.2 Spojité normalizační modely a metoda FFJORD

Spojité normalizační modely (poprvé publikovány v [9]) jsou metodou pro odhad hustoty pravděpodobnosti založené na tzv. **diferenciálních neuronových sítích** (opět publikovany ve stejné práci [9]). Metoda FFJORD (publikována v [7]) je pak pouze jejich modifikací pro redukci výpočtu determinantu Jacobiho matice. Koncept

diferenciálních neuronových je podrobně rozebrán v mé bakalářské práci [3] a jejich přesný popis je nad rámec této práce. Proto na tento typ sítí budeme nahlížet pouze jako na *černou skříňku*.

Mějme soustavu D obyčejných diferenciálních rovnic tvaru

$$\frac{d\boldsymbol{z}(t)}{dt} = f(\boldsymbol{z}(t), t, \theta) \tag{2.4}$$

řešenou na intervalu (t_0, t_1) s počáteční podmínkou $\boldsymbol{z}(t_0) = \boldsymbol{z}_0 \in \mathbb{R}^D$ a s parametry $\theta \in \Theta \subset \mathbb{R}^K$. Dále mějme ztrátovou funkci $\mathcal{L}$ závislou na koncovém řešení této soustavy, tedy

$$\mathcal{L} = \mathcal{L}(\boldsymbol{z}(t_1), \theta) = \mathcal{L}\left(\int_{t_0}^{t_1} f(\boldsymbol{z}(t), t, \theta)dt\right). \tag{2.5}$$

Příkladem takto zvolené ztrátové funkce může být

$$\mathcal{L}(\boldsymbol{z}(t_1)) = \sum_{i=1}^{D}((\boldsymbol{z}(t_1)_i)^2 - (\hat{\boldsymbol{z}}_i)^2), \tag{2.6}$$

kde $\hat{\boldsymbol{z}} \in \mathbb{R}^D$ je námi požadované koncové řešení soustavy (2.4). Naším cílem je nyní nalézt gradient $\frac{\partial \mathcal{L}}{\partial \theta}$, který spolu s optimalizačními algoritmy (např. pomocí metody největšího spádu) využijeme k nalezení parametrů $\hat{\theta}$, které minimalizují ztrátovou funkci $\mathcal{L}$, neboli

$$\mathcal{L}(\boldsymbol{z}(t_1), \hat{\theta}) \approx \min_{\theta \in \Theta} \mathcal{L}(\boldsymbol{z}(t_1), \theta).$$

Poznámka. Nechť soustava (2.4) např. popisuje trajektorii vystřeleného šípu. Tato trajektorie závisí na parametrech θ. Dále mějme terč se středem v $\hat{\boldsymbol{z}} \in \mathbb{R}^D$. Zavedeme-li ztrátovou funkci tvaru (2.6), pak minimalizací této funkce vůči parametrům θ nalezneme ideálně trajektorii, která zasáhne střed terče.

Diferenciální neuronové sítě jsou algoritmem, kterému poskytneme tvar obyčejné diferenciální rovnice (2.4) s konkrétním tvarem zobrazení f, počáteční podmínku $\boldsymbol{z}(t_0) = \boldsymbol{z}_0 \in \mathbb{R}^D$, interval (t_0, t_1), parametry θ a v poslední řadě ztrátovou funkci $\mathcal{L}$ typu (2.5). Algoritmus pak zpětným řešením tzv. *rozšířené soustavy obyčejných diferenciálních rovnic* nalezne hledaný gradient $\frac{\partial \mathcal{L}}{\partial \theta}$. Tento gradient pak můžeme využít pro požadovanou optimalizaci.

Vraťme se k odhadu hustoty pravděpodobnosti. Spojité normalizační modely využívají k transformaci hustoty pravděpodobnosti právě soustavu obyčejných diferenciálních rovnic. To, jak se bude s touto transformací měnit hustota pravděpodobnosti náhodné veličiny, nám osvětlí následující věta.

Věta 4 (Instanční změna náhodných veličin)**.** Nechť $\boldsymbol{Z}(t)$ je konečná v čase spojitá náhodná veličina s hustotou pravděpodobnosti $p_{z(t)}$ závislou na čase $t \in (t_0, t_1)$. Nechť je dále

$$\frac{d\boldsymbol{Z}}{dt} = f(\boldsymbol{Z}(t), t) \tag{2.7}$$

obyčejná diferenciální rovnice popisující spojitý vývoj náhodné veličiny $\boldsymbol{Z}(t)$ na intervalu (t_0, t_1) s počáteční podmínkou $\boldsymbol{Z}(t_0) = \boldsymbol{Z}_0$, kde $\boldsymbol{Z}_0$ je náhodná veličina s

hustotou pravděpodobnosti p_{z_0}. Pokud je f diferencovatelné zobrazení v proměnné $\boldsymbol{Z}(t)$ a spojité v proměnné t, pak vývoj logaritmu pravděpodobnostního rozdělení $p_{z(t)}$ popisuje následující diferenciální rovnice:

$$\frac{\partial \log(p_{z(t)}(\boldsymbol{z}(t)))}{\partial t} = -\operatorname{Tr}\left(\frac{\partial f}{\partial z(t)}\right), \tag{2.8}$$

pro $t \in (t_0, t_1)$ s počáteční podmínkou $\log(p_{z(t_0)}(\boldsymbol{z}(t_0))) = \log(p_{z_0}(\boldsymbol{z}_0))$.

Důkaz této věty byl poprvé zveřejněn v dodatku článku [9].

Poznámka. Tato věta je analogií věty 3 pro transformaci náhodné veličiny pomocí obyčejných diferenciálních rovnic.

Nyní sestavíme zmíněné **spojité normalizační modely**. Mějme opět soubor nezávislých pozorování $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in (\mathcal{X}, p_x(\boldsymbol{x}))$. Rovnice (2.7) a (2.8) ve větě 4 řešíme na stejném intervalu (t_0, t_1). Zapišme je proto do jedné soustavy obyčejných diferenciálních rovnic

$$\begin{aligned} \frac{d\boldsymbol{z}}{dt} &= f(\boldsymbol{z}(t), t), \\ \frac{\partial \log(p(\boldsymbol{z}(t))}{\partial t} &= -\operatorname{Tr}\left(\frac{\partial f}{\partial \boldsymbol{z}(t)}\right) \end{aligned} \tag{2.9}$$

na intervalu (t_0, t_1), kde $\boldsymbol{z}(t)$ značí pozorování náhodné veličiny $\boldsymbol{Z}(t)$.

Poznámka. V této soustavě jsme zanedbali značení spodního argumentu hustoty pravděpodobnosti $p_{z(t)}$. To, k jaké náhodné veličině hustota náleží, je jednoznačně určeno z argumentu hustoty $p(\boldsymbol{z}(t))$.

Dále se budeme zabývat tím, jak vhodně určit počáteční podmínku soustavy (2.9). Pro budoucí účely uveďme diferenciální rovnici v integrálním tvaru

$$\begin{aligned} \boldsymbol{z}(t_1) &= \boldsymbol{z}(t_0) + \int_{t_0}^{t_1} f(\boldsymbol{z}(t), t)dt, \\ \log(p(\boldsymbol{z}(t_1))) &= \log(p(\boldsymbol{z}(t_0))) + \int_{t_0}^{t_1} -\operatorname{Tr}\left(\frac{\partial f}{\partial \boldsymbol{z}(t)}\right) dt. \end{aligned} \tag{2.10}$$

Ve druhé rovnici odečteme na obou stranách výraz $\log(p_x(\boldsymbol{x}))$ - což je námi hledaná hustota pravděpodobnosti. Touto úpravou získáme tvar rovnic

$$\begin{aligned} \boldsymbol{z}(t_1) &= \boldsymbol{z}(t_0) + \int_{t_0}^{t_1} f(\boldsymbol{z}(t), t)dt, \\ \log(p(\boldsymbol{z}(t_1))) - \log(p_x(\boldsymbol{x})) &= (\log(p(\boldsymbol{z}(t_0))) - \log(p_x(\boldsymbol{x}))) + \int_{t_0}^{t_1} -\operatorname{Tr}\left(\frac{\partial f}{\partial \boldsymbol{z}(t)}\right) dt. \end{aligned} \tag{2.11}$$

Předpokládejme, že náhodná veličina $\boldsymbol{Z}_0$ dobře aproximuje náhodnou veličinu $\boldsymbol{X}$ na prostoru $(\mathcal{X}, p_x(\boldsymbol{x}))$. Neboli uvažujeme, že pokud pozorování $\boldsymbol{x} = \boldsymbol{z}_0$, pak $\log(p(\boldsymbol{z}(t_0))) \approx$

$\log(p_x(\boldsymbol{x}))$. Pro pozorování $\boldsymbol{x} \in (\mathcal{X}, p_x(\boldsymbol{x}))$ proto položme $\boldsymbol{z}(t_0) = \boldsymbol{x}$ a $\log(p(\boldsymbol{z}(t_0))) - \log(p_x(\boldsymbol{x})) = 0$. Rovnice se poté upraví na

$$\begin{aligned} \boldsymbol{z}(t_1) &= \boldsymbol{x} + \int_{t_0}^{t_1} f(\boldsymbol{z}(t), t)dt, \\ \log(p(\boldsymbol{z}(t_1))) - \log(p_x(\boldsymbol{x})) &= \int_{t_0}^{t_1} -\operatorname{Tr}\left(\frac{\partial f}{\partial \boldsymbol{z}(t)}\right) dt. \end{aligned} \tag{2.12}$$

Tuto integrální rovnici přepišme opět do diferenciálního tvaru

$$\begin{aligned} \frac{d\boldsymbol{z}}{dt} &= f(\boldsymbol{z}(t), t), \\ \frac{\partial \Delta_{log}(\boldsymbol{z}(t))}{\partial t} &= -\operatorname{Tr}\left(\frac{\partial f}{\partial \boldsymbol{z}(t)}\right) \end{aligned} \tag{2.13}$$

na intervalu (t_0, t_1) s **počáteční podmínkou** $\begin{bmatrix} \boldsymbol{z}(t_0) \\ \Delta_{log}(\boldsymbol{z}(t_0)) \end{bmatrix} = \begin{bmatrix} \boldsymbol{x} \\ 0 \end{bmatrix}$.

Všimněme si přeznačení trajektorie druhé rovnice ve výrazu (2.13) na $\Delta_{log}(\boldsymbol{z}(t))$. Řešením této rovnice totiž dostaneme rozdíl logaritmů hustot pravděpodobností. Za náhodnou veličinu $\boldsymbol{Z}(t_1) = \boldsymbol{Z}_1$ nyní zvolme veličinu s normovaným Gaussovým rozdělením, tedy se známou hustotou pravděpodobnosti $p_{z_1}(\boldsymbol{z}_1) = p(\boldsymbol{z}(t_1))$. Řešením diferenciální rovnice (2.13) v čase t_1 s příslušnou počáteční podmínkou bude

$$\begin{bmatrix} \boldsymbol{z}(t_1) \\ \Delta_{log}(\boldsymbol{z}(t_1)) \end{bmatrix} = \begin{bmatrix} \boldsymbol{z}(t_1) \\ \log(p(\boldsymbol{z}(t_1))) - \log(p_x(\boldsymbol{x})) \end{bmatrix} = \begin{bmatrix} \boldsymbol{z}_1 \\ \log(p_{z_1}(\boldsymbol{z}_1)) - \log(p_x(\boldsymbol{x})) \end{bmatrix}.$$

Nyní máme vše potřebné pro vyjádření hustoty pravděpodobnosti p_x v bodě $\boldsymbol{x}$, jelikož

$$\log(p_{z_1}(\boldsymbol{z}_1)) - \Delta_{log}(\boldsymbol{z}(t_1)) = \log(p_{z_1}(\boldsymbol{z}_1)) - (\log(p_{z_1}(\boldsymbol{z}_1)) - \log(p_x(\boldsymbol{x}))) = \log(p_x(\boldsymbol{x})).$$

Zjednodušeně, řešením první rovnice v (2.13) získáme hodnotu $\boldsymbol{z}_1$. Tuto hodnotu dosadíme do známé hustoty p_{z_1} a poté odečteme řešení druhé rovnice. Pro přehlednost zapišme hustotu p_x v integrálním tvaru

$$\begin{aligned} p_x(\boldsymbol{x}) &= \log(p(\boldsymbol{z}(t_1))) - \Delta_{log}(\boldsymbol{z}(t_1)) \\ &= \log p_{z_1}\left(\boldsymbol{x} + \int_{t_0}^{t_1} f(\boldsymbol{z}(t), t)dt\right) - \int_{t_0}^{t_1} -\operatorname{Tr}\left(\frac{\partial f}{\partial \boldsymbol{z}(t)}\right) dt. \end{aligned} \tag{2.14}$$

Do této chvíle jsme však předpokládali, že zobrazení f transformuje pomocí diferenciální rovnice náhodnou veličinu $\boldsymbol{Z}_1$ (která aproximuje $\boldsymbol{X}$) na náhodnou veličinu $\boldsymbol{Z}_1$ se známým rozdělením. Jak ovšem zobrazení f najít? Dosaďme za f neuronovou síť s rozměrem vstupu a výstupu D s parametry θ, tedy $f = f(\cdot, \theta)$ (explicitní závislost na čase neuvažujeme). Tato volba nám rekapitulací předešlého postupu vytvoří hustotu pravděpodobnosti

$$p_x^\theta(\boldsymbol{x}) = \log p_{z_1}\left(\boldsymbol{x} + \int_{t_0}^{t_1} f(\boldsymbol{z}(t), \theta)dt\right) - \int_{t_0}^{t_1} -\operatorname{Tr}\left(\frac{\partial f}{\partial \boldsymbol{z}(t)}(\boldsymbol{z}(t), \theta)\right) dt.$$

Nyní pro soubor $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in (\mathcal{X}, p_x(\boldsymbol{x}))$ opět využijeme maximálně věrohodného odhadu a sestavíme ztrátovou funkci tvaru

$$\mathcal{L}(\theta|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = -\frac{1}{n}\sum_{i=1}^{n}\left(\log p_{\boldsymbol{z}_1}\left(\boldsymbol{x}_i + \int_{t_0}^{t_1} f(\boldsymbol{z}(t), \theta)dt\right)\right.$$
$$\left. - \int_{t_0}^{t_1} -\operatorname{Tr}\left(\frac{\partial f}{\partial \boldsymbol{z}(t)}(\boldsymbol{z}(t), \theta)\right)dt\right).$$

Vidíme, že tato ztrátová funkce je přímo závislá na řešení diferenciální rovnice (2.13), tedy je typu funkce (2.5). Z tohoto faktu plyne, že pro optimalizaci můžeme použít zmíněné diferenciální neuronové sítě.

Vraťme se ke dvěma problémům zmíněných v první části této kapitoly - zajištění invertibility a náročnost výpočtu determinantu Jacobiho matice. Invertibilita je v tomto případě zajištěna tím, že soustavu obyčejných diferenciálních rovnic můžeme řešit pozpátku. Náročnost výpočtu determinantu se díky nahrazení operátorem stopy matice zredukuje na $\mathcal{O}(D^2)$ (tedy na náročnost výpočtu samotné Jacobiho matice).

Tímto jsme popsali metodu spojitých normalizačních modelů. Tu dále modifikujeme na metodu zvanou **FFJORD** (z anglického *Free-form Jacobian of Reversible Dynamics*). Tato metoda aproximuje hodnotu stopy Jacobiho matice ve větě 4.

Použitá aproximace se nazývá Hutchinsonův odhad, který byl poprvé zveřejněn v [8]. Pro zavedení tohoto odhadu mějme absolutně spojitou náhodnou veličinu ϵ rozměru D. Po této veličině požadujeme, aby splňovala $\mathbb{E}[\epsilon] = 0$ a $\operatorname{Cov}[\epsilon] = I$, kde $\operatorname{Cov}[\epsilon]$ značí kovarianční matici. Typicky opět volíme ϵ z normovaného Gaussova rozdělení. Mějme matici $\boldsymbol{A} \in \mathbb{R}^{D\times D}$. Pro její stopu pak dle Hutchinsonova odhadu platí

$$\operatorname{Tr}(\boldsymbol{A}) = \mathbb{E}_\epsilon[\epsilon^T \boldsymbol{A}\epsilon]. \tag{2.15}$$

Uvedeme si pouze formální odvození tohoto odhadu:

$$\operatorname{Tr}(\boldsymbol{A}) = \operatorname{Tr}(\boldsymbol{A}I) = \operatorname{Tr}(\boldsymbol{A}\mathbb{E}[\epsilon\epsilon^T]) =$$
$$= \mathbb{E}[\operatorname{Tr}(\boldsymbol{A}\epsilon\epsilon^T)] = \mathbb{E}[\epsilon^T\boldsymbol{A}\epsilon],$$

kde jsme na prvním řádku díky předpokladům použili vztah

$$\mathbb{E}[\epsilon\epsilon^T] = \operatorname{Cov}[\epsilon] + \mathbb{E}[\epsilon]\mathbb{E}[\epsilon]^T =$$
$$= \operatorname{Cov}[\epsilon] = I.$$

Hutchinsonův odhad použijeme v soustavě diferenciálních rovnic (2.13) čímž získáme soustavu pro metodu FFJORD ve tvaru

$$\frac{d\boldsymbol{z}}{dt} = f(\boldsymbol{z}(t), t),$$
$$\frac{\partial \Delta_{log}(\boldsymbol{z}(t))}{\partial t} = -\mathbb{E}_\epsilon\left[\epsilon^T\left(\frac{\partial f}{\partial \boldsymbol{z}(t)}\right)\epsilon\right] \tag{2.16}$$

na intervalu (t_0, t_1) s **počáteční podmínkou** $\begin{bmatrix}\boldsymbol{z}(t_0)\\ \Delta_{log}(\boldsymbol{z}(t_0))\end{bmatrix} = \begin{bmatrix}\boldsymbol{x}\\ 0\end{bmatrix}$, kde $\epsilon \sim \mathcal{N}(0, I)$.

Na první pohled nemusí být zřejmé, kde nastala redukce náročnosti výpočtu oproti stopě Jacobiho matice. Trik spočívá v tom, že nebudeme počítat celou Jacobiho matici $\frac{\partial f}{\partial \boldsymbol{z}(t)}$, ale pouze derivaci ve směru ϵ, tedy přímo hodnotu $\left(\frac{\partial f}{\partial \boldsymbol{z}(t)}\right) \cdot \epsilon$. Takzvané zpětné metody pro výpočet derivace dokáží derivaci ve směru spočítat s náročností $\mathcal{O}(D)$ (Přímý výpočet determinantu Jacobiho matice má náročnost $\mathcal{O}(D^3)$ - jedná se tedy o značné urychlení). Po tomto výpočtu hodnotu jednoduše přenásobíme transponovaným vektorem ϵ^T.

Nyní opět analogicky ke spojitým normalizačním modelům vyjádřeme hustotu pravděpodobnosti p_x^θ při použití zobrazení $f = f(\cdot, \theta)$ ve tvaru

$$p_x^\theta(\boldsymbol{x}) = \log p_{z_1}\left(\boldsymbol{x} + \int_{t_0}^{t_1} f(\boldsymbol{z}(t), \theta) dt\right) - \mathbb{E}_\epsilon\left[\int_{t_0}^{t_1} -\epsilon^T \left(\frac{\partial f}{\partial \boldsymbol{z}(t)}(\boldsymbol{z}(t), \theta)\right) \epsilon dt\right], \tag{2.17}$$

kde jsme provedly záměnu střední hodnoty a integrálu. V této chvíli můžeme pro soubor nezávislých pozorování $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in (\mathcal{X}, p_x(\boldsymbol{x}))$ zavést pomocí maximálně věrohodného odhadu ztrátovou funkci

$$\begin{aligned}\mathcal{L}(\theta|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = -\frac{1}{n}\sum_{i=1}^{n}\Bigg(&\log p_{\boldsymbol{z}_1}\left(\boldsymbol{x}_i + \int_{t_0}^{t_1} f(\boldsymbol{z}(t), \theta) dt\right) \\ &- \int_{t_0}^{t_1} -\epsilon_i^T \left(\frac{\partial f}{\partial \boldsymbol{z}(t)}(\boldsymbol{z}(t), \theta)\right) \epsilon_i dt\Bigg),\end{aligned}$$

kde jsme ke každému vzorku $\boldsymbol{x}_i$, $i = 1, \ldots, n$, vygenerovali náhodnou hodnotu ϵ_i z normovaného Gaussova rozdělení. Pro dostatečně vysoká n jsme takto aritmetickým průměrem odhadly střední hodnotu nacházející se v Hutchinsonově odhadu. Z tohoto faktu ovšem plyne, že metodu FFJORD můžeme v tomto tvaru použít pouze pro učení modelů - neboli při používání ztrátové funkce v tomto tvaru. Pokud bychom chtěli určit hustotu pravděpodobnosti individuálního pozorování $\boldsymbol{x}_0 \in (\mathcal{X}, p_x(\boldsymbol{x}))$, museli bychom sestavit analogický model spojitých normalizačních modelů a přenést nalezené parametry θ metodou FFJORD do tohoto nového modelu. Spojitý normalizační model s těmito parametry pak určí přesnou hustotu pravděpodobnosti individuálního pozorování.

Poznámka. Při implementaci metody FFJORD je nutné, abychom jednotlivé hodnoty ϵ_i, $i = 1, \ldots, n$, generovali mimo výpočet integrálu. Jinými slovy nesmíme v každém kroku např. Runge-Kuttova algoritmu generovat nová ϵ_i.

K trénování modelu FFJORD můžeme opět využít zmíněné diferenciální neuronové sítě.

2.1.3 Metoda MAF a RealNVP

Metody MAF [13] a RealNVP [4] opět využívají transformaci náhodné veličiny k modelování hustoty pravděpodobnosti. Implementací těchto modelů jsem se osobně

nezabýval, ale využijeme ve výpočetní studii této práce. Z tohoto důvodu si je nebudeme rozebírat podrobně jako např. metodu FFJORD, ale pouze si v této části práce zmíníme jejich hlavní podstatu.

Metoda MAF opět transformuje normované Gaussovo rozdělení $\mathcal{N}(\mathbf{0}, I)$ s hustotou pravděpodobnosti p_z. Mějme opět pozorování s neznámou hustotou pravděpodobnosti $\boldsymbol{x} \sim p_x(\boldsymbol{x})$ rozměru $D \in \mathbb{N}$. Zaveďme dvě maskované neuronové sítě f_μ a f_α s rozměry vstupu a výstupu D (více o maskovaných neuronových sítích v [13]). Proveďme transformaci tvaru

$$\boldsymbol{z} = (\boldsymbol{x} - \boldsymbol{\mu}) \cdot \exp(-\boldsymbol{\alpha}), \text{ kde } \boldsymbol{\mu} = f_\mu(\boldsymbol{x}) \text{ a kde } \boldsymbol{\alpha} = f_\alpha(\boldsymbol{x}).$$

Výraz $\exp(-\boldsymbol{\alpha})$ představuje vyčíslení exponenciální funkce po prvcích vektoru. Jde o inverzi transformace Gaussova rozdělení $\mathcal{N}(\mathbf{0}, I)$ na Gaussovo rozdělení $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$, kde $\Sigma \in \mathbb{R}^{D \times D}$ a $\operatorname{diag}(\Sigma) = \exp(\boldsymbol{\alpha})^2$. Výslednou hustotu pravděpodobnosti spočteme jako

$$p_x^\theta(\boldsymbol{x}) = p_z(\boldsymbol{z}) + \exp\left(\sum_i^D \alpha_i\right),$$

kde θ představuje parametry použitých neuronových sítí a kde $\exp\left(\sum_i^D \alpha_i\right)$ představuje determinant Jacobiho matice užité transformace.

Přesuňme se ke stručnému popisu metody RealNVP. Mějme opět pozorování $\boldsymbol{x} \sim p_x(\boldsymbol{x})$ a hustotu pravděpodobnosti normovaného Gaussova rozdělení p_z. Označme $\boldsymbol{x}_{1:d} = (x_1, \ldots, x_d)^T$, kde $d < D$. Proveďme pro $d < D$ transformaci

$$\begin{aligned} \boldsymbol{z}_{1:d} &= \boldsymbol{x}_{1:d} \\ \boldsymbol{z}_{d+1:D} &= \boldsymbol{x}_{d+1:D} \odot \exp(s(\boldsymbol{x}_{1:d})) + t(\boldsymbol{x}_{1:d}), \end{aligned}$$

kde s a t značí škálovací a translační funkci z $\mathbb{R}^d \to \mathbb{R}^{D-d}$ a kde operátor $\odot$ značí násobení po prvcích. Modelovanou hustotu pravděpodobnosti poté spočteme jako

$$p_x^\theta(\boldsymbol{x}) = p_z(\boldsymbol{z}) + \left|\det \frac{\partial \boldsymbol{z}}{\partial \boldsymbol{x}}\right|,$$

kde nacházející se jacobiho matice má tvar

$$\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{x}} = \begin{pmatrix} I_d & \mathbf{0} \\ \frac{\partial \boldsymbol{z}_{d+1:D}}{\partial \boldsymbol{x}_{d+1:D}} & \operatorname{diag}(\exp(s(\boldsymbol{x}_{1:d}))) \end{pmatrix}.$$

Parametry θ představují prvky matice a vektoru vyskytujících se v zobrazeních s a t.

Jednotlivé transformace můžeme v obou případech dále skládat jako v případě planárních a radiálních normalizačních modelů.

2.2 Modely s grafovou reprezentací

Modely s grafovou reprezentací jsou další třídou metod pro odhad hustoty pravděpodobnosti. Známým představitelem těchto modelů jsou např. Bayesovské sítě

[11]. Obecně tyto modely různými způsoby kombinují zvolené pravděpodobnostní distribuce. Toto kombinování a vztahy jednotlivých distribucí poté dokážeme reprezentovat pomocí grafu - ve většině případů orientovaným acyklickým grafem. V této kapitole si popíšeme dva modely tohoto typu. Konkrétně se bude jednat o tzv. **Součtové-produktové sítě**, které si podrobně rozebereme, a poté jejich koncept rozšíříme na tzv. **Součtové-produktové transformační sítě**.

2.2.1 Součtové-produktové sítě

Definice modelu **Součtových-produktových sítí** (anglicky *Sum-product networks* - v textu proto budeme používat zkratku **SPN**, publikováno v [15]) je poměrně abstraktní. V první řadě tuto definici vyslovíme a pak si podrobně na ukázkových příkladech rozebereme princip tohoto modelu. Nejprve však vyslovme potřebné definice z teorie grafů, které budeme v budoucnu používat.

Definice 5. Nechť $G = (V, E)$ je orientovaný acyklický graf.

- Množinu potomků vrcholu v označíme $\mathrm{Ch}(v)$, $v \in V$.
- Množinu všech listů (vrcholů bez potomků) grafu G označíme $l(G)$.
- Množinu všech kořenů (vrcholů bez rodičů) grafu G označíme $r(G)$.
- Množinu $G \backslash (l(G) \cup r(G))$ nazveme množinou vnitřních vrcholů G.

V následující definici SPN se objeví i dosud nedefinované pojmy. Definici těchto pojmů vyslovíme ihned poté, abychom se vyhnuli dlouhé a komplikované definici.

Definice 6 (SPN)**.** Nechť $\mathfrak{X} = (\Omega, \mathcal{A}, \mathbb{P})$ je pravděpodobnostní prostor, kde $\mathcal{A}$ je σ-algebra na Ω. Součtová-produktová síť je uspořádaná trojice (G, ψ, Θ), kde $G = (V, E)$ je orientovaný acyklický graf s jedním kořenem $r \in r(G)$, Θ je parametrický prostor sítě a $\psi : V \to 2^{\mathfrak{X}}$ je **rozsahová funkce**, kde $2^{\mathfrak{X}}$ značí množinu

$$2^{\mathfrak{X}} = \left\{ (\hat{\Omega}, \hat{\mathcal{A}}, \hat{\mathbb{P}}) |\ \hat{\Omega} \subset \Omega, \hat{\mathcal{A}} \text{ je } \sigma\text{-algebra na } \hat{\Omega},\ \hat{\mathbb{P}} \text{ je pravděpodobnostní míra} \right\}.$$

Vnitřní vrcholy a kořen tohoto grafu představují takzvané součtové a součinové vrcholy (definované později). Každou hranu vystupující ze součtového vrcholu i do libovolného vrcholu j ohodnotíme vahou $w_{ij} > 0$. Tato váha je složkou prvku prostoru Θ.

Označme $u \in V, \phi(u) := (\Omega_u, \mathcal{A}_u, \mathbb{P}_u)$. Dále požadujeme, aby

1. $(\forall v \in V \backslash l(G)) \left(\Omega_v = \bigcup_{u \in \mathrm{Ch}(v)} \Omega_u \wedge \bigcup_{u \in \mathrm{Ch}(v)} \mathcal{A}_u \subset \mathcal{A}_v \right)$
2. $\psi(r) = \mathfrak{X}$
3. $(\forall \text{ součinové uzly } v \in V) \left(\bigcap_{u \in \mathrm{Ch}(v)} \mathcal{A}_u = \emptyset \right)$ (Rozložitelnost)

4. $(\forall$ součtové uzly $v \in V)(\forall u \in \mathrm{Ch}(v))\,(\Omega_u = \Omega_v \wedge \mathcal{A}_u = \mathcal{A}_v)\,.$ (Úplnost)

Poznámka. Vrcholy grafu G Součtové-produktové sítě budeme také nazývat jako **uzly** sítě.

Tato definice byla vyslovena s obecnou mírou pravděpodobnosti $\mathbb{P}$. V této práci budeme předpokládat, že tato pravděpodobnostní míra bude dále absolutně spojitá. Nechť $\boldsymbol{X} = (X_1, \ldots, X_D)$ je D-rozměrná náhodná veličina na prostoru $\mathfrak{X}$. Mějme její pozorování $\boldsymbol{x} = (x_1, \ldots, x_D)^T \in \mathbb{R}^D$. Na listech grafu G sami zvolíme pravděpodobnostní distribuce. Tyto distribuce budou buďto příslušet celé náhodné veličině $\boldsymbol{X}$, nebo pouze její marginální části - např. (X_1, X_3, X_D), nebo např. pouze jedné složce X_2. Distribuce opět volíme absolutně spojité se známou hustotou pravděpodobnosti.

Příklad. Pro marginální část náhodné veličiny (X_1, X_3, X_D) zvolme distribuci na příslušném listu jako Gaussovké rozdělení se střední hodnotou $\boldsymbol{\mu} \in \mathbb{R}^3$ a kovarianční maticí $\Sigma \in \mathbb{R}^{3\times 3}$. Pro pozorování $\boldsymbol{x} = (x_1, \ldots, x_D)^T$ náhodné veličiny $\boldsymbol{X}$ vypočteme hustotu pravděpodobnosti na tomto listu jako

$$p(\boldsymbol{x}) = p(x_1, x_3, x_D) = \frac{\exp(-\frac{1}{2}(x_1, x_3, x_D)^T - \boldsymbol{\mu})^T \Sigma^{-1}(x_1, x_3, x_D)^T - \boldsymbol{\mu}))}{\sqrt{(2\pi)^3 \det \Sigma}}.$$

Poznámka. Zde nastává bohužel nekorektní použití statistických pojmů. Mějme náhodnou veličinu $\boldsymbol{X}$ na prostoru $\mathfrak{X}$ a její marginální část - opět např. (X_1, X_3, X_D). Z předešlého příkladu bylo na listu voleno třírozměrné Gaussovské rozdělení, kterému přísluší třírozměrná náhodná veličina $(\hat{X}_1, \hat{X}_3, \hat{X}_D)$. Zřejmě bude platit, že $(X_1, X_3, X_D) \neq (\hat{X}_1, \hat{X}_3, \hat{X}_D)$ jakožto funkcí z prostoru náhodných jevů do Borelovských množin. V budoucnu pokud řekneme, že pro marginální část náhodné veličiny (X_1, X_3, X_D) volíme distribuci s hustotou pravděpodobnosti $\hat{p}$, pak tím myslíme, jak vyčíslit tuto hustotu pravděpodobnosti v příslušné části pozorování náhodné veličiny $\boldsymbol{X}$. Poprosil bych čtenáře o odpuštění při nekorektním používání těchto matematických pojmů.

Parametry zvolených hustot pravděpodobností budou složkami prvku parametrického prostoru Θ v SPN. Z podmínky 2. v definici 6 vyplývá, že listy sítě musejí svými distribucemi dohromady pokrývat celou náhodnou veličinu $\boldsymbol{X}$. Nyní se přesuneme k definicím zmíněných součtových a součinových uzlů, které udávají tvar distribucí (resp. hustot pravděpodobností) na na zbylých uzel SPN.

Definice 7 (Součtový uzel)**.** Nechť $\mathfrak{X} = (\Omega, \mathcal{A}, \mathbb{P})$ je pravděpodobnostní prostor s absolutně spojitou pravděpodobnostní mírou, $S = (G, \psi, \Theta)$ je SPN na $\mathfrak{X}$ a nechť $i \in G \backslash l(G)$ je libovolný vrchol G, který není listem. Pak tento vrchol nazveme **součtovým uzlem** (značíme znaménkem "+"), pokud pro hustotu pravděpodobnosti p_i příslušné distribuce tohoto uzlu platí

$$p_i(\boldsymbol{x}) = \sum_{j \in \mathrm{Ch}(i)} w_{ij} p_j(\boldsymbol{x}),$$

kde p_j značí hustoty pravděpodobností potomků j uzlu i.

Poznámka. Nechť $\boldsymbol{X}$ je náhodná veličina na $\mathfrak{X}$. Dle požadavku 4. (Úplnost) v definici SPN musí být všechny hustoty p_j potomků součtového uzlu závislé na stejných marginálních částech náhodné veličiny $\boldsymbol{X}$. Např. pro pozorování $\boldsymbol{x}$ musí pro všechny potomky j platit $p_j(\boldsymbol{x}) = p_j(x_1, x_3, x_D)$. Z toho plyne, že i pro součtový uzel i platí $p_i(\boldsymbol{x}) = p_i(x_1, x_3, x_D)$

Definice 8 (Součinový uzel)**.** Nechť $\mathfrak{X} = (\Omega, \mathcal{A}, \mathbb{P})$ je pravděpodobnostní prostor s absolutně spojitou pravděpodobnostní mírou, $S = (G, \psi, \Theta)$ je SPN na $\mathcal{X}$ a nechť $v \in G \backslash l(G)$ je libovolný vrchol G, který není listem. Pak tento vrchol nazveme **součinovým uzlem** (značíme znaménkem "$\times$"), pokud pro hustotu pravděpodobnosti p_v příslušné distribuce tohoto uzlu platí

$$p_v(\boldsymbol{x}) = \prod_{u \in \mathrm{Ch}(v)} p_u(\boldsymbol{x}),$$

kde p_u značí hustoty pravděpodobností potomků u uzlu v.

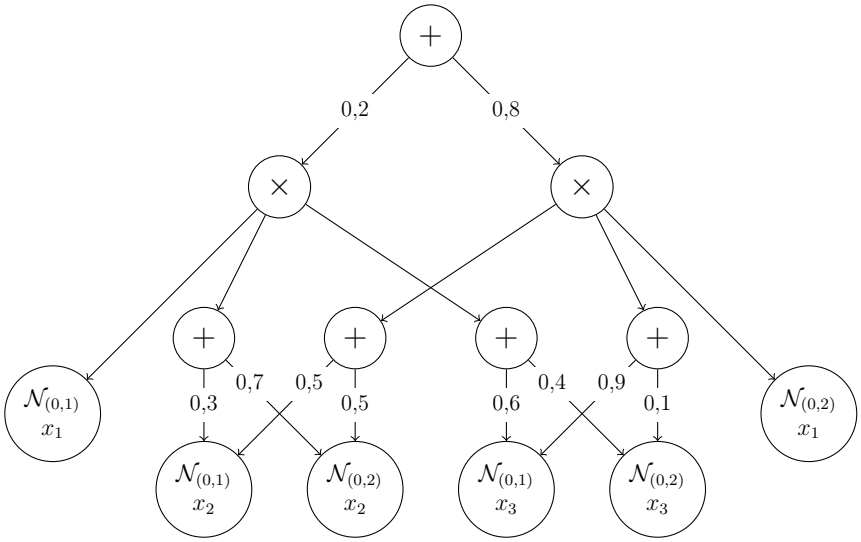
Poznámka. Nechť $\boldsymbol{X}$ je náhodná veličina na $\mathfrak{X}$. Dle požadavku 3. (Rozlišitelnost) v definici SPN musí být všechny hustoty p_u potomků uzlu závislé na jiných marginálních částech náhodné veličiny $\boldsymbol{X}$. Pokud např. budeme mít dva potomky součinového uzlu u a w, jejich hustoty mohou být např. závislé od $p_u(\boldsymbol{x}) = p_u(x_1, x_3, x_D)$ a $p_w(\boldsymbol{x}) = p_w(x_2)$. Naopak nesmí např. nastat situace $p_u(\boldsymbol{x}) = p_u(x_1, x_3, x_D)$ a $p_w(\boldsymbol{x}) = p_w(x_1)$. Požadavek rozlišitelnosti dále implikuje, že disjunktní marginální části náhodné veličiny $\boldsymbol{X}$ jsou na sobě nezávislé.

Definice 9. Nechť $S = (G, \psi, \theta)$ je SPN. S nazveme **normovanou**, pokud pro každý součtový uzel i sítě S platí

$$\sum_{j \in \mathrm{Ch}(i)} w_{ij} = 1.$$

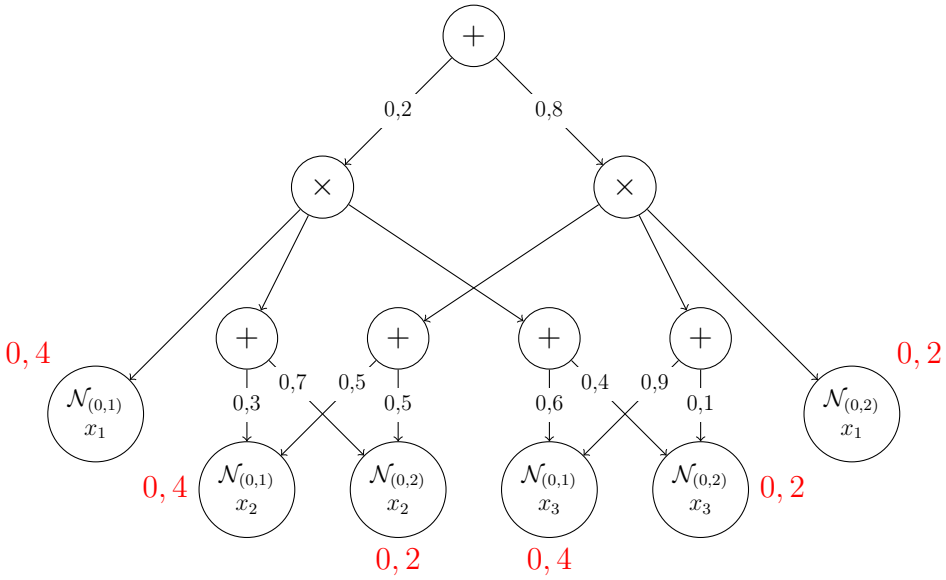
Dále budeme vždy předpokládat, že následující příklady SPN jsou normované.

Na následující sérii obrázku uvedeme příklad výpočtu hustoty pravděpodobnosti na SPN. Mějme prostor $\mathfrak{X} = (\Omega, \mathcal{A}, \mathbb{P})$ s 3-rozměrnou náhodnou veličinou $\boldsymbol{X}$ a její pozorování $\boldsymbol{x} \in \mathbb{R}^3$. Příklad SPN příslušné prostoru $\mathfrak{X}$ můžeme vidět na obrázku 2.1.



Obrázek 2.1: Příklad grafu normované SPN s listy představující jednorozměrná Gaussova rozdělení s parametry $\mu = 0,\ \sigma^2 = 1$ a $\mu = 0,\ \sigma^2 = 2$. Veličina x_i, $i = 1, 2, 3$, popisuje, které marginální části náhodné veličiny $\boldsymbol{X}$ dané rozdělení odpovídá.

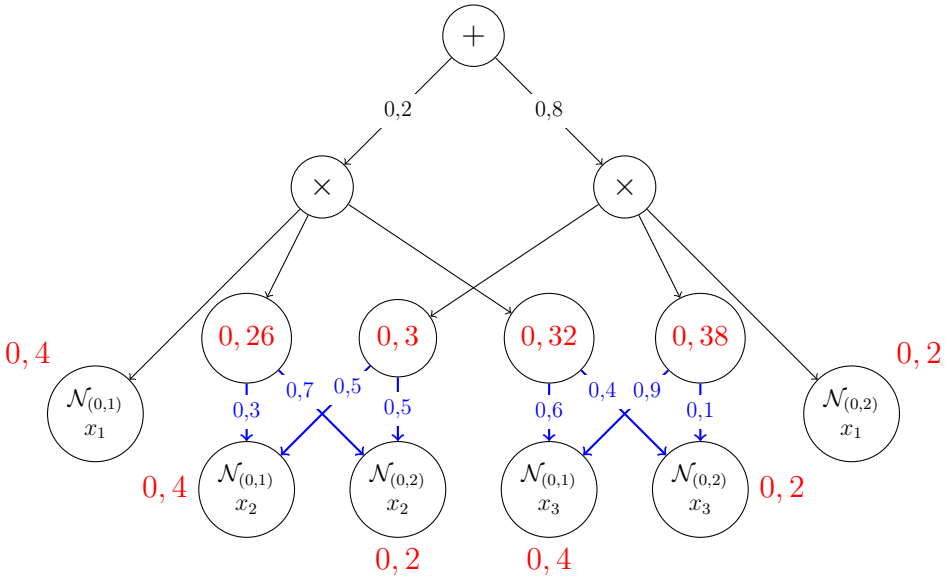
Mějme pozorování s hodnotami $\boldsymbol{x} = (x_1, x_2, x_3) = (0, 0, 0)$. Pro každou složku náhodné veličiny nejprve vyčíslíme hodnotu hustoty pravděpodobnosti normalizovaného Gaussova rozdělení, resp. Gaussova rozdělení s $\mu = 0,\ \sigma^2 = 2$, v bodě 0. Pro hustotu normálního rozdělení v bodě 0 platí $p(x) \approx 0,4$, resp. $p(x) \approx 0,2$ pro $\sigma^2 = 2$. Na obrázku 2.2 můžeme vidět SPN s hodnotami hustot pravděpodobností v bodech 0 označené červenou barvou.



Obrázek 2.2: Příklad grafu normované SPN. Červenou barvou jsou vyznačený hustoty příslušných distribucí pro pozorování $\boldsymbol{x} = (0, 0, 0)$.

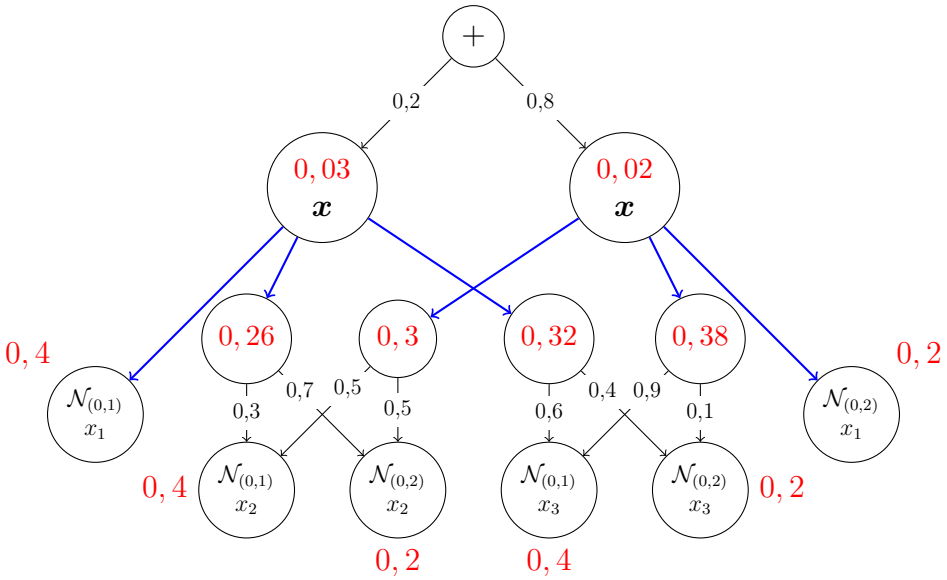
Na obrázku 2.3 můžeme vidět hodnoty hustot pravděpodobností na součtových uz-

lech dle definice 7.



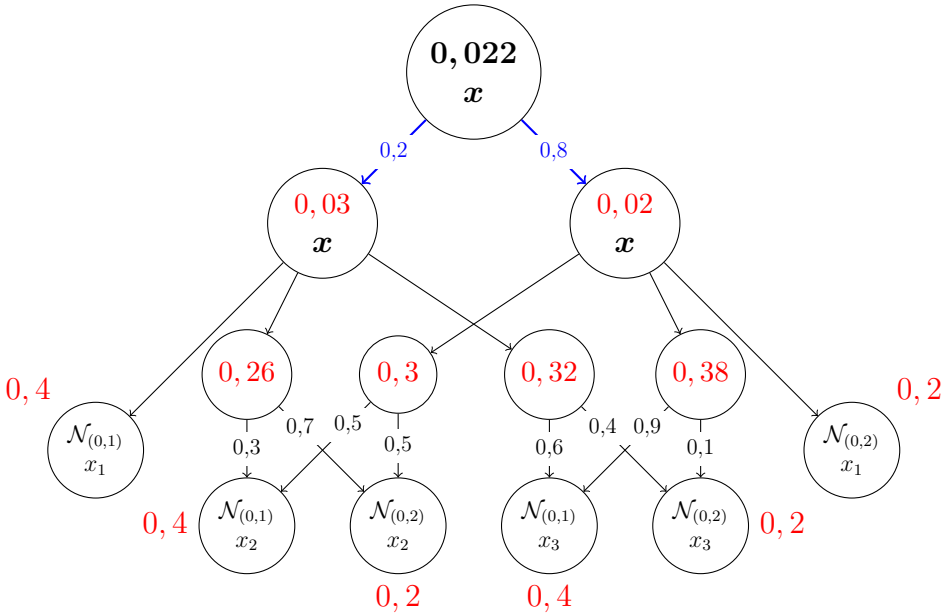
Obrázek 2.3: Příklad grafu normované SPN s hodnotami součtových uzlů pro $\boldsymbol{x} = (0, 0, 0)$.

Dále na obrázku 2.4 je uvedena SPN s hodnotami hustot pravděpodobností na součinových uzlech.



Obrázek 2.4: Příklad grafu normované SPN s hodnotami součinových uzlů pro $\boldsymbol{x} = (0, 0, 0)$.

V posledním obrázku 2.5 můžeme vidět hodnotu hustoty pravděpodobnosti v kořeni SPN (zde se jedná opět o součtový uzel).



Obrázek 2.5: Příklad grafu normované SPN s celkovou hustotou pravděpodobnosti sítě pro $\boldsymbol{x} = (0, 0, 0)$.

Hustota pravděpodobnosti této SPN, označme ji p_x^θ, bude pro pozorování $\boldsymbol{x} = (0, 0, 0)$ nabývat hodnoty $p_x^\theta(\boldsymbol{x}) \approx 0,022$. Parametry θ zde značí prvek prostoru Θ z definice SPN. Jedná se o parametry použitých rozdělení na listech grafu a vah hran vycházejících ze součtových uzlů.

Poznámka. Silnou vlastností SPN oproti většině pravděpodobnostních modelů je možnost výpočtu marginálních hustot pravděpodobností. Tato vlastnost je zaručena díky předpokladům 3. a 4. v definici 6. Podrobně na příkladech je možnost marginalizace vysvětlena v mém výzkumném úkolu [2].

Mějme opět soubor nezávislých pozorování $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in (\mathcal{X}, p_x(\boldsymbol{x}))$, $\mathcal{X} \subseteq \mathbb{R}^D$. Dle maximálně věrohodného odhadu můžeme opět sestavit ztrátovou funkci tvaru

$$\mathcal{L}(\theta|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = -\frac{1}{n}\sum_{i=1}^{n} \log p_x^\theta(\boldsymbol{x}_i).$$

Pokud je model SPN dostatečně robustní, bude pro optimální $\theta_0 \in \Theta$ platit $p_x^{\theta_0} \approx p_x$.

2.2.2 Součtové-produktové transformační sítě

Jak již bylo zmíněno, Součtové-produktové transformační sítě (anglicky *Sum-product transform networks*, v textu budeme dále používat zkratku **SPTN**, SPTN byly poprvé zveřejněny v [14]) jsou modifikací popsaných SPN. V modelu SPN jsme kombinovali různé pravděpodobnostní distribuce pouze pomocí lineárních transformací. V modelu SPTN budeme schopni modelovat nelineární transformace jednotlivých distribucí pomocí zavedení tzv. transformačního uzlu. Nejprve vyslovíme potřebné definice k tomuto modelu.

Definice 10. Nechť $G = (V, E)$ je orientovaný acyklický graf. Poté řekneme, že $w \in V$ dělí hranu mezi $v, u \in V, v \neq u$, pokud

1. $((v, w) \in E \wedge (w, u) \in E) \vee ((u, w) \in E \wedge (w, v) \in E)$
2. $(\forall t \in V)(t \neq u \wedge t \neq v)((w, t) \notin E \wedge (t, w) \notin E)$.

Definice 11 (SPTN)**.** Nechť $\mathfrak{X} = (\Omega, \mathcal{A}, \mathbb{P})$ je pravděpodobnostní prostor, kde $\mathcal{A}$ je σ-algebra na Ω. Součtová-produktová transformační síť je uspořádaná trojice (G, ψ, Θ), kde $G = (V, E)$ je orientovaný acyklický graf s jedním kořenem $r \in r(G)$, Θ je parametrický prostor sítě a $\psi : V \to 2^{\mathfrak{X}}$ je **rozsahová funkce**, kde $2^{\mathfrak{X}}$ značí množinu

$$2^{\mathfrak{X}} = \left\{ (\hat{\Omega}, \hat{\mathcal{A}}, \hat{\mathbb{P}}) |\ \hat{\Omega} \subset \Omega, \hat{\mathcal{A}} \text{ je } \sigma\text{-algebra na } \hat{\Omega},\ \hat{\mathbb{P}} \text{ je pravděpodobnostní míra} \right\}.$$

Vnitřní vrcholy a kořen tohoto grafu představují takzvané součtové, součinové a transformační vrcholy.

Každou hranu vystupující ze součtového vrcholu i do libovolného vrcholu j ohodnotíme vahou $w_{ij} > 0$. Tato váha je složkou prvku prostoru Θ.

Dále požadujeme, aby každý transformační vrchol w dělil hranu mezi dvěma vrcholy $u, v \in G$, $w \neq v$, $w \neq u$, $v \neq u$. Parametry užité v transformačním vrcholu jsou složky prvku prostoru Θ.

Označme $u \in V, \phi(u) := (\Omega_u, \mathcal{A}_u, \mathbb{P}_u)$. Dále požadujeme, aby

1. $(\forall v \in V \backslash l(G)) \left(\Omega_v = \bigcup_{u \in \mathrm{Ch}(v)} \Omega_u \wedge \bigcup_{u \in \mathrm{Ch}(v)} \mathcal{A}_u \subset \mathcal{A}_v \right)$
2. $\psi(r) = \mathfrak{X}$
3. $(\forall \text{ součinové uzly } v \in V) \left(\bigcap_{u \in \mathrm{Ch}(v)} \mathcal{A}_u = \emptyset \right)$ (Rozložitelnost)
4. $(\forall \text{ součtové uzly } v \in V)(\forall u \in \mathrm{Ch}(v)) \left(\Omega_u = \Omega_v \wedge \mathcal{A}_u = \mathcal{A}_v \right)$. (Úplnost)

Poznámka. Pro prostor $\mathfrak{X}$ s absolutně spojitou mírou zůstávají definice součtových a součinových uzlů stejné jako v modelu SPN. V tomto modelu poté na libovolnou hranu můžeme *položit* transformační uzel, jehož definici si nyní vyslovíme.

Definice 12 (Transformační uzel)**.** Nechť $\mathfrak{X} = (\Omega, \mathcal{A}, \mathbb{P})$ je pravděpodobnostní prostor s absolutně spojitou pravděpodobnostní mírou, $T = (G, \psi, \theta)$ je SPTN na $\mathfrak{X}$, $v \in G \backslash l(G)$ je libovolný vrchol G, který není listem, a nechť $f(\cdot, \eta) : \mathbb{R}^m \to \mathbb{R}^m$ je diferencovatelné, bijektivní zobrazení (η značí parametry zobrazení f, které jsou složkami prvku parametrického prostoru Θ). Pak vrchol v nazveme **transformačním uzlem** (značíme znaménkem "f", dle použitého zobrazení), pokud pro hustotu pravděpodobnosti p_v příslušné distribuce tohoto uzlu platí

$$p_v(\boldsymbol{x}) = p_u(f(\boldsymbol{x})) \cdot \left| \det \left(\frac{\partial f}{\partial \boldsymbol{x}}(\boldsymbol{x}) \right) \right|,$$

kde p_u značí hustotu pravděpodobnosti potomka u uzlu v (dle definice SPTN jediného) a $\frac{\partial f}{\partial x}$ značí Jacobiho matici (resp. derivaci pro jednorozměrný případ) zobrazení f. Dále m značí dimenzi marginální části náhodné veličiny $\boldsymbol{X}$ na prostoru $\mathfrak{X}$ příslušné uzlu u.

Poznámka. V modelech využívajících transformaci náhodné veličiny jsme uváděli transformaci hustoty veličiny v logaritmickém tvaru. Obecně se častěji pracuje právě s hustotami v tomto tvaru. V následujícím odstavci si proto uvedeme transformace hustot pravděpodobnostní na součtových, součinových a transformačních uzlech v logaritmickém tvaru:

1. Součtový uzel: $\log p_i(\boldsymbol{x}) = \log\left(\sum_{j\in \mathrm{Ch}(i)} \exp(\log(w_{ij}) + \log(p_j(\boldsymbol{x})))\right)$.
   (Tento tvar je volen kvůli přítomnosti $\log(p_j(\boldsymbol{x}))$)
2. Součinový uzel: $\log p_v(\boldsymbol{x}) = \sum_{u\in \mathrm{Ch}(v)} \log p_u(\boldsymbol{x})$.
3. Transformační uzel: $\log p_v(\boldsymbol{x}) = \log p_u(f(\boldsymbol{x})) + \log\left(\left|\det\left(\frac{\partial f}{\partial \boldsymbol{x}}(\boldsymbol{x})\right)\right|\right)$.

Nyní nastává otázka, jak volit zobrazení f užité pro transformaci hustoty pravděpodobnosti. Můžeme např. volit zobrazení popsané v sekci 2.1.1 - tedy zobrazení užité v planárních a radiálních normalizačních modelech.

Ve výpočetní studii této práce budeme však využívat modely SPTN, ve kterých za zobrazení f v transformačních uzlech volíme jednu vrstvu neuronové sítě. Mějme regulární matici $\boldsymbol{A} \in \mathbb{R}^{m\times m}$, vektor $b \in \mathbb{R}^m$ a aktivační funkci $\sigma : \mathbb{R}^m \to \mathbb{R}^m$ (zpravidla ostře rostoucí a diferencovatelnou). Zobrazení f poté zavedeme jako

$$f(\boldsymbol{x}) = \sigma\left(\boldsymbol{A}\boldsymbol{x} + b\right).$$

Díky vysloveným předpokladům budeme takto volené zobrazení invertibilní. Jelikož je matice $\boldsymbol{A}$, můžeme jednoznačně najít její SVD rozklad

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T,$$

kde $\boldsymbol{U}, \boldsymbol{V} \in \mathbb{R}^{m\times m}$ jsou unitární matice a $\boldsymbol{D} \in \mathbb{R}^{m\times m}$ je matice diagonální. Pro změnu logaritmu determinantu potřebnou pro transformaci hustoty pravděpodobnosti bude pro zobrazení f v tomto tvaru platit

$$\log\left(\left|\det\left(\frac{\partial f}{\partial \boldsymbol{x}}(\boldsymbol{x})\right)\right|\right) = \sum_{i=1}^{d} \log|d_{ii}| + \sum_{i=1}^{d} \log\left|\frac{\partial \sigma_i}{\partial o_i}\right|,$$

kde $o = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T\boldsymbol{x} + b$ a kde d_{ii} jsou diagonální prvky matice $\boldsymbol{D}$. Pokud aktivační funkci σ zvolíme jako identitu, dostaneme afinní transformaci.

Můžeme samozřejmě použít i komplikovanější transformace hustoty pravděpodobnosti. Např. můžeme v transformačních uzlech použít metodu FFJORD - přesněji vztah (2.17). Tímto případem jsem se podrobně zabýval ve svém výzkumném úkolu [2].

Na obrázku 2.6 můžeme vidět příklad SPTN, kterou jsme vytvořili z SPN na obrázku 2.1 vložením transformačních uzlů před listy grafu.

Všimněme si, že na tomto příkladu jsme na všech listech zvolili všechna pravděpodobnostní rozdělení jako normovaná Gaussova. Toto jsme si na rozdíl od modelu SPN mohli dovolit právě díky vložení transformačních uzlů. Předpokládáme, že užitá zobrazení v transformačních uzlech budou natolik robustní, že dokáží potřebně modifikovat parametry rozdělení na listech grafu (změnu parametrů normovaného Gaussova rozdělení např. docílíme použitím pouhé afinní transformace).

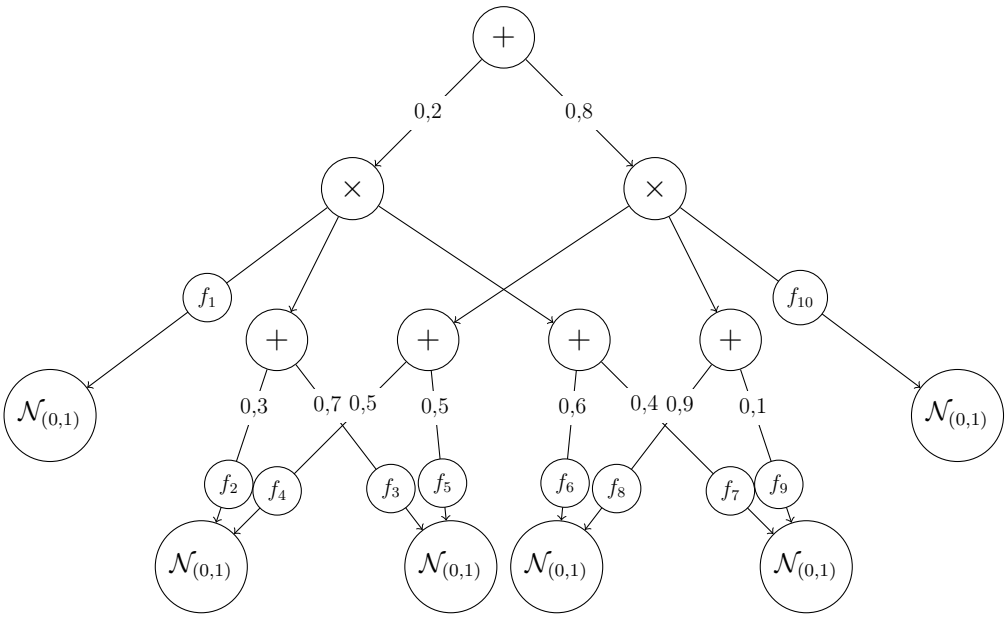
Jinými slovy můžeme volit parametry použitých rozdělení na listech grafu jako fixní - tedy nespadají do parametrického prostoru Θ v modelu SPTN. Z tohoto důvodu můžeme obrázek 2.6 zakreslit v kompaktnějším tvaru na obrázku 2.7.

Poznámka. Díky obrázku 2.7 můžeme na takto sestavenou SPTN nahlížet jako na jeden z modelů popsaných v sekci 2.1. Neboli na výslednou hustotu pravděpodobnosti SPTN můžeme nahlížet jako na přetransformovanou hustotu normovaného Gaussova rozdělení.

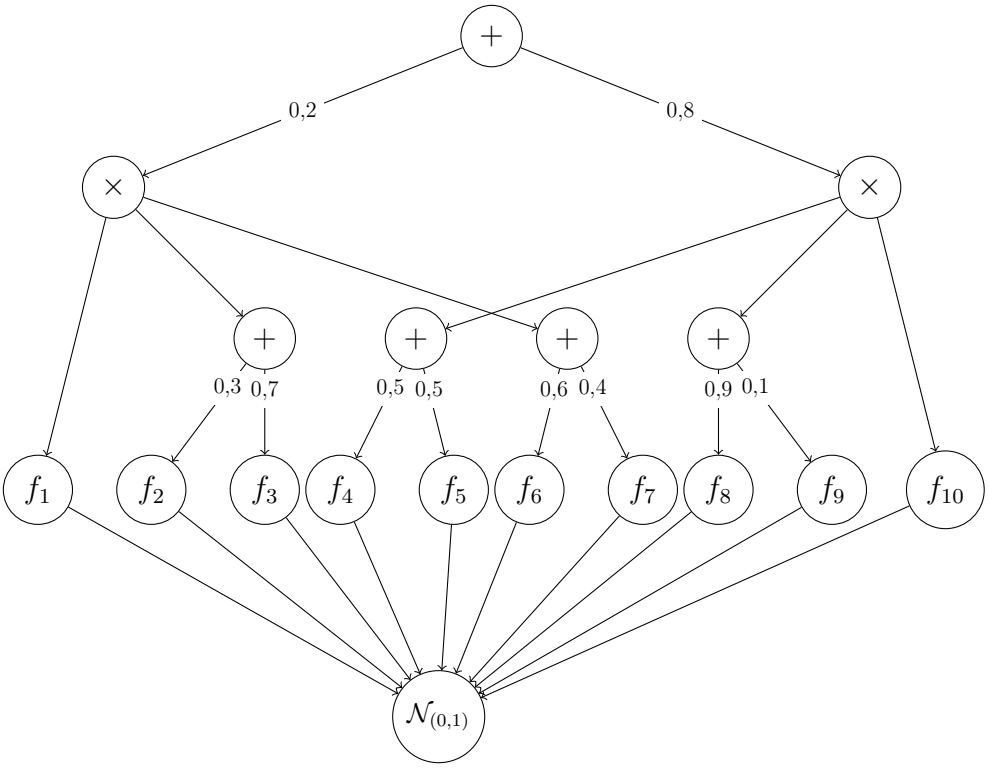
Analogickým postupem jako v SPN s přidáním transformačních uzlů můžeme vypočítat hustotu pravděpodobnosti p_x^θ. Pro nezávislá pozorování $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in (\mathcal{X}, p_x(\boldsymbol{x}))$, $\mathcal{X} \subseteq \mathbb{R}^D$ pak opět zavedeme ztrátovou funkci tvaru

$$\mathcal{L}(\theta|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = -\frac{1}{n}\sum_{i=1}^{n} \log p_x^\theta(\boldsymbol{x}_i).$$

Tímto jsme zakončili popis pravděpodobnostních modelů pro odhad hustoty pravděpodobnosti. V následující kapitole se budeme zabývat jejich využitím pro detekci anomálií.



Obrázek 2.6: Příklad grafu normované SPTN vytvořené z SPN na obrázku 2.1. Listy představují jednorozměrná normovaná Gaussova rozdělení. $f_1, \ldots, f_{10}$ jsou užitá zobrazení pro transformační uzly.



Obrázek 2.7: Příklad kompaktnějšího zakreslení SPTN na obrázku 2.6.

Kapitola 3

Detekce anomálií pomocí odhadu hustoty pravděpodobnosti

V této kapitole se budeme převážně zabývat evaluaci jednotlivých modelů odhadujících hustotu pravděpodobnosti pro detekci anomálií. Nejprve zmíníme, jak porovnat jednotlivé modely pouze z hlediska jejich schopnosti odhadu hustoty pravděpodobnosti. Mějme nezávislá reálná data $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in (\mathcal{X}, p_x(\boldsymbol{x}))$, $\mathcal{X} \subseteq \mathbb{R}^D$. Nejprve data rozdělíme na tzv. trénovací a testovací soubor (validační soubor pro jednoduchost zde uvažovat nebudeme). Rozdělme data na tyto dva soubory např. v poměru $4:1$, jinými slovy náhodným výběrem vybereme 80% dat do trénovacího souboru a zbytek vložíme do testovacího souboru. Označme trénovací soubor jako $\boldsymbol{x}_1^{\text{train}}, \ldots, \boldsymbol{x}_m^{\text{train}}$ a testovací jako $\boldsymbol{x}_1^{\text{test}}, \ldots, \boldsymbol{x}_s^{\text{test}}$, kde $m + s = n$. Trénovací soubor použijeme na trénování modelu, neboli pro nalezení optimálních parametrů

$$\theta_0 \approx \operatorname*{arginf}_{\theta \in \Theta} \mathcal{L}(\theta | \boldsymbol{x}_1^{\text{train}}, \ldots, \boldsymbol{x}_m^{\text{train}}).$$

Pokud tyto parametry nalezneme spolu s tvarem hustoty pravděpodobnosti $p_x^{\theta_0}$, můžeme ohodnotit schopnost modelu odhadnout hledanou hustotu pravděpodobnosti celkovou věrohodností na testovacích datech, tedy hodnotou

$$\ell(\boldsymbol{x}_1^{\text{test}}, \ldots, \boldsymbol{x}_s^{\text{test}} | \theta_0) = \prod_{i=1}^{s} p_x^{\theta_0}(\boldsymbol{x}_i^{\text{test}}).$$

Čím větší tato hodnota bude, tím větší bude pravděpodobnost, že se model přiblížil ke skutečné hustotě pravděpodobnosti p_x.

V detekci anomálií je bohužel evaluace poněkud komplikovanější. Uvažujme soubor dat $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in (\mathcal{X}, p_x(\boldsymbol{x}))$ s hladinou normality α (opět připomeňme, že např. pro $\alpha = 0,05$ předpokládáme, že 5% dat bude anomálních). Pro jednoduchost zanedbáme rozdělení na trénovací a testovací soubor. Na těchto datech naučíme model s výslednou hustotou pravděpodobnosti $p_x^{\theta_0}$. Dále empiricky určíme práh normality vztahem

$$\hat{\tau}_\alpha = \sup_{\tau} \left\{ \tau \geq 0,\ \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{[0, p_x^{\theta_0}(\boldsymbol{x}_i))}(\tau) \geq 1 - \alpha \right\}$$

a zavedeme klasifikátor anomálií ve tvaru

$$c_\alpha(\boldsymbol{x}) = \begin{cases} +1 & \text{pro } p_x^{\theta_0}(\boldsymbol{x}) \leq \hat{\tau}_\alpha, \\ -1 & \text{pro } p_x^{\theta_0}(\boldsymbol{x}) > \hat{\tau}_\alpha. \end{cases}$$

Pro $\alpha = 0,05$ nám tento klasifikátor vyhodnotí 5% dat jako anomálních. My ovšem ve valné většině případů nebudeme mít apriorní informaci o tom, jaká data jsou a nejsou normální. Pokud by se např. jednalo o 1000 medicínských dat, potřebovali bychom ke každé hodnotě znalecký posudek lékaře, zda-li se jedná, či nejedná o anomálii. V krajním případě by mohla nastat situace, že všechny anomálie budou klasifikovány jako normální a všechny klasifikované anomálie budou ve skutečnosti normální - nemáme možnost zjistit, zda-li tato situace nastala. Pokud ovšem tuto apriorní informaci máme, jedná se o silnou výhodu. Rozdělme proto data pro evaluaci detekce anomálií do tří tříd:

1. **Data více tříd** - na počátku máme soubor dat rozdělených do dvou nebo více tříd. Jednu část tříd označíme jako normální a zbylou část jako anomální. Nevýhodou této varianty je pravděpodobné vykazování silného systematického chování anomálních dat.

2. **Data se skutečnými anomáliemi** - jedná se o soubor dat, kde expert v daném oboru přesně určil, jaká data jsou normální a jaká anomální. Jedná se o ideální případ, v praxi je však málo častý.

3. **Data s umělými anomáliemi** - máme k dispozici soubor dat. Všechna tato data označíme jako normální a sami si uměle vygenerujeme data, která označíme jako anomální.

Ve výpočetní studii této práce se budeme zabývat všemi případy. Nevýhodou třetího případu je, že pokud anomální data budou vykazovat slabé systematické chování, tak toto chování nebudeme schopni v našem modelu zaznamenat. Pokud bychom opět měli medicínská data a věděli bychom, že anomálie je ve většině případů způsobena chybným měřením krevního tlaku, tak bychom neměli možnost tuto skutečnost v modelu zahrnout.

Dále tedy budeme předpokládat, že anomálie nevykazují jakékoliv systematické chování. Umělé anomálie vytvoříme tak, že pro soubor dat $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in (\mathcal{X}, p_x(\boldsymbol{x}))$ zavedeme kompaktní množinu $\boldsymbol{K} \subset \mathcal{X}$ tak, že $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \boldsymbol{K}$. Na této kompaktní množině zavedeme rovnoměrné rozdělení s hustotou pravděpodobnosti $u_x = u_x(\boldsymbol{x})$. Dle tohoto rozdělení vygenerujeme na množině $\boldsymbol{K}$ soubor dat $\hat{\boldsymbol{x}}_1, \ldots, \hat{\boldsymbol{x}}_m$. Data $\hat{\boldsymbol{x}}_1, \ldots, \hat{\boldsymbol{x}}_m$ označíme jako anomální, nebo-li hodnotou $+1$ (tedy jako pozitivní). Data $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ označíme jako normální, nebo-li hodnotou -1 (tedy jako negativní).

Předpoklad, že anomální data uvažujeme z rovnoměrného rozdělení, nám umožní v následující sekci vyslovit rigorózní tvrzení, které dovolí chápat detekci anomálií jako klasifikační problém.

3.1 Detekce anomálií jako klasifikační problém

Teorie popsaná v této sekci práce se opírá o článek [19].

Mějme prostor $\mathcal{X} \subseteq \mathbb{R}^D$ s hustotou pravděpodobnosti p_x^+. Připomeňme definici množiny anomálií $\mathcal{A}$ z kapitoly 1

$$\mathcal{A} = \{\boldsymbol{x} \in \mathcal{X},\ p_x^+(\boldsymbol{x}) \leq \tau_\alpha\},\ \tau_\alpha \geq 0,$$

kde τ_α je práh normality na hladině $\alpha \in (0, 1)$. Označme množinu normálních dat jako

$$\{p_x^+ > \tau_\alpha\} := \mathcal{X} \setminus \mathcal{A} = \{\boldsymbol{x} \in \mathcal{X},\ p_x^+(\boldsymbol{x}) > \tau_\alpha\}.$$

Mějme klasifikátor anomálií $c_\alpha : \mathbb{R}^D \to \mathbb{R}$. Uveďme opět klasifikátor anomálií modelů odhadujících hustotu pravděpodobnosti ve tvaru

$$c_\alpha(\boldsymbol{x}) = \begin{cases} +1 & \text{pro } \ p_x^{\theta_0}(\boldsymbol{x}) \leq \hat{\tau}_\alpha, \\ -1 & \text{pro } \ p_x^{\theta_0}(\boldsymbol{x}) > \hat{\tau}_\alpha, \end{cases}$$

kde $p_x^{\theta_0}$ je odhad hustoty pravděpodobnosti a $\hat{\tau}_\alpha$ je empirický odhad prahu normality na hladině α.

Zaveďme množinu

$$\{c_\alpha < 0\} := \{\boldsymbol{x} \in \mathcal{X}, c_\alpha(\boldsymbol{x}) < 0\}.$$

Dále opět pro jednoduchost předpokládejme, že pro data $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in (\mathcal{X}, p_x(\boldsymbol{x}))$ platí $p_x \equiv p_x^+$ - jinými slovy zanedbáváme vzniklý šum při měření dat.

Schopnost klasifikátoru korektně detekovat anomálie můžeme určit pomocí tzv. výkonnostní míry S_{μ,p_x,τ_α} definované vztahem

$$S_{\mu,p_x,\tau_\alpha}(c_\alpha) := \mu\left(\{c_\alpha < 0\} \triangle \{p_x > \tau_\alpha\}\right), \tag{3.1}$$

kde operátor $\triangle$ značí symetrickou diferenci. Pokud nalezneme klasifikátor $\hat{c}_\alpha$ s vlastností $S_{\mu,p_x,\tau_\alpha}(\hat{c}_\alpha) = 0$, můžeme se domnívat, že tento klasifikátor bude ideální.

Míru S_{μ,p_x,τ_α} však nejsme ve většině případů schopni přímo vypočítat, jelikož neznáme tvar množiny $\{p_x > \tau_\alpha\}$. V následující části této sekce se budeme zabývat odhadem výkonnostní míry. Jak již bylo v této kapitole naznačeno, tento odhad bude založen na vygenerování umělých anomálií z rovnoměrného rozdělení na předem určené množině.

Poznámka. Pravděpodobnostní míra rovnoměrného rozdělení na kompaktní množině je až normovací faktor ekvivalentní Lebeguovské míře na této množině. Následující teorie je založena na Lebeguovské míře, ale samozřejmě bychom byli schopni analogicky tuto teorii zavést s mírou rovnoměrného rozdělení.

Pomocí výše uvedeného postupu zavedeme soubor normálních dat $(\boldsymbol{x}_1, -1), \ldots, (\boldsymbol{x}_n, -1)$ a soubor uměle vytvořených anomálií $(\hat{\boldsymbol{x}}_1, +1), \ldots, (\hat{\boldsymbol{x}}_m, +1)$. Dohromady tyto soubory tvoří množinu prvků prostoru $\mathcal{X} \times \mathcal{Y}$, kde $\mathcal{Y} = \{-1, +1\}$. Nyní na prostoru $\mathcal{X} \times \mathcal{Y}$ zavedeme pomocnou míru.

Definice 13. Nechť P je pravděpodobnostní míra na prostoru $\mathcal{X}$, μ je Lebegueovská míra na prostoru $\mathcal{X}$ a $s \in (0,1)$. Míru $P \ominus_s \mu$ na prostoru $\mathcal{X} \times \mathcal{Y}$ definujeme vztahem

$$P \ominus_s \mu(\boldsymbol{A}) = s\mathbb{E}_{\boldsymbol{x}\sim P}[\mathbb{1}_{\boldsymbol{A}}(\boldsymbol{x}, -1)] + (1-s)\int_{\boldsymbol{A}} \mathbb{1}_{\boldsymbol{A}}(\boldsymbol{x}, +1)d\boldsymbol{x}$$

pro měřitelné množiny $\boldsymbol{A} \subset \mathcal{X} \times \mathcal{Y}$. Užíváme zkrácený zápis charakteristické funkce na prostoru $\mathcal{X} \times \mathcal{Y}$ daný vztahem $\mathbb{1}_{\boldsymbol{A}}(\boldsymbol{x}, y) := \mathbb{1}_{\boldsymbol{A}}((\boldsymbol{x}, y))$.

Definice 14 (Riziko binárního klasifikátoru)**.** Nechť $c_\alpha : \mathbb{R}^D \to \mathbb{R}$ je binární klasifikátor, $\mathcal{X} \subseteq \mathbb{R}^D$ je prostor s pravděpodobnostní mírou P a $s \in (0,1)$. Nechť c_α je měřitelná funkce na prostoru $\mathcal{X}$. Mějme míru Q na prostoru $\mathcal{X} \times \mathcal{Y}$, kde $\mathcal{Y} = \{-1, +1\}$, danou $Q := P \ominus_s \mu$. **Riziko binárního klasifikátoru** $\mathcal{R}_Q(c_\alpha)$ definujeme vztahem

$$\mathcal{R}_Q(c_\alpha) = Q\left(\{(\boldsymbol{x}, y),\ \operatorname{sign} c_\alpha(\boldsymbol{x}) \neq y\}\right).$$

Bayesovské riziko $\mathcal{R}_Q$ vůči Q je dále definováno jako

$$\mathcal{R}_Q := \inf\left\{\mathcal{R}_Q(c_\alpha),\ c_\alpha : \mathbb{R}^D \to \mathbb{R} \text{ měřitelné}\right\}.$$

Pomocí pravděpodobnostní teorie a teorie míry lze dokázat následující tvrzení, které dává do vztahu právě výkonnostní míru a riziko binárního klasifikátoru.

Věta 5. Nechť P je pravděpodobnostní míra na $\mathcal{X}$ s hustotou pravděpodobnosti p_x, μ je Lebegeuovská míra na $\mathcal{X}$ a $\mathcal{Y} = \{-1, +1\}$. Nechť $\tau_\alpha > 0$ je práh normality, pro který platí

$$\mu\left(\{\boldsymbol{x} \in \mathcal{X},\ p_x(\boldsymbol{x}) = \tau_\alpha\right) = 0.$$

Dále volíme $s := \frac{1}{1+\tau_\alpha}$ a zavedeme míru $Q := P \ominus_s \mu$ na prostoru $\mathcal{X} \times \mathcal{Y}$. Pak pro všechny posloupnosti (c_n) měřitelných funkcí, $c_n : \mathbb{R}^D \to \mathbb{R}$, jsou následující výroky ekvivalentní:

1. $S_{\mu,p_x,\tau_\alpha}(c_n) \to 0$.
2. $\mathcal{R}_Q(c_n) \to \mathcal{R}_Q$.

Poměrně komplikovaný důkaz této věty je založen na teorii míry a je k dohledání v článku [19].

Nyní bychom mohli v této teorii pokračovat a např. zavést empirické odhady hodnoty $\mathcal{R}_Q(c_\alpha)$ pro námi zvolený klasifikátor c_α. V rámci této práce se však spokojíme s jednodušší interpretací rizika binárního klasifikátoru.

Mějme k dispozici nezávislá pozorování $(\boldsymbol{x}_1, -1), \ldots, (\boldsymbol{x}_n, -1) \in (\mathcal{X}, p_x(\boldsymbol{x})) \times \{-1\}$ a kompaktní množinu $\boldsymbol{K} \subset \mathcal{X}$, která tato pozorování pokrývá. Na kompaktní množině $\boldsymbol{K}$ poté vygenerujeme dle rovnoměrného rozdělení (které je až na normovací faktor ekvivalentní s Lebesguovou mírou na $\boldsymbol{K}$) umělé anomálie $(\hat{\boldsymbol{x}}_1, +1), \ldots, (\hat{\boldsymbol{x}}_m, +1) \in \boldsymbol{K} \times \{+1\}$. Pokud budeme mít řadu různých klasifikátorů $c_1, \ldots, c_l$, kde $c_i : \mathbb{R}^D \to \{\pm 1\}$, $i = 1, \ldots, l$, pak za nejlepší klasifikátor z této třídy můžeme volit ten, který

korektně klasifikuje největší počet poskytnutých dat a umělých anomálií. Takto zvolený klasifikátor bude pravděpodobně nejvíce minimalizovat riziko binárního klasifikátoru a tudíž dle věty 5 nejlépe vystihne tvar množiny normálních dat. Intuitivní předpoklad toho, že rovnoměrně rozložené umělé anomálie mohou zastoupit reálné anomálie, máme nyní matematicky podložen.

Při klasifikaci dále musíme rozlišovat dvě situace chybovosti klasifikátoru. Tyto dvě rozdílné situace nazveme jako:

1. **Chyba prvního druhu** - nesprávně označíme normální pozorování jako anomálii.
2. **Chyba druhého druhu** - nesprávně označíme anomálii jako normální pozorování.

Poznámka. Pokud tímto postupem vygenerujeme umělé anomálie, tak se nám s velkou pravděpodobností stane, že některé anomálie budou ležet blízko normálních dat a tudíž budou klasifikovány jako normální (chyba druhého druhu). Naopak pokud soubor dat bude obsahovat skutečné anomálie, které dle popsaného postupu označíme jako normální, tak s nejvyšší pravděpodobností je klasifikátor označí skutečně jako anomálie (chyba prvního druhu). Kouzlem výše popsané teorie spočívá v tom, že i přes tyto situace neztratíme při takto vykonstruovaném srovnání jednotlivých klasifikátorů výpovědní hodnotu o jejich schopnosti anomálie detekovat.

Tyto dva případy musíme zahrnout v evaluaci detekce anomálií v následující části této kapitoly.

3.2 Evaluace klasifikátorů anomálií

V této části nejprve navážeme přímo na chybu prvního a druhého druhu. Připomeňme si, že normální data jsme označili hodnotou -1, neboli negativně. Klasifikujeme-li chybně normální pozorování jako anomální (hodnotou $+1$), nastane chyba prvního druhu a pozorování nazveme **falešně pozitivní**. Naopak klasifikujeme-li chybně anomální pozorování, tedy chybně ho klasifikujeme hodnotou -1, nazveme pozorování jako **falešně negativní**. Pokud pozorování klasifikujeme správně, označíme ho jako **pravě negativní**, resp. **pravě pozitivní**.

Pro jednoduchost označíme počty takto definovaných pozorování pomocí následujícího značení:

$$\begin{aligned} FP &= \text{počet falešně pozitivních pozorování},\\ FN &= \text{počet falešně negativních pozorování},\\ TP &= \text{počet pravě pozitivních pozorování},\\ TN &= \text{počet pravě negativních pozorování},\\ P = FN + TP &= \text{počet pozitivních pozorování},\\ N = FP + TN &= \text{počet negativních pozorování}. \end{aligned}$$

Na první pohled bychom mohli posoudit schopnost klasifikátoru detekovat anomálie pomocí poměru celkového počtu chybných klasifikací ku celkovému počtu pozorování. Tento poměr je nazýván jako **přesnost** a je dán vztahem

$$\text{přesnost} = \frac{TP + TN}{FP + FN + TP + TN}.$$

Tato hodnota však může být zavádějící. Mějme 95 normálních pozorování a 5 anomálních pozorování. Pokud bychom zavedli triviální klasifikátor, který by všechna pozorování klasifikoval hodnotou -1, pak by celková přesnost byla 95%. To se na první pohled jeví jako dobrá hodnota i přes to, že klasifikátor v detekci anomálií zcela selhal.

Z tohoto důvodu zavádíme jiné poměry uvedených počtů. Prvním poměrem je tzv. **senzitivita**, neboli míra pravé pozitivity (anglicky *true positive rate* - zkratka TPR), která je dána poměrem

$$TPR = \frac{TP}{P} = \frac{TP}{TP + FN}.$$

V případě detekce anomálií udává senzitivita poměr korektně klasifikovaných anomálií. Obdobným způsobem můžeme zavést tzv. **selektivitu**, neboli míru pravé negativity (anglicky *true negative rate* - zkratka TNR), která je dána poměrem

$$TNR = \frac{TN}{N} = \frac{TN}{TN + FP}.$$

Jako protiklad senzitivity můžeme zavést míru falešné pozitivity (anglicky *false positive rate* - zkratka FPR), která je dána poměrem

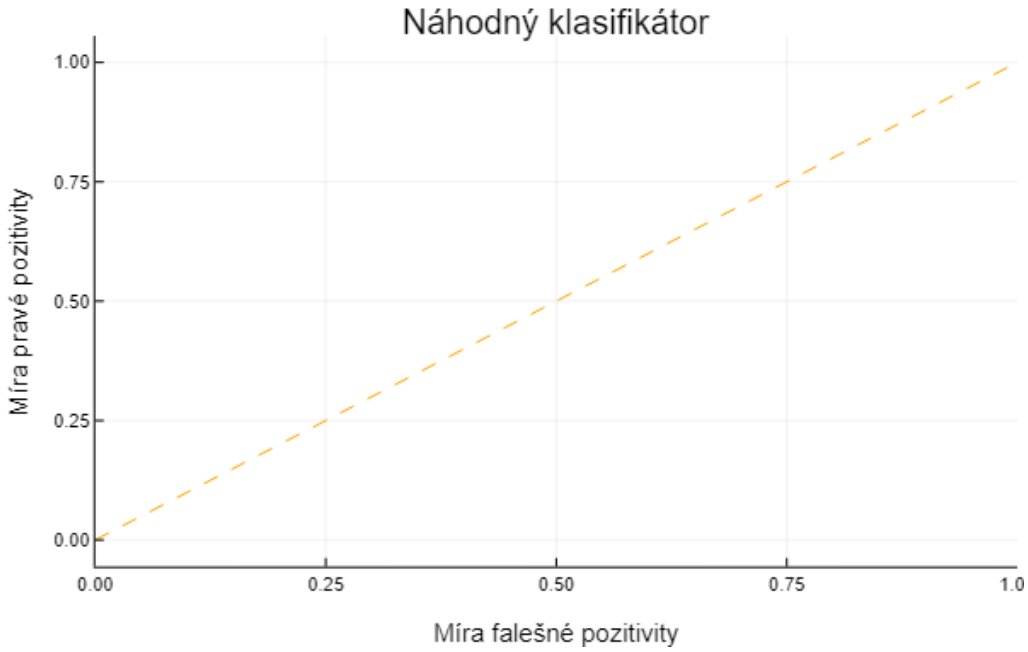
$$FPR = \frac{FP}{N} = \frac{FP}{FP + TN}.$$

V případě detekce anomálií nám míra falešné pozitivity udává poměr nesprávně klasifikovaných anomálií - jde tedy o indikátor chyby prvního typu. Obdobně můžeme opět zavést míru falešné negativity (anglicky *false negative rate* - zkratka FNR) poměrem

$$FNR = \frac{FN}{P} = \frac{FN}{FN + TP},$$

která je indikátorem chyby druhého druhu.

Nyní pro budoucí evaluaci zavedeme graf závislosti TPR na FPR. Tento graf bude dle definic použitých poměrů ležet ve čtverci $[0, 1] \times [0, 1]$. Mějme nyní triviální klasifikátor anomálií, který bude zcela náhodně rozhodovat, zda dané pozorování je, či není anomálií. Rozhodnutí tohoto klasifikátoru bude závist na Bernoulliho rozdělení s parametrem $p = 0,5$. Při klasifikaci *si tedy hodíme mincí* a na základě toho, co nám padne rozhodneme, zda-li pozorování označíme jako pozitivní, nebo negativní. Míra pravé pozitivity bude pravděpodobně právě $0,5$ a zároveň míra falešné pozitivity bude také pravděpodobně $0,5$ pro dostatečně velký soubor testovaných dat. Hodnota tohoto klasifikátoru bude ležet na zavedeném grafu v bodě $FPR = 0,5$ a $TPR = 0,5$. Tato hodnota leží na křivce identické funkce $f(x) = x$. Pokud budeme spojitě hýbat s parametrem $p \in (0, 1)$ užitého Bernoulliho rozdělení, dostaneme na



Obrázek 3.1: Příklad grafu závislosti míry pravé pozitivity na míře pravé negativity v závislosti na parametru $p \in (0, 1)$ Bernoulliho rozdělení užitého náhodného klasifikátoru anomálií.

zavedeném grafu opět hodnoty, které budou ležet na křivce identické funkce. Tímto postupem dostaneme graf na obrázku 3.1.

Vraťme se nyní k modelům odhadujícím hustotu pravděpodobnosti. Do této chvíle jsme jsme pracovali s tzv. prahem normality τ_α, resp. s jeho empirickým odhadem $\hat{\tau}_\alpha$. Tento práh nám přímo definoval množinu anomálií, jinými slovy přímo určoval to, jaká pozorování v našem hypotetickém prostoru $(\mathcal{X}, p_x(\boldsymbol{x}))$ označit jako anomální, resp. normální. Teď však budeme na tento práh nahlížet z opačného úhlu pohledu. Mějme testovací soubor dat $(\boldsymbol{x}_1, -1), \ldots, (\boldsymbol{x}_n, -1), (\boldsymbol{x}_{n+1}, +1), \ldots, (\boldsymbol{x}_{s+n}, +1) \in \mathcal{X} \times \{\pm 1\}$, tedy soubor normálních a anomálních dat. Mějme dále již optimalizovaný model odhadující neznámou p_x hustotu pravděpodobnosti pomocí hustoty $p_x^{\theta_0}$. Zaveďme hodnoty

$$\tau_{max} = \max_{i=1,\ldots,n+s} p_x^{\theta_0}(\boldsymbol{x}_i),$$
$$\tau_{min} = \min_{i=1,\ldots,n+s} p_x^{\theta_0}(\boldsymbol{x}_i).$$

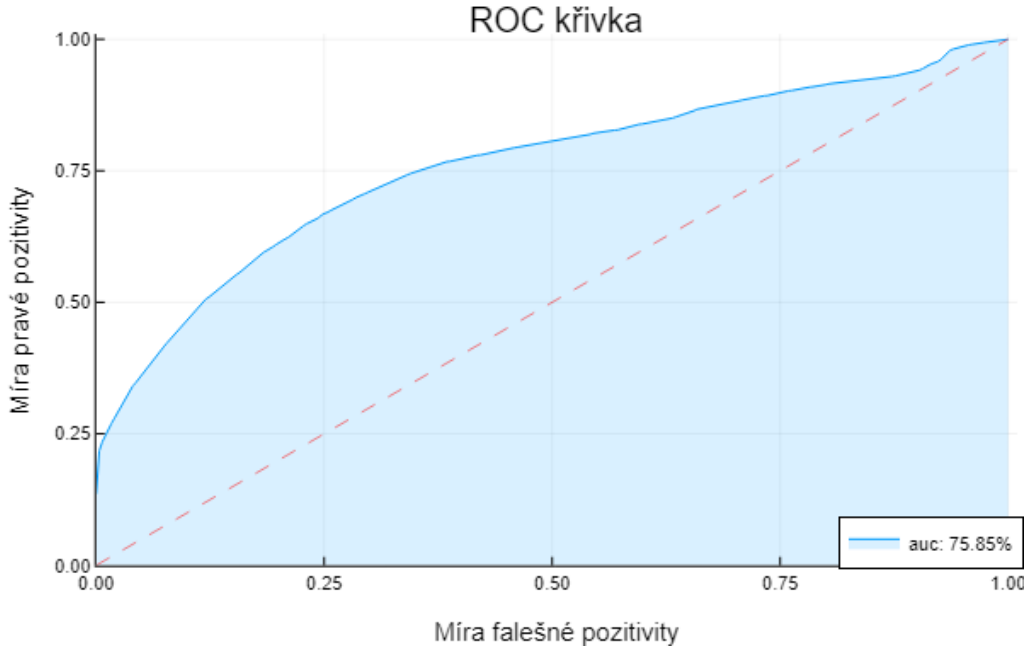
Předpokládejme, že $\tau_{max} \neq \tau_{min}$. Pak pro $\tau \in (\tau_{max}, \tau_{min})$ zavedeme klasifikátor tvaru

$$c_\tau(\boldsymbol{x}) = \begin{cases} +1 & \text{pro } p_x^{\theta_0}(\boldsymbol{x}) \leq \tau, \\ -1 & \text{pro } p_x^{\theta_0}(\boldsymbol{x}) > \tau, \end{cases}$$

a dále na testovacích datech určíme poměry TPR a FPR a jejich hodnoty vyneseme na sestrojený graf. Stejný postup provedeme pro všechna $\tau \in (\tau_{max}, \tau_{min})$ čímž obdržíme tzv. **ROC křivku** (z anglického *Receiver operating characteristic*).

Poznámka. Ve skutečnosti samozřejmě nemusíme tento postup provozovat pro nespočetně mnoho $\tau \in (\tau_{max}, \tau_{min})$. Z tvaru zavedeného klasifikátoru plyne, že výsledná ROC křivka bude schodovitá funkce. Musíme tedy určit klasifikátory s $n+s$ různými prahy ($n+s$ je celkový počet testovacích dat).

Pro rostoucí počet dat $n+s \to \infty$ a pro $\tau \to \tau_{max}$ bude platit $FPR \to 1$ a $TPR \to 1$. Naopak pro $\tau \to \tau_{min}$ bude platit $FPR \to 0$ a $TPR \to 0$. Výsledná ROC křivka tedy bude začínat v bodě $(0,0)$ a končit v bodě $(1,1)$. ROC křivka dobře navrženého klasifikátoru anomálií by tedy měla ležet nad křivkou identity. Naopak pokud by ROC křivka klasifikátoru ležela pod křivkou identity, znamenalo by to, že klasifikátor se chová hůře, než náhodný klasifikátor - v tomto případě by byla detekce anomálií spíše kontraproduktivní. Ukázku ROC křivky modelu SPTN můžeme např. vidět na obrázku 3.2.



Obrázek 3.2: Ukázka ROC křivky klasifikátoru sestrojeného pomocí SPTN. Hodnota AUC udává obsah plochy pod ROC křivkou.

Poznámka. Postup sestrojení ROC křivky jsme popsali pouze pro modely odhadující hustotu pravděpodobnosti. Stejný postup však můžeme zopakovat i pro jiné metody strojového učení pro detekci anomálií. Např. klasifikátor jedné třídy popsaný v sekci 1.2 bude záviset na předem zvoleném parametru $\alpha \in (0,1)$. Stejným postupem bychom tedy určili hodnoty FPR a TPR pro všechna $\alpha \in (0,1)$ a tím získali opět hledanou ROC křivku pro tuto třídu modelů.

ROC křivka tedy udává celkovou schopnost modelu detekovat anomálie pro všechny prahy normality. Tuto schopnost můžeme shrnout hodnotou obsahu plochy pod ROC křivkou na tomto grafu, kterou označíme jako AUC (z anglického *Area under curve*, hodnotu udáváme často v procentech). Při $AUC = 100\%$ jsme tedy nalezli perfektního klasifikátor anomálií daného problému.

V praxi nám však může být např. zadána situace, kdy chceme nalézt co nejlepší klasifikátor, po kterém požadujeme, aby $FPR \leq 5\%$ pro testovaná data. V tomto případě nám při fixním $FPR = 5\%$ zajímá pouze hodnota TPR, tedy hodnota ROC křivky jako funkce v bodě $0,05$. Mějme dva klasifikátory s vlastnostmi:

1. První klasifikátor má $AUC = 90\%$ a pro $FPR = 5\%$ je $TNR = 70\%$.

2. Druhý klasifikátor má $AUC = 80\%$ a pro $FPR = 5\%$ je $TNR = 80\%$.

Při zmíněném zadání bychom vybrali raději druhý klasifikátor i přes to, že první klasifikátor je díky vyšší hodnotě AUC flexibilnější. Touto situací se budeme zabývat v nadcházející kapitole této práce. Předtím si však v poslední sekci této kapitoly shrneme celkový postup trénování a evaluace detektoru anomálií založeném na odhadu hustoty pravděpodobnosti.

3.3 Trénování a evaluace detektoru anomálií pomocí odhadu hustoty pravděpodobnosti

Tato sekce bude sloužit jako shrnutí kapitoly 2 a dosavadních poznatků této kapitoly. Do této chvíle jsme také poměrně zanedbávali rozdělení dat na trénovací, validační a testovací soubor. V této části tedy vše zrekapitulujeme korektně v rámci metod strojového učení.

Mějme nezávislá pozorování $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in (\mathcal{X}, p_x(\boldsymbol{x}))$, kde $\mathcal{X} \subseteq \mathbb{R}^D$ a p_x je neznámá hustota pravděpodobnosti. Budeme uvažovat, že data nám byla poskytnuta v jedné ze dvou podob.

V prvním případě jsme data obdrželi bez apriorní informace o tom, která pozorování jsou a která nejsou anomální. Zvolíme si tedy kompaktní množinu $\boldsymbol{K} \subset \mathcal{X}$, která pokrývá všechna data $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$. Dle rovnoměrného rozdělení na množině $\boldsymbol{K}$ vygenerujeme umělé anomálie $\hat{\boldsymbol{x}}_1, \ldots, \hat{\boldsymbol{x}}_s$. Původní data označíme hodnotou -1 a umělé anomálie hodnotou $+1$. Výsledkem bude soubor

$$(\boldsymbol{x}_1, -1), \ldots, (\boldsymbol{x}_n, -1), (\hat{\boldsymbol{x}}_1, +1), \ldots, (\hat{\boldsymbol{x}}_s, +1) \in \mathcal{X} \times \{\pm 1\}.$$

V druhém případě již máme data předložena přímo s informací o tom, která pozorování jsou a která nejsou anomální. Máme tedy k dispozici soubor

$$(\boldsymbol{x}_1, -1), \ldots, (\boldsymbol{x}_m, -1), (\boldsymbol{x}_{m+1}, +1), \ldots, (\boldsymbol{x}_n, +1) \in \mathcal{X} \times \{\pm 1\},$$

kde $m < n$. Zároveň požadujeme, aby počet anomálních dat nebyl zanedbatelný (např. aby alespoň 5% celkového počtu dat bylo anomálních).

Dále tato data např. v poměrů $6 : 2 : 2$ náhodně rozdělíme na trénovací, validační a testovací soubor. Po tomto rozdělení požadujeme, aby se v trénovacím souboru nenacházeli žádné anomálie! Tento požadavek nám zaručí, že aproximace hledané hustoty pravděpodobnosti bude co *nejčistší*.

Poznámka. V praxi pravděpodobně zcela čistý trénovací soubor mít nebudeme. V této práci tento předpoklad ovšem používat budeme, jelikož tím se při porovnání jednotlivých modelů zbavíme neznámé veličiny (míra zašpiněný dat), se kterou bychom museli v evaluaci počítat.

Nyní zvolíme třídu modelů pro odhad hustoty pravděpodobnosti a provedeme následující postup.

1. Vytvoříme několik modelů s odlišnými strukturami a parametry (v metodě FFJORD zvolíme jiné tvary užitých neuronových sítí, v metodě SPTN vytvoříme náhodně několik odlišných struktur samotné sítě).

   Poznámka. Různé parametry a struktury modelů stejného typu nazýváme také jako **hyper-parametry**.

2. Každý model poté optimalizujeme vůči datům z trénovacího souboru, čímž získáme hledané odhady hustot pravděpodobnosti. Na každém modelu poté určíme tvar ROC křivky na validačních datech a vypočteme její hodnotu AUC na těchto datech.

   Poznámka. Ideálně pro každý model s odlišnými hyper-parametry vygenerujeme na začátku vlastní soubory trénovacích, validačních a testovacích dat.

3. Poté z modelů této třídy s odlišnými hyper-parametry vybereme ten s nejvyšší hodnotou AUC na validačních datech.

4. V poslední řadě na tomto vybraném modelu sestrojíme ROC křivku na testovacích datech a opět určíme hodnotu AUC na testovacích datech. Tato hodnota vypovídá o schopnosti modelu detekovat anomálie v praxi. Stejný postup můžeme zrekapitulovat pro jinou třídu modelů pro odhad hustoty pravděpodobnosti, resp. pro libovolnou třídu modelů pro detekci anomálií. Jednotlivé třídy modelů pak můžeme srovnat pomocí hodnoty AUC a testovacích datech.

5. Při zadání s fixním požadavkem na FPR můžeme modely srovnat v rámci hodnoty TPR na testovacích datech.

Nyní se přesuneme k poslední kapitole teoretické části této práce, kde si popíšeme modifikaci ztrátové funkce při trénování modelů odhadujících hustoty pravděpodobnosti pro detekci anomálií.

Kapitola 4

Modifikace ztrátové funkce modelů pro odhad hustoty pravděpodobnosti

V této kapitole se přesuneme k hlavnímu tématu této práce. V kapitole 1 jsme zmínili pojem předpokladu koncentrace dat. Jak za malou chvíli uvidíme, odhadem hustoty pravděpodobnosti pomocí maximálně věrohodného odhadu jsme schopni empiricky nalézt množinu s malým objemem, která bude obsahovat většinu dat. Tuto množinu ovšem určíme až po dokončení učení našeho modelu. Naše idea spočívá v tom, že využijeme schopnosti modelů pro odhad hustoty pravděpodobnosti vyčíslit věrohodnost jednotlivých dat při samotném trénování. Díky této skutečnosti modifikujeme tvar ztrátové funkce, čímž zahrneme minimalizaci zmíněného objemu do samotného učení.

4.1 Minimalizace objemu pomocí hustoty pravděpodobnosti

Nejprve se však pro názornost vraťme ke klasifikátorům jedné třídy ze sekce 1.2. V této sekci jsme si pouze popsali postup klasifikace jedné třídy pomocí metody podpůrných vektorů. Pro budoucí účely si naznačme i teoretický základ této metody (teorie vychází z článku [18]). Z předpokladu koncentrace dat tato metoda hledá množinu

$$C_\alpha = \operatorname*{arginf}_{C^\tau} \{\mu(C^\tau),\ \mathbb{P}(C^\tau) \geq 1 - \alpha\},$$

kde

$$C^\tau = \{\boldsymbol{x} \in \mathcal{X},\ p_x(\boldsymbol{x}) > \tau\},$$

$C_\alpha, C^\tau \subset \mathcal{X} \subseteq \mathbb{R}^D$, $\alpha \in (0, 1)$, a kde $\mathbb{P}$ je příslušná pravděpodobnostní míra prostoru $\mathcal{X}$. Jinými slovy metoda postupně minimalizuje hodnotu $\mu(C^\tau)$ za podmínky $\mathbb{P}(C^\tau) \geq 1-\alpha$. Hodnotu $\mathbb{P}(C^\tau)$ však v praxi obecně neznáme - proto využijeme data $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in (\mathcal{X}, p_x(\boldsymbol{x}))$, kde p_x je opět hustota pravděpodobnosti příslušná míře $\mathbb{P}$.

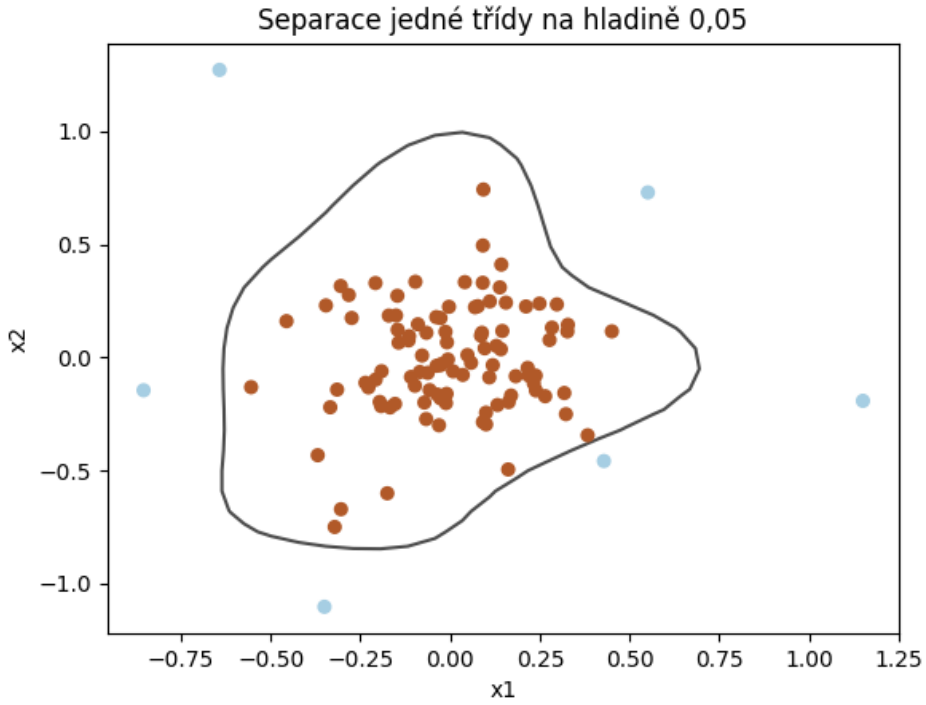
Pomocí těchto dat můžeme zavést empirický odhad míry $\mathbb{P}(C^\tau)$ tvaru

$$\mathbb{P}_n(C^\tau) := \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{C^\tau}(\boldsymbol{x}_i).$$

Metoda klasifikace jedné třídy pomocí metody podpůrných vektorů tedy v praxi nalezne množinu

$$\hat{C}_\alpha = \operatorname*{arginf}_{C^\tau}\{\mu(C^\tau),\ \mathbb{P}_n(C^\tau) \geq 1-\alpha\}.$$

Tuto situaci můžeme vidět opět na obrázku 4.1.



Obrázek 4.1: Příklad minimalizace objemu na základě empirického odhadu pravděpodobnostní míry pomocí dvourozměrných dat na hladině $\alpha = 0,05$.

Vidíme, že tvar nalezeného objemu (objem, resp. zde obsah, ohraničený vyobrazenou křivkou) závisí přímo na poloze poskytnutých dat.

Vraťme se nyní k modelům odhadujícím hustotu pravděpodobnosti. Nechť $p_x^{\theta_0}$ je nalezený odhad hustoty p_x vůči datům $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in (\mathcal{X}, p_x(\boldsymbol{x}))$. Mějme empirický odhad prahu normality na hladině α tvaru

$$\hat{\tau}_\alpha = \sup_{\tau}\left\{\tau \geq 0,\ \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{[0,p_x^{\theta_0}(\boldsymbol{x}_i))}(\tau) \geq 1-\alpha\right\}.$$

Ukažme si, že se opravdu jedná o vhodný odhad prahu normality. Nechť $\hat{\tau}$ je libovolný práh normality splňující podmínku

$$\frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{[0,p_x^{\theta_0}(\boldsymbol{x}_i))}(\hat{\tau}) \geq 1-\alpha.$$

Pak pro příslušnou množinu $C^{\hat{\tau}} = \{\boldsymbol{x} \in \mathcal{X},\ p_x(\boldsymbol{x}) > \hat{\tau}\}$ platí

$$\mathbb{P}(C^{\hat{\tau}}) = \int_{C^{\hat{\tau}}} 1 \cdot d\mathbb{P}(\boldsymbol{x}) = \int_{C^{\hat{\tau}}} p_x(\boldsymbol{x}) d\boldsymbol{x} \approx \int_{C^{\hat{\tau}}} p_x^{\theta_0}(\boldsymbol{x}) d\boldsymbol{x} \approx \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{[0, p_x^{\theta_0}(\boldsymbol{x}_i))}(\hat{\tau}) \geq 1 - \alpha.$$

Takto zvolené prahy $\hat{\tau}$ tedy pravděpodobně splňují nutnou podmínky pro nalezení množiny C_α. Zároveň z definice suprema platí

$$\mu(C^{\hat{\tau}_\alpha}) = \int_{C^{\hat{\tau}_\alpha}} 1 \cdot d\boldsymbol{x} \leq \int_{C^{\hat{\tau}}} 1 \cdot d\boldsymbol{x} = \mu(C^{\hat{\tau}}),$$

jelikož

$$C^{\hat{\tau}_\alpha} \approx \left\{\boldsymbol{x} \in \mathcal{X},\ p_x^{\theta_0}(\boldsymbol{x}) > \hat{\tau}_\alpha\right\} \subset \left\{\boldsymbol{x} \in \mathcal{X},\ p_x^{\theta_0}(\boldsymbol{x}) > \hat{\tau}\right\} \approx C^{\hat{\tau}}.$$

Z těchto vztahů vidíme, že pomocí empirického odhadu $\hat{\tau}_\alpha$ můžeme pomocí modelů pro odhad hustoty pravděpodobnosti nalézt empirický odhad množiny hladiny normality α tvaru

$$\hat{C}_\alpha = \left\{\boldsymbol{x} \in \mathcal{X},\ p_x^{\theta_0}(\boldsymbol{x}) > \hat{\tau}_\alpha\right\}.$$

Poznámka. Tímto postupem jsme tedy dokázali převést problém odhadu hustoty na problém nalezení nejmenšího objemu s vysokou koncentrací dat. Za jistých podmínek lze zkonstruovat i opačný postup - tedy pomocí metod pro nalezení nejmenšího objemu můžeme zkonstruovat odhad hustoty pravděpodobnosti, viz [12]. Za jistých podmínek můžeme tedy tyto postupy považovat za ekvivalentní.

4.2 Ztrátová funkce na kvantilovém intervalu

Jak již bylo zmíněno na počátku této kapitoly, množinu $\hat{C}_\alpha$ určíme až po nalezení odhadu hustoty $p_x^{\theta_0}$, který jsme určili pomocí metody maximálně věrohodného odhadu. Naše modifikace ztrátové funkce spočívá v tom, že empirický odhad prahu normality budeme určovat již během učení v rámci aktuálního odhadu hustoty pravděpodobnosti. Po určení tohoto prahu vybereme pro následnou optimalizaci pouze data, jejichž aktuální věrohodnost leží nad tímto prahem. Tento postup si rozepíšeme podrobněji.

Mějme trénovací soubor nezávislých pozorování $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in (\mathcal{X}, p_x(\boldsymbol{x}))$ s neznámou hustotou pravděpodobnosti p_x. Mějme model pro odhad hustoty pravděpodobnosti s počátečními parametry $\theta_1 \in \Theta$. Při užití např. metody největšího spádu bychom zavedli ztrátovou funkci

$$\mathcal{L}(\theta | \boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = -\prod_{i=1}^{n} p_x^{\theta}(\boldsymbol{x}_i).$$

Dále bychom pomocí diferenciačních algoritmů nalezli gradient

$$\nabla_\theta \mathcal{L}(\theta_1 | \boldsymbol{x}_1, \ldots, \boldsymbol{x}_n),$$

který bychom dále spolu s optimalizačními algoritmy využili pro nalezení nových parametrů $\theta_2 \in \Theta$. Tímto postupem bychom iterativně postupovali dokud bychom v ideálním případě nenalezli globálním minimum této ztrátové funkce s parametry $\theta_0 \in \Theta$.

Tento postup nyní upravíme. Určíme si hladinu normality $\alpha \in (0,1)$. Nejprve určíme empirický odhad prahu normality $\hat{\tau}_\alpha^{\theta_1} \geq 0$ vůči aktuálnímu odhadu hustoty pravděpodobnosti $p_x^{\theta_1}$, tedy hodnotu

$$\hat{\tau}_\alpha^{\theta_1} = \sup_\tau \left\{ \tau \geq 0,\ \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{[0, p_x^{\theta_1}(\boldsymbol{x}_i))}(\tau) \geq 1 - \alpha \right\}.$$

Podívejme se, jak naimplementovat nalezení této hodnoty. Nechť zobrazení $(\cdot) : \{1, \ldots, n\} \to \{1, \ldots, n\}$ je permutace, která pro data $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ splňuje

$$p_x^{\theta_1}(\boldsymbol{x}_{(1)}) \leq p_x^{\theta_1}(\boldsymbol{x}_{(2)}) \leq \cdots \leq p_x^{\theta_1}(\boldsymbol{x}_{(n)}).$$

Dále určíme index $m := \lceil \alpha \cdot n \rceil$, kde výraz $\lceil \cdot \rceil$ značí horní celou část. Práh normality pak určíme jako

$$\hat{\tau}_\alpha^{\theta_1} = p_x^{\theta_1}(\boldsymbol{x}_{(m)}).$$

Dále pomocí takto zvolené permutace zadefinujeme uspořádaní na trénovacích datech vůči hustotě $p_x^{\theta_1}$ jako

$$\boldsymbol{x}_{(1)} \leq \cdots \leq \boldsymbol{x}_{(n)} \iff p_x^{\theta_1}(\boldsymbol{x}_{(1)}) \leq \cdots \leq p_x^{\theta_1}(\boldsymbol{x}_{(n)}).$$

Ztrátovou funkci nyní zavedeme pomocí maximálně věrohodného odhadu pouze vůči datům, jejichž hodnota aktuální hustoty pravděpodobnosti leží nad zvoleným prahem normality, tedy ve tvaru

$$\mathcal{L}(\theta | \boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) := - \prod_{\substack{i=1 \\ p_x^\theta(\boldsymbol{x}_{(i)}) \in (\tau_\alpha^{\theta_1}, \infty)}}^{n} p_x^\theta(\boldsymbol{x}_i) = - \prod_{i=m+1}^{n} p_x^\theta(\boldsymbol{x}_{(i)}). \tag{4.1}$$

Gradient na parametrech θ_1 tedy počítáme pouze vůči datům $\boldsymbol{x}_{(m+1)} \leq \cdots \leq \boldsymbol{x}_{(n)}$ a využijeme ho pro nalezení parametrů $\theta_2 \in \Theta$. Naší hypotézou je, že tato modifikace zapříčiní nalezení lepšího odhadu množiny C_α na konci trénování, jelikož model bude již při učení vylučovat data, která se jeví jako anomální.

Samozřejmě můžeme opět využít toho, že logaritmická funkce je ostře rostoucí, a postup zreplikovat v logaritmickém tvaru. Logaritmus hustoty pravděpodobnosti nám zachová zvolené uspořádání

$$\boldsymbol{x}_{(1)} \leq \cdots \leq \boldsymbol{x}_{(n)} \iff \log p_x^\theta(\boldsymbol{x}_{(1)}) \leq \cdots \leq \log p_x^\theta(\boldsymbol{x}_{(n)}).$$

Pro $m := \lceil \alpha \cdot n \rceil$ můžeme určit logaritmický odhad prahu normality

$$\log \hat{\tau}_\alpha^\theta = \log p_x^\theta(\boldsymbol{x}_{(m)})$$

a zavedeme normovanou ztrátovou funkci tvaru

$$\mathcal{L}(\theta | \boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = -\frac{1}{n-m} \sum_{i=m+1}^{n} \log p_x^\theta(\boldsymbol{x}_{(i)}). \tag{4.2}$$

Nyní si pro budoucí účely zadefinujeme námi zavedený pojem tzv. **kvantilového intervalu**

Definice 15 (Kvantilový interval)**.** Mějme hodnoty $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathcal{X} \subseteq \mathbb{R}^D$ a hustotu pravděpodobnosti p_x na prostoru $\mathcal{X}$. Dále mějme uspořádání na $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ dáno vztahem

$$\boldsymbol{x}_{(1)} \leq \cdots \leq \boldsymbol{x}_{(n)} \iff p_x(\boldsymbol{x}_{(1)}) \leq \cdots \leq p_x(\boldsymbol{x}_{(n)}),$$

kde zobrazení $(\cdot) : \{1, \ldots, n\} \to \{1, \ldots, n\}$ značí příslušnou permutaci. Nechť $I \subset [0, 1]$ je interval. Řekneme, že hodnota $\boldsymbol{x}_i,\ i \in \{1, \ldots, n\}$, spadá do **kvantilového intervalu** I vůči hustotě p_x právě když

$$\frac{(i)}{n} \in I.$$

Pomocí pojmů z této definice můžeme říct, že popsaný postup modifikace ztrátové funkce je ekvivalentní s trénováním ztrátové funkce na datech z kvantilového intervalu $(\alpha, 1]$.

Poznámka. Ve skutečnosti se jedná o kvantilový interval $(\alpha + \epsilon, 1]$ (kde $\epsilon > 0$ je vhodně zvoleno) jelikož pravděpodobně bude platit, že $\boldsymbol{x}_{(m)} \in (\alpha, 1]$. Tuto hodnotu jsme však ve výše popsaném postupu při učení nepoužili. Pro vysoká n však můžeme tuto skutečnost zanedbat, proto bych poprosil čtenáře o odpuštění za tuto chybu.

Dále pomocí této definice můžeme zavést trénování i na jiných kvantilových intervalech. Uvažujme např. interval $[0,05;\ 0,1]$. Mějme opět odhad hustoty p_x^θ a vůči této hustotě seřazená data $\boldsymbol{x}_{(1)} \leq \cdots \leq \boldsymbol{x}_{(n)}$. Určíme indexy

$$\underline{m} := \lceil 0,05 \cdot n \rceil,\ \overline{m} := \lceil 0,1 \cdot n \rceil.$$

Ztrátovou funkci pak zavedeme ve tvaru

$$\mathcal{L}(\theta | \boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = -\frac{1}{\overline{m} - \underline{m} + 1} \sum_{i=\underline{m}}^{\overline{m}} \log p_x^\theta(\boldsymbol{x}_{(i)}).$$

Tato modifikace zapříčiní, že se opět snažíme nalézt odhad množiny C_α, ale již nedbáme na správnost odhadu hustoty pravděpodobnosti pro 90% nejvěrohodnějších dat. Modely strojového učení mají obecně v závislosti na jejich počtu parametrů stanovenou kapacitu toho, co jsou schopny se naučit. Tím, že z učení vyloučíme povinnost modelu dbát na exaktní naučení se hustoty pravděpodobnosti, úlohu zjednodušíme. Model poté může věnovat větší část své kapacity na nalezení objemu s nejvyšší koncentrací dat. Příklad této skutečnosti můžeme vidět později ve výpočetní studii.

Poznámka. V tomto výkladu jsme opět vztahovali ztrátovou funkci k celému trénovacímu souboru. Samozřejmě tyto postupy můžeme ekvivalentně zreplikovat např. pro stochastickou verzi metody největšího spádu, kdy trénování vztahujeme pouze k menším náhodně vybraným podsouborům dat. V této modifikaci již však musíme dbát na to, že počet nenulových složek gradientů v závislosti na délce kvantilového intervalu prudce klesá. V případě intervalu $[0,05;\ 0,1]$ pracujeme pouze v počtem pěti procent nenulových složek gradientů oproti klasickému maximálně věrohodnému odhadu. Je proto třeba zvýšit velikost jednotlivých podsouboru a maximálních iterací.

V experimentální části této práce se budeme zabývat právě porovnáním jednotlivých výsledků s různými kvantilovými intervaly. Naší původní motivací byla domněnka, že např. při volbě $FPR = 0,05$ a kvantilového intervalu $[0,05;\ 0,1]$ bude hodnota TPR vyšší, než při užití klasického maximálně věrohodného odhadu, jelikož řešená úloha je jednodušší.

Kapitola 5

Výpočetní studie

Ve výpočetní studii této práci si nejprve ukážeme výsledek modifikace ztrátové funkce na uměle vygenerovaných ukázkových datech. Konkrétně se bude jednat o porovnání metody FFJORD a SPTN na různých kvantilových intervalech. Zvolená ukázková data budou dvourozměrná, což nám umožní vizualizovat výsledné odhady hustot pravděpodobností.

V druhé části studie porovnáme více modelů na různých kvantilových intervalech na reálných datech. Toto porovnání bude více sofistikované, než v případě ukázkových dat.

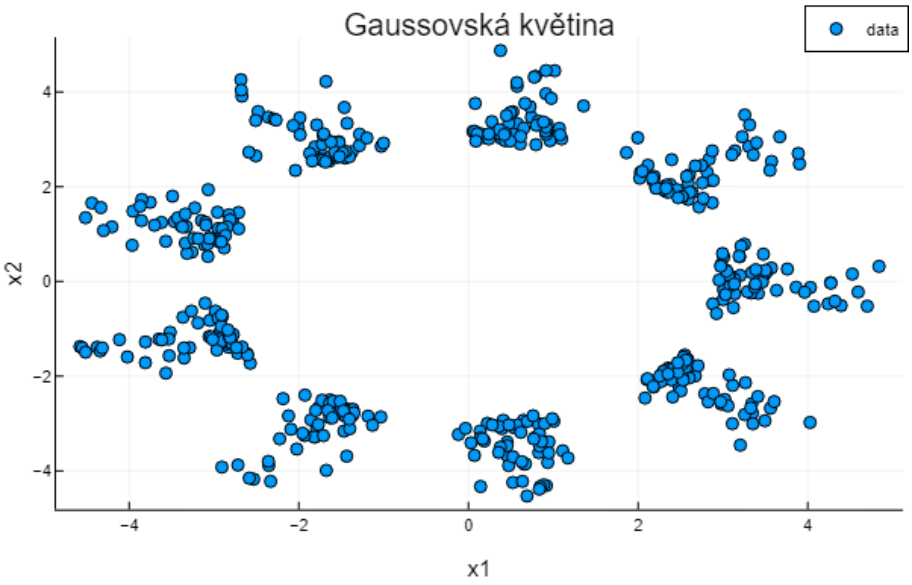
5.1 Vliv modifikace ztrátové funkce na ukázkových datech

V této části budeme prezentovat vlastnosti modifikované ztrátové funkce na souboru dat, který jsme nazvali gaussovská květina. Příklad tohoto souboru můžeme vidět na obrázku 5.1.

Tento soubor vznikl generací dat z devíti Gaussových rozdělení se stejným rozptylem a různými středními hodnotami (jednotlivé střední hodnoty leží na jednom kruhu). Dále byla data přetransformována pomocí goniometrických funkcí (a funkce tanh) do této podoby. Jak na obrázku 5.1 můžeme vidět, datový soubor dat má připomínat tvar květiny.

Tento soubor dat je pro prvotní testování modelů odhadujících hustotu pravděpodobnosti vhodný ze dvou důvodů. Za prvé jsme schopni otestovat schopnost modelu odhadnout hustotu složenou z více distribucí. Za druhé tento soubor dat vznikl použitím nelineárních transformací (zmíněných goniometrických funkcí). Pomocí těchto dat tedy můžeme posoudit schopnost jednotlivých metod modelovat nelineární transformace.

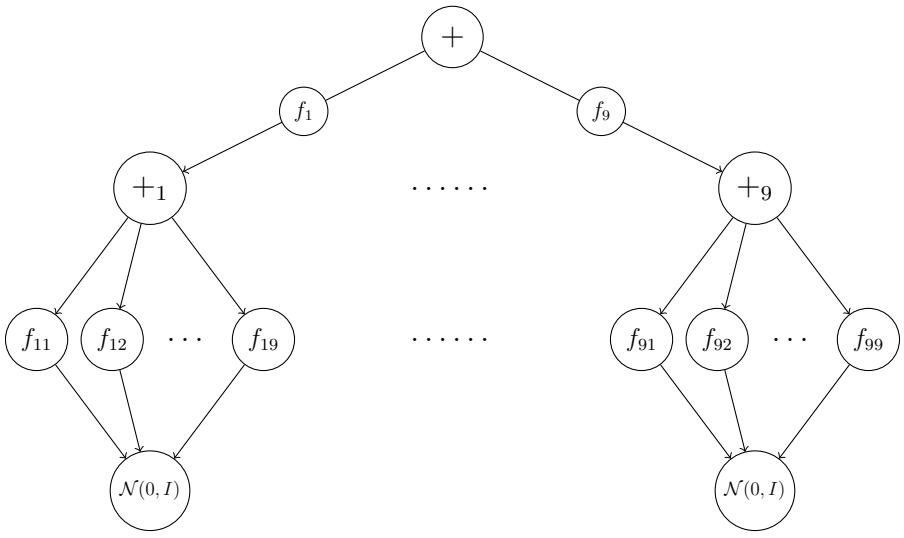
Tímto způsobem vygenerujeme trénovací soubor o velikosti 100 000, validační soubor o velikosti 1 000 a testovací soubor velikosti 10 000. Dále určíme kompaktní množinu $\boldsymbol{K} = [-5, 5] \times [-5, 5]$. S vysokou pravděpodobností bude tato množina



Obrázek 5.1: Příklad souboru dat gaussovské květiny

obsahovat všechna vygenerovaná data. Dále pro budoucí evaluaci dle popsaného postupu v kapitole 3 vygenerujeme na množině $\boldsymbol{K}$ z rovnoměrného rozdělení 500 umělých anomálií do testovacího souboru.

Nejprve si zmíníme výsledky metody SPTN. Architekturu použité sítě můžeme vidět na obrázku 5.2.

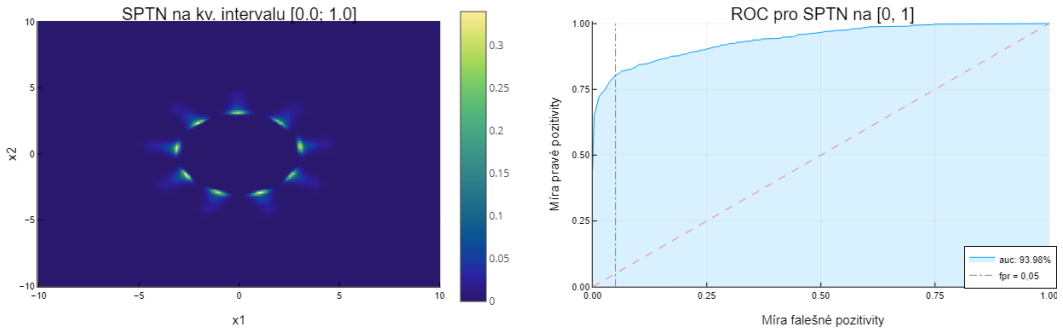


Obrázek 5.2: Architektura sítě SPTN užitá pro učení modelu na ukázkových datech.

Za zobrazení f_i, resp. f_{ij}, $i, j = 1, \ldots, 9$, byly zvoleny afinní transformace. Pro optimalizaci byla zvolena metoda ADAM [10] s velikostí kroku $\eta = 0,01$. Pro všechna trénování jsme použili 1 000 iterací.

Nejprve si ukážeme výsledek nalezené hustoty pravděpodobnosti užitím maximálně

věrohodného odhadu, což dle definice 15 v kapitole 4 odpovídá učení na kvantilovém intervalu $[0, 1]$. Pro učení jsme zvolili velikost trénovacích podsouborů 1000. Nalezenou hustotu pravděpodobnosti můžeme vidět na obrázku 5.3 spolu s grafem odpovídající ROC křivky na testovacím souboru s vygenerovanými anomáliemi. Vi-



(a) (b)

Obrázek 5.3: (a) Odhad hustoty pravděpodobnosti metodou SPTN na ukázkových datech. (b) Příslušná ROC křivka tohoto modelu.

díme, že příslušná ROC křivka leží vysoko nad křivkou identity s hodnotou plochy pod křivkou $AUC = 93,98\%$. Pro $FPR = 0,05$ (indikátor chyby prvního druhu) vychází $TPR = 0,8$ (indikátor chyby druhého druhu).

Nyní přejdeme k trénování na kvantilových intervalech $[0;\ 0,05]$, $[0,025;\ 0,075]$, $[0,05;\ 0,1]$, $[0,05;\ 1]$. Jak již bylo zmíněno v minulé kapitole 4, naší počáteční hypotézou bylo, že volbou kvantilových intervalů odpovídajících hladině normality $\alpha = 0,05$ získáme pro odpovídající $FPR = \alpha = 0,05$ vyšší hodnotu TPR. První interval $[0;\ 0,05]$ byl zvolen čistě experimentálně. Intervaly $[0,025;\ 0,075]$, $[0,05;\ 0,1]$ volíme za účelem zjištění, zda-li je vhodnější volit interval se středem v α, nebo s hodnotou α v levém okraji. Poslední interval $[0,05;\ 1]$ volíme přímo v souvislosti s výkladem v předešlé kapitole.

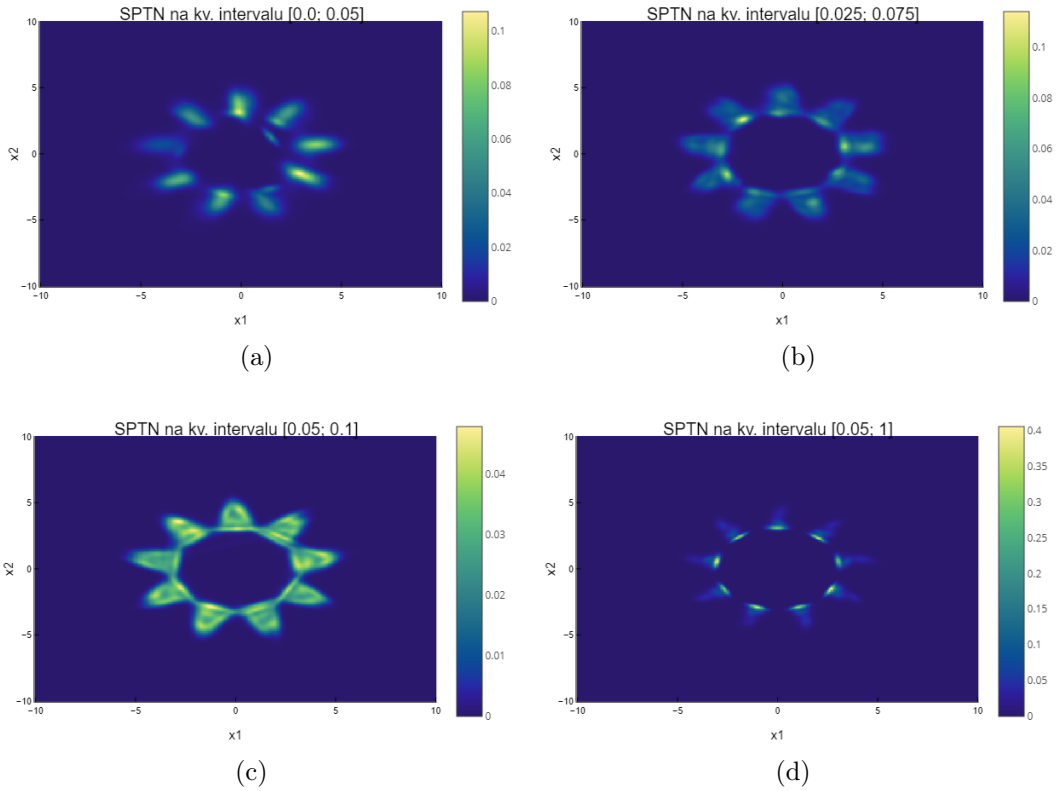
Pro první tři intervaly jsme zvolili velikosti trénovacích podsouborů 10 000 (kvůli menšímu počtu gradientů). Pro poslední interval $[0,05;\ 1]$ jsme zvolili opět velikost 1 000.

Výsledné hustoty pravděpodobností můžeme vidět na obrázku 5.4.

Pro první kvantilový interval (a) vyšla hustota nepřekvapivě hůře rozložená. Hustota není na listech květiny rovnoměrně rozložená a vznikly zde dvě defektní distribuce uvnitř kruhu.

Zajímavější výsledek jsme dostali v případě intervalů (b) a (c). Z obrázku vidíme, že tvar květiny zůstal zachován. Zároveň můžeme říci, že v obou případech metoda při učení dbala více na tvar květiny, než na korektní rozpoložení hustoty pravděpodobnosti v závislosti na datech. Tato situace je více patrná na obrázku (c), kdy tvar výsledné hustoty skutečné připomíná rozpoložení dat z obrázku 5.1.

V posledním případě intervalu (d) vidíme, že je tvar hustoty skoro identický s hustotou získanou pomocí maximálně věrohodného odhadu na obrázku 5.3. Z tohoto



(a) (b) (c) (d)

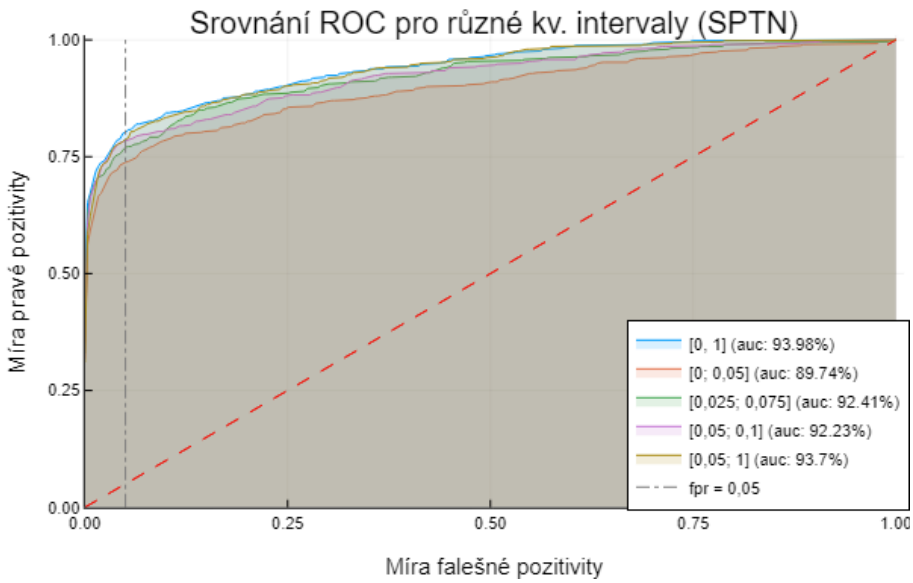
Obrázek 5.4: Výsledky hustot pravděpodobností získané trénováním metody SPTN na různých kvantilových intervalech: (a) $[0;\ 0,05]$, (b) $[0,025;\ 0,075]$, (c) $[0,05;\ 0,1]$, (d) $[0,05;\ 1]$.

důvodu tento případ považujeme za poměrně nezajímavý. Pokud bychom ovšem trénovali na trénovacím souboru, který již obsahuje nějaké anomálie, pravděpodobně bychom získali lepší výsledky než v případě užití maximálně věrohodného odhadu.

Vyobrazené hustoty pravděpodobnosti však slouží pouze k prvotní představě toho, jak se trénování modelu změní v závislosti na modifikaci ztrátové funkce. Nyní porovnáme jednotlivé případy pomocí ROC křivek na obrázku 5.5.

Z tohoto obrázku vidíme, že nejvýše položená křivka odpovídá maximálně věrohodnému odhadu. Zároveň pomocí šrafované přímky na hodnotě $FPR = 0,05$ vidíme, že i příslušná hodnota TPR vychází nejlépe pro maximálně věrohodný odhad. V tomto případě tedy naše hypotéza bohužel splněna nebyla. V tabulce 5.1 můžeme vidět porovnání hodnot AUC a TPR pro $FPR = 0,05$ jednotlivých variant kvantilových intervalů. Z této tabulky vidíme, že v obou srovnávaných hodnotách dominuje případ při použití maximálně věrohodného odhadu.

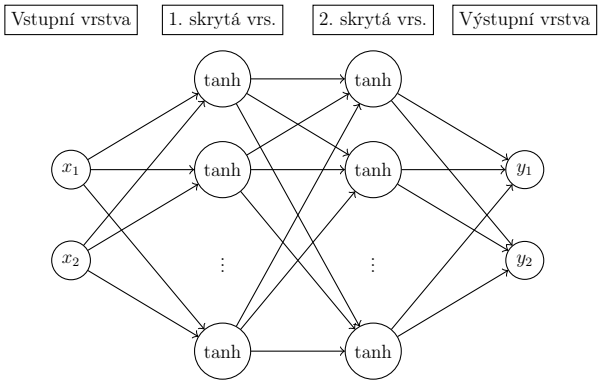
Analogický postup nyní zreplikujeme pro metodu FFJORD. Za zobrazení na pravé straně příslušné diferenciální rovnice volíme dopřednou neuronovou síť vyobrazenou na obrázku 5.6. Za časový interval volíme interval $(0, 1)$. Počet iterací a velikosti datových souborů jsou stejné jako v případě metody SPTN. Na obrázku 5.7 můžeme vidět výslednou hustotu pravděpodobnosti a příslušnou ROC křivku na testovacích datech pro trénování pomocí maximálně věrohodného odhadu, tedy pro



Obrázek 5.5: Srovnání ROC křivek metody SPTN na kvantilových intervalech $[0, 1]$, $[0;\ 0,05]$, $[0,025;\ 0,075]$, $[0,05;\ 0,1]$, $[0,05;\ 1]$.

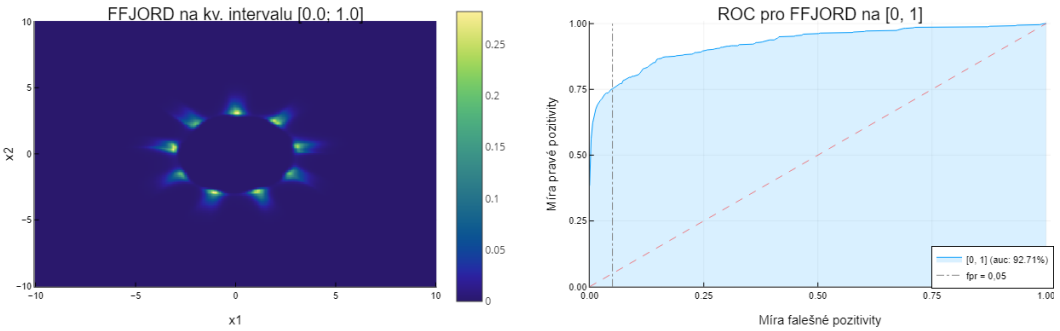
SPTN na kv. intervalu	AUC	TPR při $FPR = 0,05$
$[0, 1]$	**93,98**	**0,802**
$[0;\ 0,05]$	89,74	0,738
$[0,025;\ 0,075]$	92,41	0,77
$[0,05;\ 0,1]$	92,23	0,784
$[0,05;\ 1]$	93,7	0,788

Tabulka 5.1: Srovnání hodnot AUC a TPR při $FPR = 0,05$ modelů SPTN trénovaných na různých kvantilových intervalech.



Obrázek 5.6: Graf užité neuronové sítě v modelu FFJORD. Rozměr vstupní a výstupní vrstvy je 2, rozměry dvou skrytých vrstev jsou 20. Užité aktivační funkce jsou tanh.

trénování na kvantilovém intervalu $[0, 1]$. Opět určíme kvantilové intervaly $[0;\ 0,05]$, $[0,025;\ 0,075]$, $[0,05;\ 0,1]$, $[0,05;\ 1]$. Porovnání výsledných hustot pravděpodobností můžeme vidět na obrázku 5.8. Na tomto obrázku vidíme, že metoda FFJORD



(a) (b)

Obrázek 5.7: (a) Odhad hustoty pravděpodobnosti metodou FFJORD na ukázkových datech. (b) Příslušná ROC křivka tohoto modelu.

méně zachovává tvar květiny v porovnání s metodou SPTN. V případě (b) a (c) ovšem vidíme, že při učení bylo opět méně dbáno na hustotu rozpoložených dat a více na celkový tvar objemu obepínající květinu.

Dále na obrázku 5.9 máme srovnání ROC křivek pro jednotlivé kvantilové intervaly. Vidíme, že až na první kvantilový interval $[0;\ 0,05]$ jsou tvary ROC křivek srovnatelné.

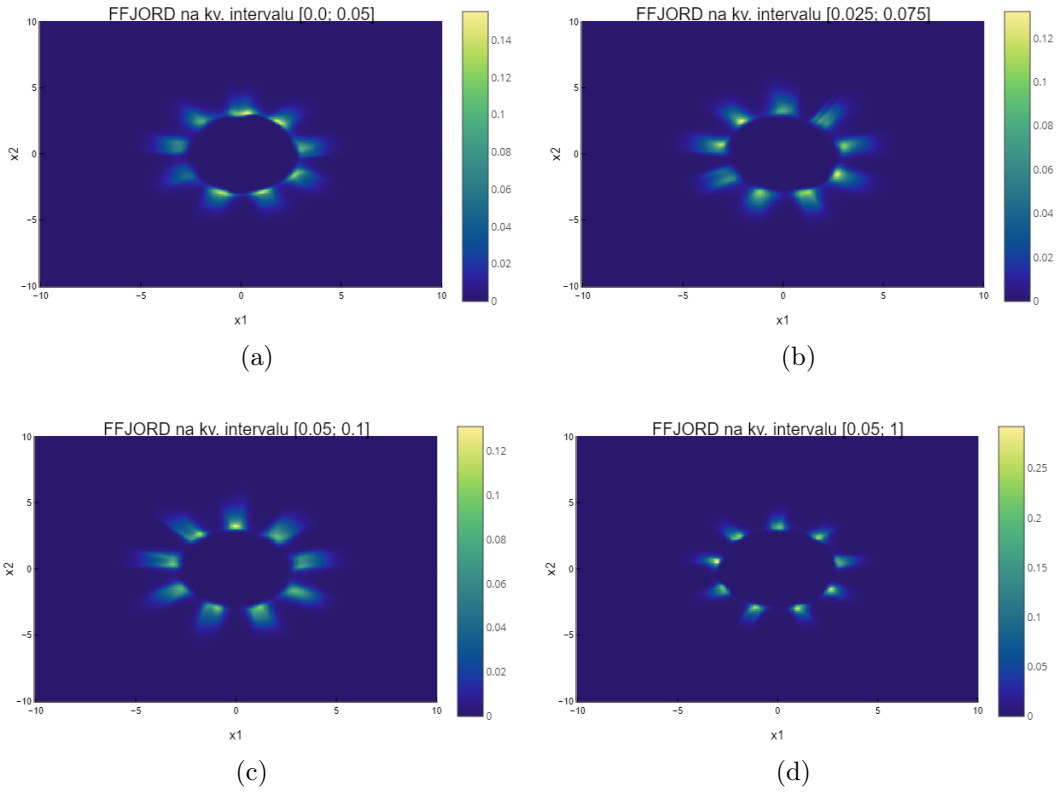
V poslední řadě opět uvedeme srovnání hodnot AUC a TPR metody FFJORD v tabulce 5.2. Z této tabulky vidíme, že nejlepší výsledky jsou získány při volbě in-

FFJORD na kv. intervalu	AUC	TPR při $FPR = 0,05$
$[0, 1]$	92,71	0,752
$[0;\ 0,05]$	91,81	0,7
$[0,025;\ 0,075]$	91,88	0,744
$[0,05;\ 0,1]$	91,87	**0,758**
$[0,05;\ 1]$	**92,86**	**0,758**

Tabulka 5.2: Srovnání hodnot AUC a TPR při $FPR = 0,05$ modelů FFJORD trénovaných na různých kvantilových intervalech.

tervalu $[0,05;\ 1]$. Stejnou hodnotu TPR nabývá i případ s intervalem $[0,05;\ 0,1]$. Tato zlepšení jsou však v porovnání s metodou užívající maximálně věrohodný odhad zanedbatelné - při trénování s jinými počátečními parametry bychom s vysokou pravděpodobností mohli dostat odlišné výsledky ve prospěch maximálně věrohodného odhadu.

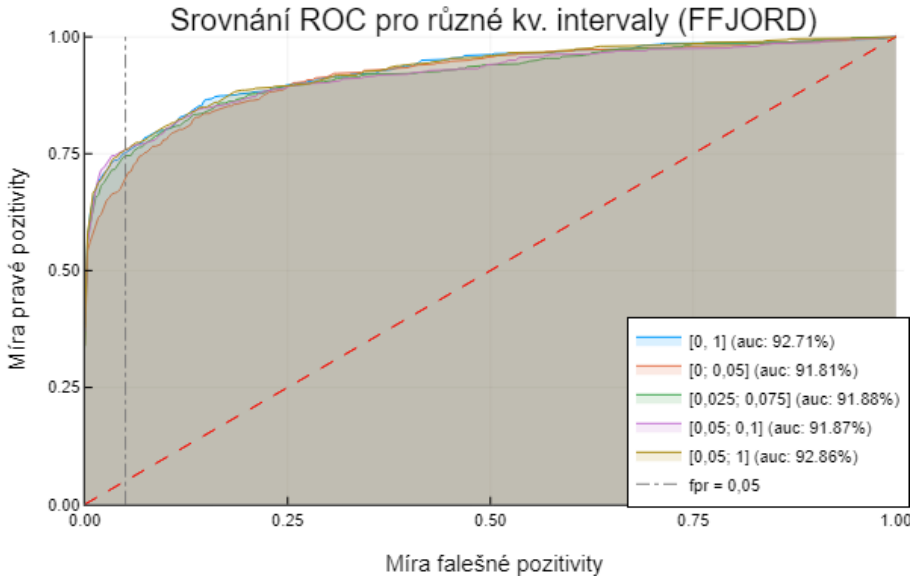
Je však třeba mít stále na paměti, že tyto uvedené postupy slouží pouze pro názornost a vizualizaci výsledných hustot. Je totiž potřeba udělat studii např. i s jinými hyper-parametry jednotlivých modelů atd. Proto v tomto případě nemá cenu ani srovnávat uvedenou metodu SPTN a FFJORD, jelikož užitá neuronová síť v metodě FFJORD má méně parametrů než architektura užité sítě SPTN. V následující sekci této práce provedeme korektnější srovnání několika modelů na různých kvantilových intervalech s reálnými daty (ty už budou často více rozměrná, proto bohužel nebudeme moci vizualizovat výslednou hustotu pravděpodobnosti).



(a) (b)

(c) (d)

Obrázek 5.8: Výsledky hustot pravděpodobností získané trénováním metody SPTN na různých kvantilových intervalech: (a) $[0;\ 0,05]$, (b) $[0,025;\ 0,075]$, (c) $[0,05;\ 0,1]$, (d) $[0,05;\ 1]$.



Obrázek 5.9: Srovnání ROC křivek metody FFJORD na kvantilových intervalech $[0,1]$, $[0;\ 0,05]$, $[0,025;\ 0,075]$, $[0,05;\ 0,1]$, $[0,05;\ 1]$.

5.2 Evaluace modifikované ztrátové funkce na reálných datech

Nyní se přesuneme k experimentům učení pomocí modifikované ztrátové funkce na reálných datech. Jedná se o různorodé soubory dat obsahující reálné předem klasifikované anomálie. Shrnutí těchto souborů dat můžeme vidět v tabulce 5.3.

Soubor dat	Dimenze	Anomálie	Normální
abalone	10	50	2151
blood-transfusion	4	16	382
breast-cancer-wisconsin	30	206	356
breast-tissue	9	22	65
cardiotocography	27	228	1830
ecoli	7	108	205
glass	10	94	112
haberman	3	14	225
ionosphere	33	122	225
iris	4	46	100
isolet	617	3300	4496
letter-recognition	617	3600	4196
libras	90	142	215
magic-telescope	10	3882	12331
miniboone	50	23922	93565
multiple-features	649	800	1200
page-blocks	10	384	4911
parkinsons	22	44	146
pendigits	16	5384	5537
pima-indians	8	176	500
sonar	60	96	110
spect-heart	44	52	211
statlog-satimage	36	2630	3592
statlog-segment	18	938	1320
statlog-shuttle	8	28	57767
statlog-vehicle	18	132	627
synthetic-control-chart	60	200	400
wall-following-robot	24	2220	2921
waveform-1	21	1482	3302
waveform-2	21	1472	3302
wine	13	70	106
yeast	8	390	751

Tabulka 5.3: Shrnutí použitých reálných dat. Tabulka obsahuje příslušné dimenze (tzn. počet příznaků), počet vyskytujících se anomálií a počet normálních dat.

Jednotlivé soubory dat jsou volně ke stažení z [5]. Pro představu si zmiňme několik příkladů toho, co jednotlivé soubory reprezentují. Soubor **iris** obsahuje popis květin z rodu kosatců, jednotlivé příznaky představují rozměry listů jednotlivých květin.

Soubor **breast-cancer-wisconsin** obsahuje data vyhotovená ze snímků prsní tkáně několika žen. Příznaky představují charakteristiky jádra buněk prsní tkáně. Za anomální data byla označena data patřící ženám podezřelých z rakoviny prsa. Soubor **wine** obsahuje výsledky chemické analýzy vín (alkoholického nápoje) z oblasti v Itálii. Anomální data byla buďto vyhodnocena experty nebo vytvořena z datového souboru více tříd.

Nyní se přesuneme k popisu evaluace jednotlivých modelů. K evaluaci byla použita knihovna **GenerativeAD.jl** v programovacím jazyce **Julia** [1] vytvořena kolegy na fakultě elektrotechnické. Knihovna slouží k porovnání několika metod strojového učení pro detekci anomálií. Výsledky tohoto porovnání jsou dostupné v článku [20]. Knihovna je dostupná na adrese `https://github.com/aicenter/GenerativeAD.jl`. Tato knihovna je rovněž připravena pro práci na výpočetním clusteru **RCI**, na kterém jsem rovněž prováděl dále popsané výpočty. Knihovna dále umožňuje provádět výpočty paralelně na více vláknech.

Knihovna již obsahovala implementaci pravděpodobnostních modelů SPTN, MAF a RealNVP popasných v kapitole 2. Tyto modely jsem rozšířil o modifikovanou ztrátovou funkci na kvantilovém intervalu popsanou v kapitole 4. Dále jsem knihovnu rozšířil o metodu FFJORD spolu se zmíněnou modifikací ztrátové funkce.

Použití knihovny můžeme shrnout do dvou kroků - trénování modelů a jejich evaluace.

Před započetím trénování určíme poměry trénovacího, validačního a testovacího souboru. Normální data opět náhodným výběrem rozdělíme v poměru $6 : 2 : 2$ opět na trénovací, validační a testovací soubor. Anomální data rozdělíme v poměrů $1 : 1$ na validační a testovací soubor. Trénovací soubor ponecháme bez anomálií a validační/testovací soubory normálních a anomálních dat smísíme dohromady. Jedná se tedy o *čisté* trénování popsané v sekci 3.3.

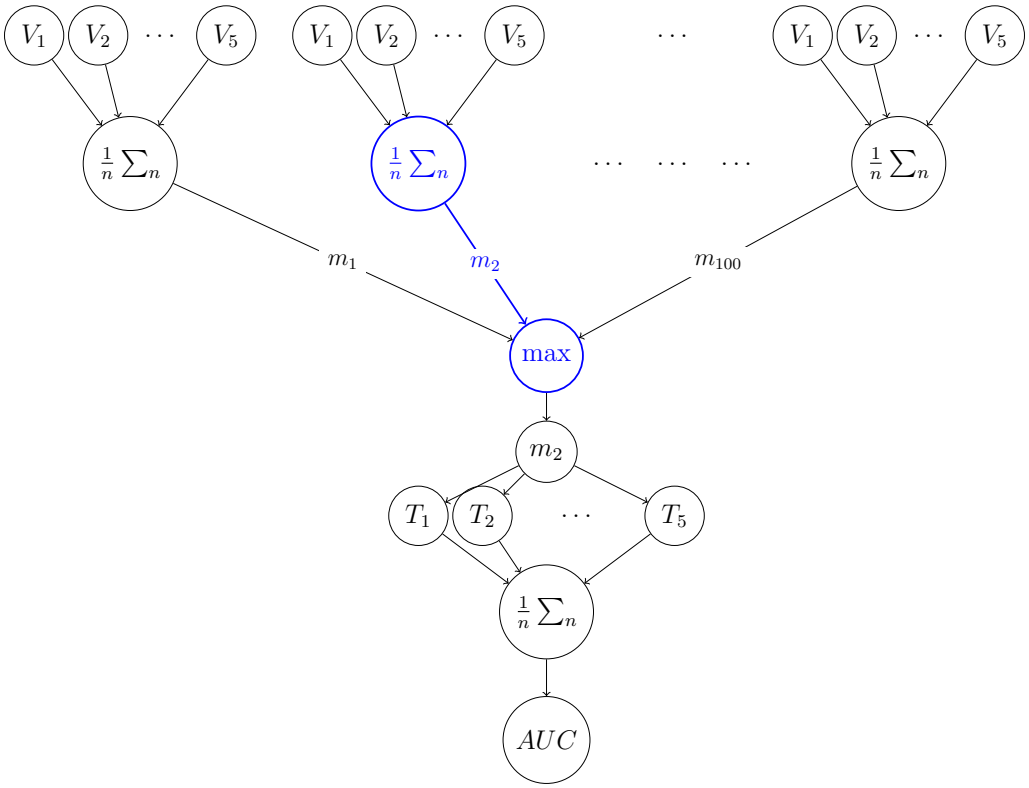
Pro většinu metod strojového učení předem nevíme, jak vzhledem k datům nejlépe zkonstruovat nejlepší model. Proto ke každému souboru dat náhodně vykonstruujeme 100 modelů s různými hyper-parametry. V případě metody FFJORD např. náhodně určíme v nacházející se neuronové síti počet skrytých vrstev, jejich rozměry a příslušné aktivační funkce. Zároveň pro každý ze 100 modelů provedeme 5 různých křížových validací - neboli provedeme 5 různých náhodných rozdělení na trénovací, validační a testovací soubor. Pro každý soubor dat tedy nezávisle natrénujeme 500 modelů. Dohromady pro každou metodu natrénujeme pro 32 zmíněných datových souborů 16 000 modelů.

Pro zastavení trénování byla zvolena dvě kritéria. Buď bylo provedeno 10 000 iterací pro optimalizaci, nebo od počátku trénování modelu uplynulo 24 hodin. K optimalizaci pomocí gradientů byla použita metoda ADAM [10].

Po natrénování všech modelů se můžeme přesunout k evaluaci. Na každém datovém souboru chceme vybrat nejlepší model. Zároveň na každém souboru máme natrénováno 100 modelů s různými hyper-parametry a pro každý z nich varianty na 5 různých křížových validacích. Pro každou variantu hyper-parametrů vyčíslíme hodnotu zvolené metriky (v této práci hodnotu AUC, resp. TPR pro $FPR = 0,05\%$)

na všech 5 křížových validacích na validačních datech. Těchto 5 hodnot dále zprůměrujeme a přiřadíme ji danému modelu s příslušnými hyper-parametry. Pro datový soubor máme nyní 100 modelů s příslušnými hodnotami na validačních datech. Z těchto modelů vybereme ten s nejvyšší hodnotou na validačních datech.

Takto vybraný model dále vyhodnotíme se zvolenou metrikou na 5 různých křížových validacích na testovacích datech a získaných 5 hodnot opět zprůměrujeme. Tato získaná hodnota je indikátor toho, jak se model bude chovat v praxi - tedy pomocí této hodnoty budeme porovnávat jednolité metody strojového učení na daném datovém souboru. Popsaný postup můžeme vidět graficky znázorněný na obrázku 5.10.



Obrázek 5.10: Grafické znázornění evaluace modelů knihovny GenerativeAD.jl. Vrcholy $V_1, \ldots, V_5$ znázoňují křížové validace na validačních datech, resp. $T_1, \ldots, T_5$ na datech testovacích. Hrany $m_1, m_2, \ldots, m_{100}$ znázorňují modely s různými hyper-parametry. Jako nejlepší model na validačních datech byl zvolen model m_2.

V následujících tabulkách uvidíme výsledky metod SPTN, RealNVP, MAF, FFJORD a jejich modifikací pomocí upravené ztrátové funkce na testovacích datech jednotlivých datových souborů z hlediska hodnot AUC a TPR pro $FPR = 5\%$. Zároveň na posledním řádku každé tabulky uvedeme tzv. *průměrné pořadí*. Tato hodnota udává, jak si metoda průměrně vedla oproti ostatním metodám ve stejné tabulce. Neboli pro každý datový soubor seřadíme podle velikosti dané metriky v tabulce všechny metody (řadíme metody přes řádek tabulky) a zapamatujeme si jejich umístění (1., 2. místo atd.). Dále pro každou metodu zprůměrujeme její pořadí na všech datových souborech (průměrujeme přes sloupce tabulek). Průměrné pořadí je dobrým

indikátorem toho, jak metoda obstojí oproti jiným metodám na nových datových souborech.

Poznámka. Jednotlivé metody a jejich modifikace pomocí ztrátové funkce na kvantilovém intervalu zde chápeme jako rozdílné metody. V tabulkách tedy budeme srovnávat např. SPTN a SPTN na kvantilovém intervalu $[5\%, 10\%]$.

Dále se přesuneme k výsledkům jednotlivých metod. V tabulkách 5.4 a 5.5 můžeme vidět srovnání modifikace ztrátové funkce pro metodu SPTN z hlediska hodnot AUC a TPR při $FPR = 5\%$.

Soubor dat	**sptn**	**sptn na** $[2,5\%,7,5\%]$	**sptn na** $[5\%,10\%]$
abalone	0.91	**0.92**	**0.92**
blood-transfusion	0.94	**0.96**	**0.96**
breast-cancer-wisconsin	0.95	**0.98**	0.97
breast-tissue	0.99	**1.00**	**1.00**
cardiotocography	0.50	**0.66**	0.58
ecoli	0.88	**0.90**	**0.90**
glass	0.78	0.80	**0.81**
haberman	**0.96**	**0.96**	0.94
ionosphere	0.97	**0.99**	**0.99**
iris	**0.93**	0.79	0.82
isolet	**0.60**	0.59	**0.60**
letter-recognition	**0.67**	0.64	0.65
libras	0.55	**0.56**	0.55
magic-telescope	**0.96**	0.95	0.95
miniboone	0.86	0.86	**0.87**
multiple-features	**0.94**	0.90	0.89
page-blocks	0.98	**0.99**	**0.99**
parkinsons	0.74	**0.85**	0.81
pendigits	**0.99**	**0.99**	**0.99**
pima-indians	**0.84**	**0.84**	0.83
sonar	0.58	0.63	**0.64**
spect-heart	0.28	0.32	**0.33**
statlog-satimage	**0.84**	0.81	0.83
statlog-segment	0.93	0.93	**0.94**
statlog-shuttle	**1.00**	**1.00**	**1.00**
statlog-vehicle	**0.74**	0.72	0.71
synthetic-control-chart	0.90	0.91	**0.93**
wall-following-robot	**0.81**	0.78	0.79
waveform-1	**0.77**	0.73	0.71
waveform-2	**0.77**	**0.77**	0.71
wine	0.96	0.97	**0.98**
yeast	0.67	**0.73**	0.71
Průměrné pořadí	2.0	**1.7**	**1.7**

Tabulka 5.4: (AUC) Porovnání metody SPTN a její modifikace na kvantilových intervalech $[2,5\%,7,5\%]$ a $[5\%,10\%]$ pomocí hodnoty AUC.

Soubor dat	**sptn**	**sptn na** $[2,5\%,7,5\%]$	**sptn na** $[5\%,10\%]$
abalone	0.46	0.55	**0.58**
blood-transfusion	**0.85**	0.80	0.80
breast-cancer-wisconsin	0.54	**0.75**	0.74
breast-tissue	0.85	0.95	**0.98**
cardiotocography	**0.15**	0.08	0.06
ecoli	0.40	**0.52**	0.46
glass	0.05	**0.31**	0.20
haberman	**0.69**	0.60	0.66
ionosphere	0.91	**0.92**	0.90
iris	**0.60**	0.16	0.32
isolet	0.16	**0.17**	0.16
letter-recognition	**0.21**	0.19	0.20
libras	0.05	**0.06**	0.03
magic-telescope	**0.79**	0.78	0.77
miniboone	**0.47**	0.45	0.46
multiple-features	**0.64**	0.48	0.47
page-blocks	0.92	0.93	**0.95**
parkinsons	0.15	0.28	**0.37**
pendigits	**0.98**	**0.98**	**0.98**
pima-indians	**0.30**	0.22	0.24
sonar	0.01	**0.06**	**0.06**
spect-heart	**0.00**	**0.00**	**0.00**
statlog-satimage	0.24	**0.31**	0.28
statlog-segment	**0.75**	0.74	0.69
statlog-shuttle	**1.00**	**1.00**	**1.00**
statlog-vehicle	**0.17**	0.16	**0.17**
synthetic-control-chart	0.64	0.73	**0.74**
wall-following-robot	**0.22**	0.20	**0.22**
waveform-1	**0.31**	0.22	0.19
waveform-2	**0.29**	**0.29**	0.20
wine	0.64	0.81	**0.90**
yeast	0.10	**0.12**	0.10
Průměrné pořadí	**1.8**	**1.8**	1.9

Tabulka 5.5: (TPR) Porovnání metody SPTN a její modifikace na kvantilových intervalech $[2,5\%,7,5\%]$ a $[5\%,10\%]$ pomocí hodnoty TPR při $FPR = 5\%$.

Pro metriku AUC vidíme, že modifikace přinesla v obou případech mírné zlepšení na většině datových souborů. V případě srovnání pomocí TPR jsou pro většinu dat výsledky pro všechny 3 varianty srovnatelné.

Nyní v tabulkách 5.6 a 5.7 uveďme výsledky pro metodu FFJORD. Metodu FFJORD jsme však neměli paralelizovanou na grafické kartě (paralelizace na grafické kartě by byla využita při výpočtu vyskytujících se diferenciálních rovnic). Z tohoto důvodu byla metoda při učení výrazně pomalejší než ostatní. Proto jsme zanedbali trénování na datových souborech s dimenzí větší nebo rovnou než 50. Z těchto tabulek dále vidíme, že modifikace ztrátové funkce na kvantilovém intervalu $[5\%, 10\%]$ nepřinesla z hlediska obou metrik zlepšení na většině datových souborů.

Nakonec v tabulkách 5.8 a 5.9 můžeme vidět výsledky pro metodu MAF a v tabulkách 5.10 a 5.11 výsledky pro metodu RealNVP. Z těchto tabulek bohužel opět

Soubor dat	ffjord	ffjord na $[5\%, 10\%]$
abalone	**0.91**	0.88
blood-transfusion	0.92	**0.93**
breast-cancer-wisconsin	**0.97**	0.96
breast-tissue	**0.98**	**0.98**
cardiotocography	0.56	**0.62**
ecoli	**0.82**	0.81
glass	**0.92**	0.76
haberman	**0.96**	0.89
ionosphere	0.97	**0.98**
iris	0.88	**0.94**
magic-telescope	**0.96**	0.93
page-blocks	**0.99**	0.98
parkinsons	0.74	**0.75**
pendigits	**0.99**	0.97
pima-indians	**0.82**	0.81
spect-heart	0.35	**0.48**
statlog-satimage	0.67	**0.70**
statlog-segment	0.82	**0.89**
statlog-shuttle	**1.00**	**1.00**
statlog-vehicle	**0.73**	0.66
wall-following-robot	**0.79**	0.74
waveform-1	**0.66**	0.63
waveform-2	**0.64**	0.61
wine	0.98	**0.99**
yeast	0.66	**0.69**
Průměrné pořadí	**1.4**	1.5

Tabulka 5.6: (AUC) Porovnání metody FFJORD a její modifikace na kvantilovém intervalu $[5\%, 10\%]$ pomocí hodnoty AUC.

Soubor dat	ffjord	ffjord na $[5\%, 10\%]$
abalone	**0.51**	0.46
blood-transfusion	**0.85**	0.82
breast-cancer-wisconsin	0.60	**0.74**
breast-tissue	**0.89**	**0.89**
cardiotocography	**0.07**	**0.07**
ecoli	**0.30**	0.29
glass	**0.57**	0.14
haberman	**0.74**	0.60
ionosphere	**0.92**	0.89
iris	0.40	**0.61**
magic-telescope	**0.78**	0.72
page-blocks	**0.93**	0.90
parkinsons	**0.20**	0.11
pendigits	**0.94**	0.87
pima-indians	0.23	**0.24**
spect-heart	0.00	**0.04**
statlog-satimage	0.07	**0.14**
statlog-segment	0.49	**0.59**
statlog-shuttle	**1.00**	**1.00**
statlog-vehicle	**0.21**	0.15
wall-following-robot	**0.26**	0.20
waveform-1	**0.15**	0.12
waveform-2	**0.15**	0.10
wine	**0.81**	0.76
yeast	0.12	**0.13**
Průměrné pořadí	**1.3**	1.6

Tabulka 5.7: (TPR) Porovnání metody FFJORD a její modifikace na kvantilovém intervalu $[5\%, 10\%]$ pomocí hodnoty TPR při $FPR = 5\%$.

vidíme, že v obou případech nepřinesla modifikace ztrátové funkce zlepšení z hlediska průměrného pořadí.

5.3 Diskuze

V poslední sekci této práce se pokusíme okomentovat výsledky modifikace ztrátové funkce. Před tím však zopakujeme naši původní motivaci k provedení této modifikace. Jak je vidno z obrázků hustot pravděpodobností v sekci 5.1, model po úpravě ztrátové funkce preferuje nalezení objemu s nejvyšší koncentrací dat nad exaktním nalezení hustoty pravděpodobnosti. V metodách strojového učení se obecně preferuje užívat metody, které se při řešení daného problému neučí zbytečné informace. Naučení se exaktní hustoty pravděpodobnosti za cenu horšího odhadu objemu s nejvyšší koncentrací dat bychom při detekci anomálií mohli považovat právě za učení se zbytečné informace. Tuto hypotézu můžeme podpořit výsledky z článku [20], kdy klasifikátor jedné třídy (osvm) dosahuje oproti ostatním metodám nejlepšího průměrného pořadí. Dále jsme předpokládali, že při apriorním požadavku na hodnotu FPR, v práci $FPR = 5\%$, dosáhneme lepších hodnot TPR.

Bohužel jak vidíme z výsledků v předešlé sekci, razantního zlepšení jsme úpravou ztrátové funkce nedosáhli. Pokusíme se tedy alespoň nalézt odpověď na to, proč jsme chtěného zlepšení nedosáhli.

Soubor dat	MAF	MAF na $[2,5\%,7,5\%]$
abalone	**0.91**	0.90
blood-transfusion	**0.96**	0.93
breast-cancer-wisconsin	**0.99**	0.97
breast-tissue	**0.99**	0.97
cardiotocography	0.60	**0.65**
ecoli	**0.90**	0.84
glass	0.75	**0.77**
haberman	**0.96**	0.88
ionosphere	**0.98**	**0.98**
iris	**0.79**	0.74
isolet	0.71	**0.72**
letter-recognition	0.76	**0.77**
libras	**0.73**	0.72
magic-telescope	**0.96**	0.94
miniboone	**0.90**	0.86
multiple-features	**0.98**	0.97
page-blocks	**0.99**	0.98
parkinsons	0.72	**0.75**
pendigits	**0.98**	**0.98**
pima-indians	**0.86**	0.79
sonar	**0.65**	**0.65**
spect-heart	0.31	**0.43**
statlog-satimage	0.91	**0.93**
statlog-segment	0.92	**0.95**
statlog-shuttle	**1.00**	**1.00**
statlog-vehicle	**0.76**	0.73
synthetic-control-chart	**0.99**	**0.99**
wall-following-robot	**0.78**	0.77
waveform-1	**0.75**	0.69
waveform-2	**0.74**	0.72
wine	**0.98**	0.96
yeast	**0.72**	0.69
Průměrné pořadí	**1.2**	1.6

Tabulka 5.8: (AUC) Porovnání metody MAF a její modifikace na kvantilovém intervalu $[2,5\%,7,5\%]$ pomocí hodnoty AUC.

Soubor dat	MAF	MAF na $[2,5\%,7,5\%]$
abalone	0.51	**0.54**
blood-transfusion	**0.85**	**0.85**
breast-cancer-wisconsin	**0.92**	0.89
breast-tissue	**0.93**	0.82
cardiotocography	**0.11**	0.10
ecoli	**0.39**	0.30
glass	0.13	**0.33**
haberman	**0.74**	0.38
ionosphere	**0.92**	0.91
iris	**0.36**	0.30
isolet	**0.24**	**0.24**
letter-recognition	0.32	**0.33**
libras	0.11	**0.15**
magic-telescope	**0.84**	0.79
miniboone	**0.48**	0.45
multiple-features	**0.88**	0.80
page-blocks	**1.00**	0.92
parkinsons	**0.33**	0.29
pendigits	0.93	**0.94**
pima-indians	**0.31**	0.22
sonar	**0.06**	0.05
spect-heart	**0.00**	**0.00**
statlog-satimage	0.52	**0.66**
statlog-segment	0.75	**0.79**
statlog-shuttle	**1.00**	**1.00**
statlog-vehicle	0.17	**0.25**
synthetic-control-chart	0.95	**0.96**
wall-following-robot	0.27	**0.37**
waveform-1	**0.30**	0.17
waveform-2	**0.21**	**0.21**
wine	0.77	**0.82**
yeast	**0.14**	0.10
Průměrné pořadí	**1.3**	1.5

Tabulka 5.9: (TPR) Porovnání metody MAF a její modifikace na kvantilovém intervalu $[2,5\%,7,5\%]$ pomocí hodnoty TPR při $FPR = 5\%$.

Soubor dat	RealNVP	RealNVP na $[5\%,10\%]$
abalone	0.90	**0.91**
blood-transfusion	0.94	**0.96**
breast-cancer-wisconsin	**0.98**	0.97
breast-tissue	0.99	**1.00**
cardiotocography	0.51	**0.70**
ecoli	0.85	**0.86**
glass	**0.75**	0.71
haberman	**0.96**	0.85
ionosphere	**0.99**	**0.99**
iris	**0.80**	**0.80**
isolet	0.70	**0.72**
letter-recognition	0.75	**0.76**
libras	**0.77**	0.74
magic-telescope	**0.96**	0.94
miniboone	**0.90**	0.87
multiple-features	**0.99**	0.97
page-blocks	**0.99**	**0.99**
parkinsons	**0.77**	0.75
pendigits	**0.99**	**0.99**
pima-indians	**0.85**	0.79
sonar	**0.66**	**0.66**
spect-heart	0.31	**0.38**
statlog-satimage	**0.93**	0.92
statlog-segment	0.92	**0.93**
statlog-shuttle	**0.99**	0.92
statlog-vehicle	**0.77**	0.74
synthetic-control-chart	**0.96**	**0.96**
wall-following-robot	**0.82**	0.74
waveform-1	**0.75**	0.69
waveform-2	**0.79**	0.69
wine	**0.95**	0.94
yeast	**0.72**	0.71
Průměrné pořadí	**1.3**	1.5

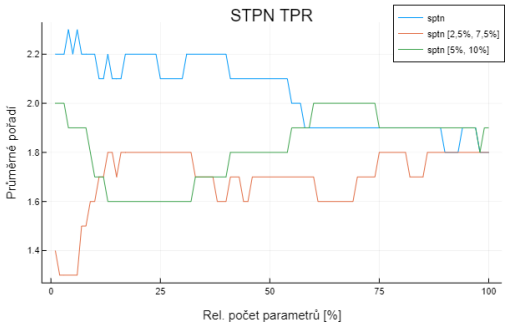
Tabulka 5.10: (AUC) Porovnání metody RealNVP a její modifikace na kvantilovém intervalu $[5\%,10\%]$ pomocí hodnoty AUC.

Soubor dat	RealNVP	RealNVP na $[5\%,10\%]$
abalone	**0.47**	0.42
blood-transfusion	0.82	**0.85**
breast-cancer-wisconsin	**0.91**	0.86
breast-tissue	0.95	**0.98**
cardiotocography	**0.15**	0.09
ecoli	**0.48**	0.35
glass	**0.10**	0.09
haberman	**0.69**	0.43
ionosphere	**0.94**	0.90
iris	0.37	**0.45**
isolet	**0.25**	0.23
letter-recognition	**0.28**	0.26
libras	0.15	**0.19**
magic-telescope	**0.82**	0.76
miniboone	**0.47**	0.40
multiple-features	**0.96**	0.79
page-blocks	**0.93**	0.90
parkinsons	**0.36**	0.25
pendigits	0.93	**0.94**
pima-indians	**0.25**	0.21
sonar	**0.09**	0.06
spect-heart	**0.04**	0.00
statlog-satimage	0.67	**0.71**
statlog-segment	**0.78**	**0.78**
statlog-shuttle	**0.97**	0.44
statlog-vehicle	0.27	**0.30**
synthetic-control-chart	0.84	**0.87**
wall-following-robot	**0.32**	0.25
waveform-1	**0.19**	0.18
waveform-2	0.18	**0.19**
wine	**0.77**	0.61
yeast	0.10	**0.11**
Průměrné pořadí	**1.3**	1.7

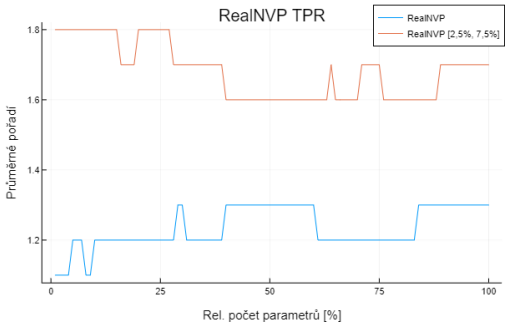
Tabulka 5.11: (TPR) Porovnání metody RealNVP a její modifikace na kvantilovém intervalu $[5\%,10\%]$ pomocí hodnoty TPR při $FPR = 5\%$.

Nejprve se podíváme na to, zda-li robustnost evaluační knihovny GenerativeAD.jl nemohla paradoxně způsobit horší výsledky pro metody s učením na kvantilovém intervalu. Knihovna umožnila vygenerovat pro každý datový soubor velký počet různých modelů, takže i modely s vysokým počtem parametrů. Modely využívající maximálně věrohodný odhad tedy mohly mít dostatečný počet parametrů na to, aby se naučily jak exaktní hustotu pravděpodobnosti, tak objem s nejvyšší koncentrací dat. Pomocí grafu jsme se pokusili zjistit, zda-li tato skutečnost nastala. Graf znázorňuje průměrné pořadí modelů v závislosti na relativním počtu parametrů jednotlivých modelů. Neboli pro každý datový soubor provedeme evaluaci jen vůči pouze např. 1%, 2%, 3%, ... modelům s nejnižším počtem parametrů. Hodnota 100% tedy představuje všechny modely použité při evaluaci v předešlé sekci.

Na obrázku 5.11 můžeme vidět výše popsaný graf pro metodu SPTN z hlediska hodnoty TPR pro $FPR = 5\%$.



Obrázek 5.11: Graf znázorňující závislost průměrného pořadí na relativním počtu parametrů modelu SPTN a příslušných modifikací na kvantilových intervalech $[2,5\%, 7,5\%]$ a $[5\%, 10\%]$.



Obrázek 5.12: Graf znázorňující závislost průměrného pořadí na relativním počtu parametrů modelu RealNVP a příslušné modifikace na kvantilovém intervalu $[2,5\%, 7,5\%]$.

Vidíme, že v tomto případě se naše premisa splnila, jelikož průměrné pořadí pro učení pomocí maximálně věrohodného odhadu klesá (pro kvantilové intervaly naopak stoupá). Bohužel pro ostatní modely příslušné křivky jak z hlediska hodnoty TPR i hodnoty AUC zůstávají přibližně konstantní. Příklad této situace můžeme vidět na obrázku 5.12 pro metodu RealNVP z hlediska hodnoty TPR.

Dále jsme se zabývali možností, že modely s modifikovanou ztrátovou funkcí mají větší tendenci se *přeučit*. Jinými slovy metody s touto modifikací dosahují výrazně lepších výsledků na trénovacích datech, než na datech testovacích. Trénovací soubory jednotlivých datových souborů neobsahovaly anomálie. Na konci trénování tedy pro každá data vybereme anomálie z testovacího souboru a přidáme je do příslušných trénovacích souborů (nebo-li trénovací soubor zašpiníme). Dále provedeme analogickou evaluaci popsanou v předešlé sekci s tím, že výběr nejlepších modelů provádíme z hlediska metrik na zašpiněných trénovacích souborech. V následujících tabulkách 5.12-5.15 uvádíme hodnoty TPR na těchto trénovacích datech.

Z těchto tabulek vidíme, že modifikace metod MAF a RealNVP dosahuje lepších výsledků na trénovacích datech, než při použití maximálně věrohodného odhadu.

Zároveň z tabulek 5.9 a 5.11 vidíme, že tato modifikace dosahuje na testovacích datech horších výsledků. Z toho můžeme usoudit, že modifikace ztrátové funkce v případě těchto dvou metod má větší tendenci k přeučení.

Naopak v případě metody SPTN dostáváme lepší výsledky na trénovacích datech při použití maximálně věrohodného odhadu. V případě testovacích dat dosahovala lepších výsledků její modifikace. Z toho můžeme usoudit, že provedená modifikace naopak v případě metody SPTN snížila tendenci k přeučení.

Metoda FFJORD průměrně dosahuje na trénovacích datech podobných výsledků pro obě varianty. Zároveň však jeho modifikace podává na testovacích datech horší výsledky. Berme však v potaz, že metoda FFJORD se kvůli absenci paralelizace na grafické kartě učila výrazně pomaleji. To mohlo způsobit zkreslení samotných výsledků, jelikož metoda neměla dostatečný čas pro naučení (dříve uplynul 24 hodinový limit než 10 000 iterací).

Shrnutím výsledků můžeme usoudit, že modifikace ztrátové funkce prospěla pouze metodě SPTN. Modifikace v tomto případě dosahovala lepších výsledků na testovacích datech a zároveň vykazovala nižší tendenci k přeučení. Tento výsledek by mohl souviset s tím, že model SPTN patří to jiné třídy metod pro odhad hustoty pravděpodobnosti, než zbylé 3 modely. Metoda SPTN využívá k modelování hustoty směs více pravděpodobnostních distribucí. Může se tedy stát, že tato metoda bude při učení více preferovat např. jednu distribuci než ostatní. Tím, že premisou modifikace ztrátové funkce je dbát více na objem s nejvyšší koncentrací, se mohlo stát, že metoda měla menší tendenci k zanedbání zbylých distribucí. V budoucnu by tedy předmětem výzkumu mohlo být využití modifikace ztrátové funkce pro jiné modely, které využívají směšování více pravděpodobnostních distribucí.

Soubor dat	sptn	sptn na $[2, 5\%, 7, 5\%]$	sptn na $[5\%, 10\%]$
abalone	0.86	0.94	**0.95**
blood-transfusion	**0.95**	0.92	0.90
breast-cancer-wisconsin	**1.00**	0.99	0.97
breast-tissue	**1.00**	**1.00**	**1.00**
cardiotocography	**0.24**	0.19	0.22
ecoli	**0.97**	0.91	0.93
glass	**1.00**	0.97	**1.00**
haberman	**1.00**	**1.00**	**1.00**
ionosphere	0.99	**1.00**	**1.00**
iris	**0.82**	0.37	0.33
isolet	0.23	0.25	**0.28**
letter-recognition	0.29	0.28	**0.31**
libras	**0.18**	0.10	0.11
magic-telescope	0.92	0.93	**0.96**
miniboone	**0.48**	0.46	0.46
multiple-features	**1.00**	0.85	0.88
page-blocks	0.94	0.96	**0.98**
parkinsons	**0.92**	0.85	0.80
pendigits	**1.00**	**1.00**	**1.00**
pima-indians	0.91	**0.95**	**0.95**
sonar	**0.88**	0.82	0.79
spect-heart	0.27	**0.33**	0.12
statlog-satimage	**0.44**	**0.44**	0.42
statlog-segment	0.86	0.86	**0.90**
statlog-shuttle	**1.00**	**1.00**	**1.00**
statlog-vehicle	**0.88**	0.81	0.71
synthetic-control-chart	**0.94**	0.88	0.90
wall-following-robot	0.77	**0.79**	0.77
waveform-1	0.52	**0.71**	0.70
waveform-2	0.55	0.72	**0.74**
wine	**1.00**	0.99	**1.00**
yeast	**0.74**	0.68	0.67
Průměrné pořadí	**1.5**	2.0	1.8

Tabulka 5.12: (TPR) Evaluace metody SPTN a jejích modifikací na kvantilových intervalech na trénovacích datech z hlediska hodnoty TPR pro $FPR = 5\%$.

Soubor dat	ffjord	ffjord na $[5\%, 10\%]$
abalone	0.67	**0.70**
blood-transfusion	**0.85**	**0.85**
breast-cancer-wisconsin	**0.99**	**0.99**
breast-tissue	**1.00**	**1.00**
cardiotocography	**0.10**	0.08
ecoli	0.54	**0.78**
glass	**1.00**	0.93
haberman	0.66	**0.80**
ionosphere	**0.99**	**0.99**
iris	0.90	**0.97**
magic-telescope	**0.80**	0.74
page-blocks	0.94	**0.95**
parkinsons	**0.91**	0.84
pendigits	**0.96**	0.95
pima-indians	0.70	**0.79**
spect-heart	**1.00**	0.25
statlog-satimage	0.09	**0.13**
statlog-segment	0.52	**0.70**
statlog-shuttle	**1.00**	**1.00**
statlog-vehicle	**0.72**	0.48
wall-following-robot	**0.56**	**0.56**
waveform-1	**0.24**	0.23
waveform-2	**0.32**	0.22
wine	**1.00**	**1.00**
yeast	0.26	**0.27**
Průměrné pořadí	**1.4**	**1.4**

Tabulka 5.13: (TPR) Evaluace metody FFJORD a její modifikace na kvantilovém intervalu na trénovacích datech z hlediska hodnoty TPR pro $FPR = 5\%$.

Soubor dat	MAF	MAF na $[2, 5\%, 7, 5\%]$
abalone	0.72	**0.73**
blood-transfusion	**0.91**	0.90
breast-cancer-wisconsin	0.98	**1.00**
breast-tissue	**1.00**	**1.00**
cardiotocography	0.17	**0.37**
ecoli	0.69	**0.87**
glass	0.35	**0.90**
haberman	**0.74**	0.57
ionosphere	0.99	**1.00**
iris	0.48	**0.68**
isolet	0.84	**0.96**
letter-recognition	0.88	**0.99**
libras	0.70	**0.91**
magic-telescope	**0.86**	0.81
miniboone	**0.49**	0.45
multiple-features	**1.00**	**1.00**
page-blocks	**1.00**	0.95
parkinsons	0.68	**0.80**
pendigits	0.96	**0.98**
pima-indians	0.56	**0.67**
sonar	0.82	**1.00**
spect-heart	0.05	**0.27**
statlog-satimage	0.62	**0.89**
statlog-segment	0.88	**0.94**
statlog-shuttle	**1.00**	**1.00**
statlog-vehicle	0.40	**0.79**
synthetic-control-chart	0.98	**1.00**
wall-following-robot	0.51	**0.72**
waveform-1	0.37	**0.53**
waveform-2	0.32	**0.56**
wine	0.99	**1.00**
yeast	0.27	**0.32**
Průměrné pořadí	1.6	**1.3**

Tabulka 5.14: (TPR) Evaluace metody MAF a její modifikace na kvantilovém intervalu na trénovacích datech z hlediska hodnoty TPR pro $FPR = 5\%$.

Soubor dat	RealNVP	RealNVP na na $[5\%, 10\%]$
abalone	0.70	**0.74**
blood-transfusion	**0.90**	0.88
breast-cancer-wisconsin	0.98	**1.00**
breast-tissue	**1.00**	**1.00**
cardiotocography	0.24	**0.40**
ecoli	0.71	**0.83**
glass	0.35	**0.70**
haberman	**0.80**	0.57
ionosphere	**1.00**	**1.00**
iris	0.42	**0.67**
isolet	0.92	**0.95**
letter-recognition	0.95	**0.98**
libras	0.88	**0.93**
magic-telescope	**0.83**	0.79
miniboone	**0.48**	0.41
multiple-features	**1.00**	**1.00**
page-blocks	0.96	**0.97**
parkinsons	0.77	**0.92**
pendigits	0.97	**0.98**
pima-indians	**0.53**	0.43
sonar	0.96	**1.00**
spect-heart	**0.15**	0.07
statlog-satimage	0.77	**0.93**
statlog-segment	0.88	**0.92**
statlog-shuttle	**1.00**	0.50
statlog-vehicle	0.73	**0.85**
synthetic-control-chart	0.99	**1.00**
wall-following-robot	0.58	**0.65**
waveform-1	0.34	**0.56**
waveform-2	0.33	**0.64**
wine	0.99	**1.00**
yeast	0.29	**0.31**
Průměrné pořadí	1.6	**1.4**

Tabulka 5.15: (TPR) Evaluace metody RealNVP a její modifikace na kvantilovém intervalu na trénovacích datech z hlediska hodnoty TPR pro $FPR = 5\%$.

Závěr

V diplomové práce jsme se zabývali detekcí anomálií pomocí metod strojového učení. Konkrétně jsme se věnovali modelům odhadujícím hustotu pravděpodobnosti a modifikaci jejich procesu učení pomocí úpravy tzv. ztrátové funkce.

V první kapitole jsme se seznámili s matematickým pojmem anomálie a představili problematiku detekce anomálií pomocí modelů pro odhad hustoty pravděpodobnosti. Kapitolu jsme zakončili stručným popisem klasifikátorů jedné třídy a rekonstrukčních modelů. Klasifikátory jedné třídy umožňují nalézt tzv. nejmenší objem s nejvyšší koncentrací dat. Koncept tohoto objemu jsme se později snažili využít při učení modelů odhadujících hustotu pravděpodobnosti.

Z tohoto důvodu jsme se dále zabývali detailnějším popisem metod pro odhad hustoty pravděpodobnosti. Konkrétně jsme se nejprve zabývali třídou metod využívající transformaci náhodné veličiny. Z této třídy jsme si podrobně popsali zejména metodu FFJORD a stručně popsali princip metod MAF a RealNVP. Druhou třídu modelů tvořily tzv. Součtové-produktové a Součtové-produktové transformační sítě, které spadají to třídy modelů s grafovou reprezentací.

Jak již název napovídá, popsaná skupina modelů nám umožnila modelovat hustotu pravděpodobnosti neznámého rozdělení v závislosti na poskytnutých datech. Po nalezení aproximace neznáme hustoty pravděpodobnosti můžeme určit věrohodnost budoucích dat a na základě této hodnoty rozhodnout, zda-li je dané pozorování anomální, či nikoliv. Jinými slovy určíme tzv. práh normality - pokud věrohodnost daného pozorování leží nad tímto prahem, pak je toto pozorování vyhodnoceno jako normální (neanomální). Zároveň jsme popsali skutečnost, že množina hodnot, jejichž hustota pravděpodobnosti leží nad tímto prahem, je ekvivalentní se zmíněným nejmenším objemem s nejvyšší koncentrací dat.

Určení tohoto objemu jsme doposud prováděli až po natrénování daných modelů, tedy až po nalezení aproximace hustoty pravděpodobnosti. Hlavní ideou této práce bylo využít schopnost popsaných modelů určit věrohodnost v libovolném čase a tedy zahrnout minimalizaci objemu s nejvyšší koncentrací dat do samotného procesu učení. Toho jsme docílili modifikací ztrátové funkce pomocí tzv. kvantilového intervalu. Domnívali jsme se, že tato modifikace zároveň dovolí modelům soustředit svoji kapacitu na nalezení tohoto objemu za cenu méně přesné aproximace hustoty pravděpodobnosti. V rámci problému detekce anomálií by se ve finále mělo jednat o jednodušší úlohu.

Ve výpočetní studii této práce jsme se zabývali tím, zda-li popsaná modifikace při-

nesla lepší výsledky, než v případě klasického učení pomocí maximálně věrohodného odhadu. Studii jsme provedli v rámci zmíněných metod Součtových-produktových transformačních sítí, FFJORD, MAF a RealNVP. Modely jsme natrénovali na několika rozmanitých souborech dat. Průměrně lepších výsledků jsme bohužel dosáhli pouze v případě Součtových-produktových transformačních sítí.

Práci jsme zakončili diskuzí, kde jsme dospěli k závěru, že modifikace metod MAF a RealNVP způsobila větší náchylnost k jejich přeučení. V případě metody FFJORD mohla výsledky zkreslit její výrazně delší doba potřebná k naučení modelu. Naopak lepší výsledky Součtových-produktových transformačních sítí mohla zapříčinit principiální odlišnost tohoto modelu oproti zbylým třem, jelikož k modelování hustoty pravděpodobnosti používá směsi několika předem zvolených pravděpodobnostních distribucí. Předmětem budoucího výzkumu by tedy mohlo být využití modifikace ztrátové funkce pro modely, které k aproximaci hustoty pravděpodobnosti také využívají zmíněné směšování pravděpodobnostních distribucí.

Literatura

[1] BEZANSON, Jeff, Alan EDELMAN, Stefan KARPINSKI a Viral B. SHAH. Julia: A Fresh Approach to Numerical Computing. *SIAM Review*. 2017, **59**(1), 65-98. ISSN 0036-1445. Dostupné z: doi:10.1137/141000671

[2] BÍLEK, Viktor. *Odhad pravděpodobnostních rozdělení pomocí neuronových sítí*. Praha, Česká Republika, 2020. Výzkumný úkol. Fakulta jaderná a fyzikálně inženýrská, České vysoké učení technické v Praze. Vedoucí práce Tomáš Pevný.

[3] BÍLEK, Viktor. *Algoritmy strojového učení pro aproximaci inverzních matic*. Praha, Česká Republika, 2019. Bakalářská práce. Fakulta jaderná a fyzikálně inženýrská, České vysoké učení technické v Praze. Vedoucí práce Tomáš Oberhuber.

[4] DINH, Laurent, Jascha SOHL-DICKSTEIN a Samy BENGIO. *Density estimation using Real NVP* [online]. 27.5.2016 [cit. 2021-03-30]. Dostupné z: https://arxiv.org/abs/1605.08803

[5] DUA, Dheeru a Casey GRAFF. *UCI Machine Learning Repository* [online]. University of California, Irvine, School of Information and Computer Sciences, 2019 [cit. 2021-04-14]. Dostupné z: http://archive.ics.uci.edu/ml

[6] GOODFELLOW, Ian, Yoshua BENGIO a Aaron COURVILLE. *Deep learning* [online]. Cambridge, Massachusetts: The MIT Press, 2016 [cit. 2019-07-02]. ISBN 978-026-2035-613. Dostupné z: https://www.deeplearningbook.org/

[7] GRATHWOHL, Will, Ricky T. Q. CHEN, Jesse BETTENCOURT a David K. DUVENAUD. *FFJORD: Free-form Continuous Dynamics for Scalable Reversible Generative Models* [online]. 30.10.2018, (1810.01367) [cit. 2019-07-02]. Dostupné z: http://arxiv.org/abs/1810.01367

[8] HUTCHINSON, M.F. A Stochastic Estimator of the Trace of the Influence Matrix for Laplacian Smoothing Splines. *Communications in Statistics - Simulation and Computation* [online]. 1989, **18**(3), 1059-1076 [cit. 2019-07-10]. ISSN 0361-0918. Dostupné z: doi:10.1080/03610918908812806

[9] CHEN, Tian Qi, Yulia RUBANOVA, Jesse BETTENCOURT a David K. DUVENAUD. *Neural Ordinary Differential Equations* [online]. 19.6.2018 [cit. 2019-07-02]. Dostupné z: http://arxiv.org/abs/1806.07366

[10] KINGMA, Diederik P. a Jimmy BA. *Adam: A Method for Stochastic Optimization* [online]. 22.12.2014 [cit. 2021-04-17]. Dostupné z: https://arxiv.org/abs/1412.6980

[11] KOSKI, Timo a John NOBLE. *Bayesian Networks: An Introduction*. New York City, New York: Wiley, 2009. ISBN 978-0-470-74304-1.

[12] NUÑEZ GARCIA, Javier, Zoltan KUTALIK, Kwang-Hyun CHO a Olaf WOLKENHAUER. Level sets and minimum volume sets of probability density functions. *International Journal of Approximate Reasoning* [online]. 2003, **34**(1), 25-47 [cit. 2021-04-08]. ISSN 0888613X. Dostupné z: doi:10.1016/S0888-613X(03)00052-5

[13] PAPAMAKARIOS, George, Theo PAVLAKOU a Iain MURRAY. *Masked Autoregressive Flow for Density Estimation* [online]. 19.4.2017 [cit. 2020-08-20]. Dostupné z: https://arxiv.org/abs/1705.07057

[14] PEVNÝ, Tomáš, Václav ŠMÍDL, Martin TRAPP, Ondřej POLÁČEK a Tomáš OBERHUBER. *Sum-Product-Transform Networks: Exploiting Symmetries using Invertible Transformations* [online]. 4.4.2020 [cit. 2020-07-25]. Dostupné z: https://arxiv.org/pdf/2005.01297.pdf

[15] POON, Hoifung a Pedro DOMINGOS. *Sum-Product Networks: A New Deep Architecture* [online]. 14.2.2012 [cit. 2020-07-25]. Dostupné z: https://arxiv.org/ftp/arxiv/papers/1202/1202.3732.pdf

[16] REZENDE, Danilo Jimenez a Shakir MOHAMED. *Variational Inference with Normalizing Flows* [online]. 21.5.2015 [cit. 2021-03-30]. Dostupné z: https://arxiv.org/abs/1505.05770

[17] RUFF, Lukas, Jacob R. KAUFFMANN, Robert A. VENDERMEULEN, Grégoire MONTAVON, Wojciech SAMEK, Marius KLOFT, Thomas G. DIETTERICH a Klaus-Robert MÜLLER. *A Unifying Review of Deep and Shallow Anomaly Detection* [online]. 24.9.2021 [cit. 2021-03-30]. Dostupné z: doi:10.1109/JPROC.2021.3052449

[18] SCHÖLKOPF, Bernhard, John C. PLATT, John SHAWE-TAYLOR, Alex J. SMOLA a Robert C. WILLIAMSON. Estimating the Support of a High-Dimensional Distribution. *Neural Computation* [online]. 2001, 7.1.2001, **13**(7), 1443-1471 [cit. 2021-03-30]. ISSN 0899-7667. Dostupné z: doi:10.1162/089976601750264965

[19] STEINWART, Ingo, Don HUSH a Clint SCOVEL. A Classification Framework for Anomaly Detection. *Journal of Machine Learning Research* [online]. 2005, **6**(8), 211-232 [cit. 2021-03-30]. Dostupné z: http://jmlr.org/papers/v6/steinwart05a.html

[20] ŠKVÁRA, Vít, Jan FRANCŮ, Matěj ZOREK, Tomáš PEVNÝ a Václav ŠMÍDL. *Comparison of Anomaly Detectors: Context Matters* [online]. 11.12.2020 [cit. 2021-03-30]. Dostupné z: doi:Comparison of Anomaly Detectors: Context Matters

Poděkování

Rovněž děkuji za přístup k výpočetní infrastruktuře projektu financovaného CZ.02.1.01/0.0/0.0/16_019/0000765 “Research Center for Informatics”.