Univerzita Pardubice

Fakulta ekonomicko-správní

Modelování predikce časových řad návštěvnosti web domény pomocí SVM

Bc. Vlastimil Flegl

Diplomová práce

2011

Univerzita Pardubice
Fakulta ekonomicko-správní
Akademický rok: **2010/2011**

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení:	**Bc. Vlastimil FLEGL**
Osobní číslo:	**E090485**
Studijní program:	**N6209 Systémové inženýrství a informatika**
Studijní obor:	**Informatika ve veřejné správě**
Název tématu:	**Modelování predikce časových řad návštěvnosti web domény pomocí SVM**
Zadávající katedra:	**Ústav systémového inženýrství a informatiky**

Zásady pro vypracování:

- Analyzujte vstupní data (parametry) pro následující predikci.
- Charakterizujte SVM z hlediska aproximace a predikce.
- Navrhněte model na predikci návštěvnosti web domény.
- Verifikujte navrhnutý model.
- Uskutečněte analýzu výsledků.

Rozsah grafických prací:

Rozsah pracovní zprávy:

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

KVASNIČKA, V. a kol.: Úvod do teórie neurónových sietí. Iris, Bratislava, 1997.

HAYKIN, S.: Neural Networks: A Comprehensive Foundation. 2nd edition, New Jersey, Prentice-Hall, Inc., 1999, 842s.

Vedoucí diplomové práce: **prof. Ing. Vladimír Olej, CSc.**
Ústav systémového inženýrství a informatiky

Datum zadání diplomové práce: **5. října 2010**

Termín odevzdání diplomové práce: **6. května 2011**

L.S.

doc. Ing. Renáta Myšková, Ph.D.
děkanka

doc. Ing. Jiří Křupka, Ph.D.
vedoucí ústavu

V Pardubicích dne 5. října 2010

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 22. 3. 2011

Bc. Vlastimil Flegl

Poděkování

Na tomto místě bych chtěl poděkovat svému vedoucímu diplomové práce, panu prof. Ing. Vladimíru Olejovi, CSc. za vedení, podporu a poskytnuté odborné rady při psaní diplomové práce.

ANOTACE

Diplomová práce se zabývá modelováním predikce návštěvnosti Web domény pomocí SVM neuronových sítí. Jako data pro modelování je použitá naměřená návštěvnost webových stránek Univerzity Pardubice. Součástí práce je charakteristika Web miningu a SVM neuronových sítí. V této práci jsou navrženy optimální modely a zhodnocení dosažených výsledků.

KLÍČOVÁ SLOVA

Support Vector Machine, Support Vector Regression, Web Mining, Predikce, Modelování

TITLE

Modelling of Time Series Prediction Visit of Web Domain by SVM Neural Networks

ANNOTATION

The graduation theses focuses on modelling of prediction visit of Web domain by SVM neural networks. As data for modelling is used measuring visit rate of University Pardubice´s web pages. Part of this work is characterization of Web mining and SVM neural networks. In this work are designed optimal models and evaluation of achieved results.

KEYWORDS

Support Vector Machine, Support Vector Regression, Web Mining, Prediction, Modelling

Obsah

Úvod

Web mining je skupina metod a nástrojů, poskytující přínosy v podobě snadnějšího a přesnějšího vyhledávání informací pro návštěvníky internetu. Především ale poskytuje mnoho důležitých využitelných údajů o návštěvnících pro majitele webových stránek. Správně cílený Web mining a následné využití jím poskytnutých údajů může znamenat konkurenční výhodu v internetovém byznysu.

V první části práce jsou popsány přínosy Web miningu pro běžné webové stránky, internetové obchody i vyhledávače. Web mining je rozdělen na jednotlivé části a každá část je popsána z hlediska principu a využití.

Diplomová práce popisuje predikci návštěvnosti webových stránek Univerzity Pardubice. Návštěvnost Webu je jeden ze základních údajů zkoumaných pomocí Web miningových technik. Predikce je prováděna pomocí metody Support Vector Machine, což je metoda blízká neuronovým sítím, ale má své specifické vlastnosti a princip práce.

V druhé části této práce je charakterizována metoda Support Vector Machine, její základy a princip. Tato metoda je zde popsána z hlediska klasifikace i regrese. V této části je dále vysvětlen pojem jádrová funkce, příklady jádrových funkcí a jejich souvislost s metodou Support Vector Machine.

Další části práce se věnují charakteristice dat pro modelování, jejich předzpracování a použití při procesu modelování. Následně je popsán postup při modelování a jsou analyzovány získané výsledky.

Cílem této práce je nalezení optimálních matematických modelů, které slouží k predikci návštěvnosti Web domény Univerzity Pardubice. Predikce je provedena pro různě dlouhé časové řady. Cílem je také experimentálně nalezené modely popsat, dále na základě těchto nalezených modelů popsat souvislosti a poznatky nabyté při modelování a zhodnotit úspěšnost predikce.

1 Web Mining

1.1 Základní charakteristika Data miningu

Data miningem se nazývá objevování znalostí v databázích [1]. Český ekvivalent pro tento pojem je dolování z dat. Jinak řečeno, je to dolování z dat za účelem získání dosud neznámých informací a souvislostí. Data mining obvykle začíná pochopením dat, pomocí datových analýz. Ty provádějí datoví analyzátoři. Dolování dat je obvykle prováděno ve třech hlavních krocích:

- Předzpracování: Nezpracovaná data nejsou obvykle vhodná pro dolování z různých důvodů: Je třeba očištění a odstranění šumu, odstranění abnormalit a odlehlých hodnot, data můžou být také příliš obsáhlá, s nepotřebnými atributy, apod.
- Data mining: Zpracovaná data jsou připravena k aplikaci Data miningových algoritmů, které budou produkovat výsledky a znalosti
- Následné zpracování: V tomto kroku se roztřídí data a provádí se vizualizace výsledků. Pro různé aplikace a použití v praxi se vybírají příslušná vhodná data.

Data mining je hlavně používán v oblasti firem se zaměřením na spotřebitele, ve finančních, komunikačních a marketingových organizacích. To umožňuje těmto firmám stanovit vztahy mezi faktory jako je cena, umístění produktu, ekonomické ukazatele, demografické údaje zákazníka a další. Data miningové techniky mají pozitivní vliv na prodej, zisky a spokojenost zákazníků. Mezi používané nástroje Data miningu patří umělé neuronové sítě, rozhodovací stromy, genetické algoritmy, shlukování pomocí různých metod, IF-THEN pravidla a v neposlední řadě také vhodná vizualizace dat. Tyto nástroje pracují s jedním nebo svíce vztahy. Mezi základní čtyři typy vztahů patří: Třídy, shluky, asociace a sekvenční data (vzory).

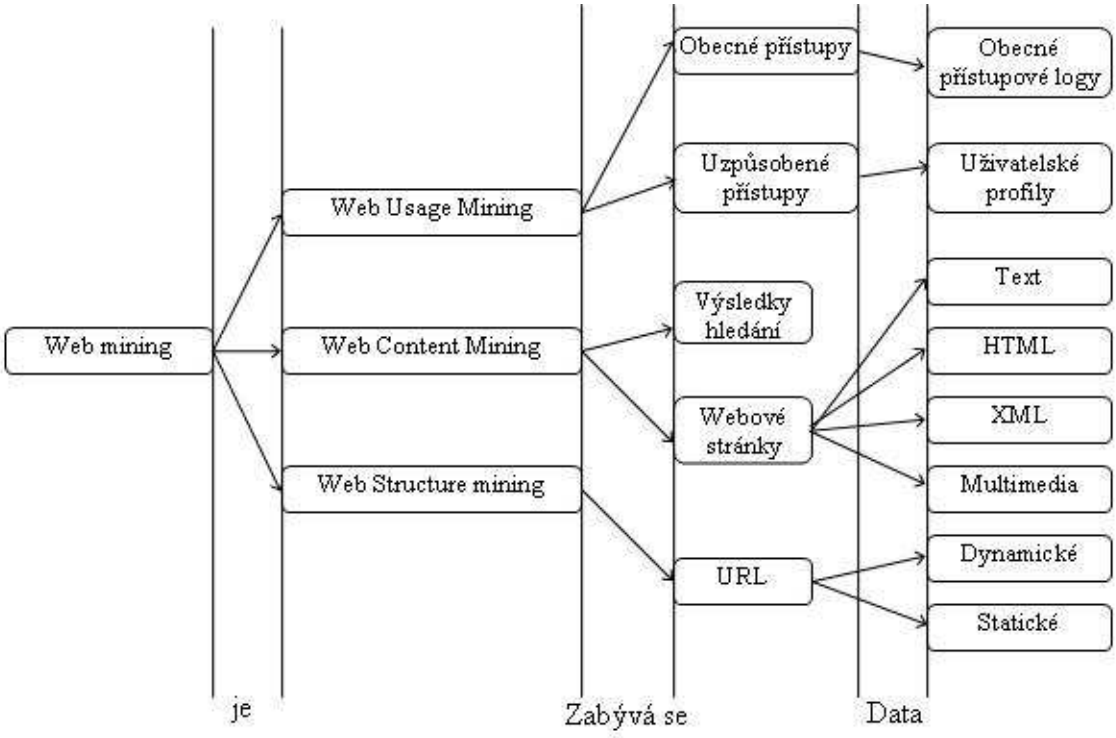
Text mining a Web mining jsou další dva podobné pojmy používané spolu s Data miningem. Tato práce v další části přiblíží druhý z těchto pojmů.

1.2 Základní charakteristika Web miningu

Se zvyšujícím se množstvím informací a informačních zdrojů přístupných on-line na webu je stále potřebnější tyto informace nějakým způsobem získat a využívat.

S tím také stoupají nároky na nástroje pro sledování těchto dat, jejich automatické sbírání, analyzování a případné další možnosti využití. To vede k potřebě vytvořit inteligentní systémy na straně serveru i klienta.

Web mining si klade za cíl nalézt užitečné informace a znalosti z webové struktury, z obsahu stránek a z používaných dat. Web mining využívá mnoho Data miningových technik, ale není pouze aplikací tradičního Data miningu a to kvůli různorodosti a polo-strukturované nebo nestrukturované povaze webových dat. Rozdíl mezi Data miningovým a Web miningovým procesem je obvykle ve sběru dat. V tradičním Data miningu jsou často data již sesbíraná a skladovaná v datovém úložišti. Naopak Web mining počítá jako se značnou částí svých úloh s různým způsobem sběru dat. Na sesbíraná data jsou aplikovány stejné tři kroky jaké byly uvedeny v předchozí kapitole o Data miningu. Techniky se však můžou pro každý krok oproti Data miningu lišit. Na obr.1 je znázorněn koncept Web miningu.



Obr.1: Konceptuální mapa Web miningu [4]

Web mining je dobrým nástrojem především pro majitele internetových obchodů. Analyzuje pohyb zákazníků či návštěvníků, dobu strávenou na jednotlivých stránkách, místa příchodů a odchodů, sekvence průchodů stránkami, apod. Díky těmto informacím lze zjistit mnoho užitečných informací o zákaznících a jejich preferencích.

Je však přínosem i pro běžné uživatele internetu především v oblasti vyhledávání. Web mining se dělí na tři následující části:

- Web Structure Mining (WSM) neboli zkoumání struktury webových stránek
- Web Content Mining (WCM) neboli získávání informací z obsahu webu
- Web Usage Mining (WUM) neboli zkoumání chování uživatelů

O každé z těchto tří skupin pojednává v této práci samostatná kapitola.

1.3 Web Structure Mining

První skupina Web miningu je Web structure mining. Tento typ Web miningu je založen na uspořádání a analyzování spojení a vztahů mezi webovými stránkami. Typickým výstupem jsou informace o vzájemném propojení množin webových stránek nebo jejich adres. Rozlišují se dva odlišné přístupy a to Link Topology Mining a Link URL Mining. Oba používají odlišná data a metody. V případě přístupu Link Topology Mining je zacházeno s webem jako s grafem, stránky potom představují uzly grafu a odkazy představují hrany nebo oblouky. Tento přístup pracuje nezávisle na informacích o obsahu stránek. Pro Link URL Mining představuje topologie jednotlivé URL zdroje a cíle každé stránky, což dovoluje přístup ke konkrétnějším adresovým datům. Oba tyto typy můžou být použity samostatně nebo spolu s jinými přístupy k řešení problémů Web miningu (např. identifikace témat webu).

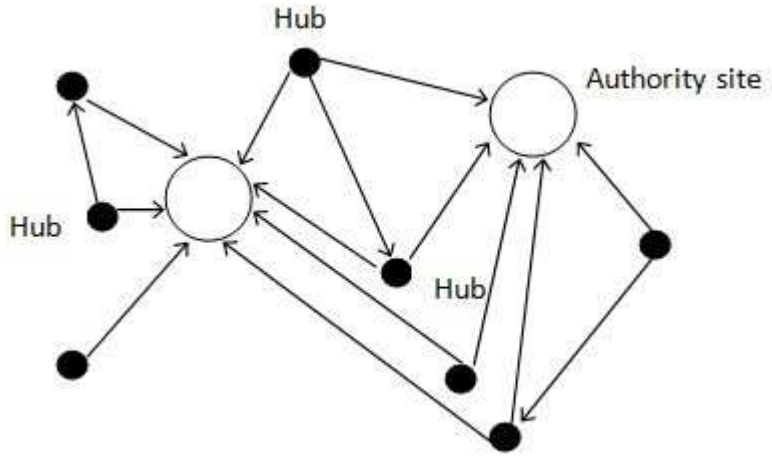
Jak je patrno z obr.1, stránky můžou být statické nebo můžou být generovány dynamicky. V případě dynamických stránek je jejich mapování komplikovanější, protože dynamické prostředí je zatíženo potřebou více technik.

V roce 1998 byly navrženy dva nejdůležitější vyhledávací algoritmy založené na hypertextových odkazech: PageRank a HITS (Hypertext Induced Topic Search) [6]. Využívají strukturu hypertextových odkazů Webu k hodnocení stránek podle jejich stupně prestiže nebo autority. Autorita je určitá tematická stránka s velkým množstvím referencí (hypertextových odkazů směrovaných na ni).

Algoritmus PageRank je založen na míře hodnocení prestiže. Hodnota PageRank je vypočítávána pro každou indexovanou off-line stránku. Hlavní koncept tohoto algoritmu vypadá následovně:

- Hypertextový odkaz ze stránky odkazující na jinou stránku je implicitní přenos autority k cílové stránce. Jinak řečeno čím více odkazů na příslušnou webovou stránku odkazuje, tím je důležitější.
- Hypertextový odkaz z prestižní stránky je důležitější než odkaz z méně prestižní stránky. Z toho vyplývá, že stránka je důležitá, když je na ní odkazováno z jiných důležitých stránek.

Algoritmus HITS má základní myšlenku v tom, že pokud má stránka dobrý a věrohodný obsah na nějaké téma, tak ji důvěřuje hodně lidí a odkazují se na ni. Hub (centrální stránka) se nazývá stránka s velkým množstvím odkazů na cizí stránky. Pod pojmem Hub si lze představit např. katalog stránek. Uživatel, který navštíví tuto Hub stránku nalezne užitečné odkazy na stránky různých tématik. Klíčová myšlenka je, že dobré Hub stránky zaměřují dobré autority a naopak dobré autority jsou zaměřovány mnoha dobrými Hub stránkami. Tuto myšlenku ilustruje obr.2.

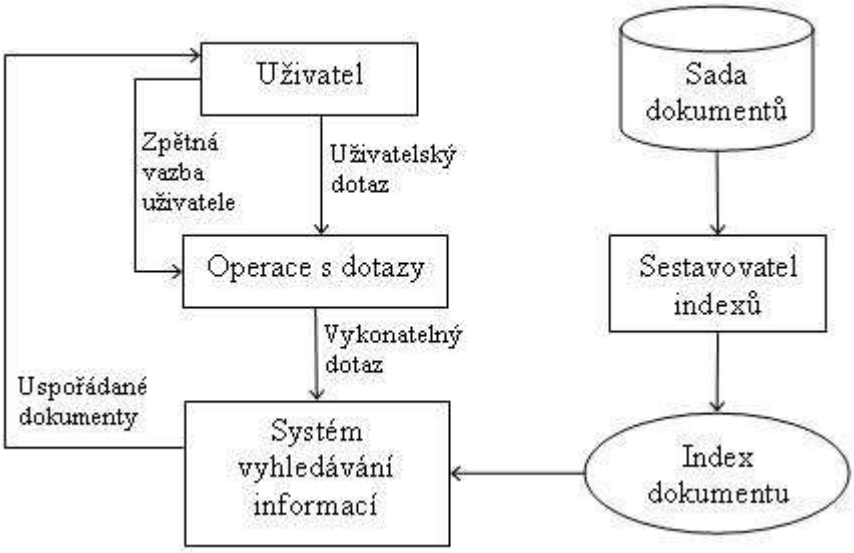


Obr.2: Struktura Hubů a Autorit [7]

1.4 Web Content Mining

Web content mining je druhou skupinou Web miningu. Tato část Web miningu má svým principem nejblíže k tradičnímu Data miningu. Myšlenkou je zde hledání informací prostřednictvím klíčových slov, což může být jedno slovo nebo víceslovné spojení. Klíčovým slovům se říká termy. Hledání na webu je zakořeněno v IR modelu (Information Retrieval, neboli model získávání informací) [1]. Problémem je orientace v obrovském množství dokumentů (webových stránek) na internetu a nalezení nevhodnějších z nich pro konkrétní dotaz uživatele. Důležité je také rozlišení různých

částí dokumentu. Některé části obsahující požadované termy jsou nevýznamné (např. reklama na stránkách), jiné mají vyšší význam. Uvažuje se zde nejen samotný text webové stránky, ale i např. hlavička, titulek a další textové části. Problémem můžou být také slova, která mají v daném jazyce více významů. Toto všechno je potřeba zohlednit při relevantní odezvě výsledků hledání. Na obr.3 je znázorněna architektura IR modelu.



Obr.3: Architektura IR modelu [1]

Výše zmíněný IR model řídí, jak jsou dokumenty a dotazy reprezentované a jaká je relevance dokumentu na uživatelský dotaz [1]. Jsou čtyři hlavní IR modely: Boolean model, model vektorového prostoru (vector space model), jazykový model a pravděpodobnostní model.

Nejznámější a nejrozšířenější je model vektorového prostoru. Tento model představuje reprezentaci množiny dokumentů jako vektorů ve společném vektorovém prostoru [6]. Myšlenka je přiřazení váhy ke každému termu *t* v dokumentu *d*. Takto se spočítá pro každý dokument skóre pro určitý dotaz obsahující daný term. Váha určitého termu *t* v dokumentu *d* závisí buď pouze na počtu výskytů termu v dokumentu nebo se bere ohled na významnost termu. Tomuto váhovému schématu se říká četnost termu a značí se $t\ f_{t,d}$ [6]. V případě, že se bere v úvahu významnost termu, se používá zmírnění efektu termů, které se často v souborech vyskytují pomocí redukce vah. Redukování váhy termu může být provedeno pomocí faktoru, který roste s frekvencí výskytu [6]. Redukce váhy by měla pomoci k relevantnějším výsledkům hledání.

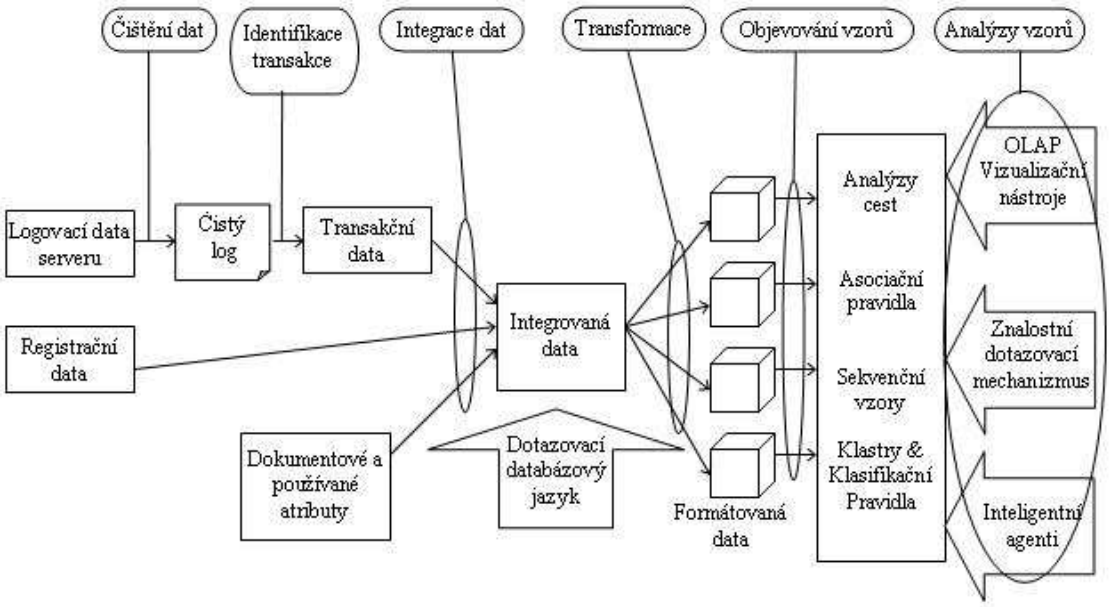
Boolean model uvažuje pouze s tím, zda se daný term v dokumentu vyskytuje nebo nevyskytuje. Výsledná váha termu je v tomto případě pouze 0 nebo 1. Jazykový model je založen na pravděpodobnosti s využitím Bayesovských pravidel.

1.5 Web Usage Mining

Poslední část Web miningu je Web usage mining. Tato část se zabývá chováním uživatele na webu. Zajímá se především o to, odkud uživatel přišel na danou stránku, jak dlouho se tam zdržel, jakou stránku navštívil jako další, zda-li je to jeho první návštěva nebo navštěvuje Web často, v jakou dobu navštěvuje dané stránky, apod. Samozřejmě se zaznamenává IP adresa, ale i další údaje o uživateli jako je jeho operační systém nebo typ a verze prohlížeče.

Tato část Web miningu je tak užitečná hlavně pro internetové obchody případně komerční weby, kde je potřeba zjistit, které produkty zákazníky zajímají, z jakých stránek zákazníci odcházejí a z toho vyvodit další marketingové změny nebo přímo reorganizaci struktury webu. Výhodou zde může být přizpůsobení webu směrem k pohodlí zákazníků, nevýhodou může být do jisté míry ztráta soukromí zákazníků (uživatelů). Architektura Web usage miningu je zobrazena na obr.4. Web usage mining lze rozdělit na dvě základní části:

- Předzpracování dat
- Využití předzpracovaných dat (objevování znalostí a souvislostí v datech)



Obr.4: Architektura Web usage miningu [2]

1.5.1 Předzpracování dat pro Web mining

Pod pojmem předzpracování dat si lze představit přípravu dat pro další použití zahrnující: Čištění dat (výběr užitečných použitelných dat a vyřazení nezajímavých dat), integraci dat z různých zdrojů a transformaci takto integrovaných dat do formy vhodné pro další Data miningové postupy. Vše je názorně patrné z obr.4, kde levá část až po naformátovaná data zahrnuje předzpracování a pravá část využití dat. Proces přípravy dat je často časově a výpočetně nejnáročnější krok v procesu Web usage mining [1]. Je zároveň klíčový při správném výběru užitečných dat.

Data pro předzpracování jsou získávána z různých datových zdrojů. Asi nejčastějším zdrojem jsou logovací soubory (logy) umístěné na straně serveru. Tyto soubory můžou mít různou strukturu a obsahovat různé informace (časové informace a chování uživatele nebo zákazníka zmíněné výše, reference z jiné stránky, apod.). Jejich nevýhodou může být u frekventovanějších serverů zbytečné zatížení důsledkem velkého množství zaznamenávaných dat. V souvislosti s přístupem na server se používají dva základní pojmy:

- Page View (shlédnutá stránka): Reprezentuje jednu uživatelskou událost, např. otevření stránky, přidání produktu do košíku, přihlášení uživatele apod. I každé znovunačtení stejné stránky je chápáno jako nové Page View.
- Session nebo jinak Server Session je posloupnost Page View událostí a reprezentuje jednu unikátní návštěvu webu se všemi kroky uživatele.

Mezi další zdroje dat patří tzv. cookies na straně klienta. Jsou to soubory, ve kterých jsou uchovávány údaje z navštívených stránek, resp. weby do nich zapisují své údaje. Pomocí nich pak lze zjistit unikátnost návštěvy uživatele na daném webu. Cookies můžou být na straně klienta povolené nebo zakázané. Data je možné získat i z jiných externích zdrojů, např. z demografických nebo z tzv. clickstreamových zdrojů. Pojem clickstream znamená posloupnost uživatelem otevřených odkazů, neboli cestu uživatele Webovým prostředím. Využívají se i např. meta-data nebo doménové znalosti. Při předzpracování dat se typicky vyskytují problémy jako [8]:

- Jedna IP adresa, více přístupů na server: Poskytovatelé internetu mají většinou společné proxy servery a jeden proxy server tak může obsluhovat několik uživatelů přistupující na stejnou stránku v ten samý čas.

- Více IP adres, jeden serverový přístup: Opačný případ, kdy se při požadavku stejného uživatele přiděluje pokaždé jiná náhodná IP adresa.
- Více IP adres, jeden uživatel: Jeden uživatel může přistupovat na Web z odlišných počítačů a tudíž pod odlišnými IP adresami.
- Více prohlížečů, jeden uživatel: Případ kdy uživatel používá na stejném počítači více prohlížečů. Přístup z různých prohlížečů je pak brán jako přístup různých uživatelů.

Tyto problémy jsou potom především příčinami sporů, kolik přesně uživatelů (nebo zákazníků) na Web přistupuje nebo jak dlouho se uživatel na stránce zdrží. To lze částečně řešit registrací uživatele na Webu, což zaručuje jednoznačnost každého přihlášeného uživatele. Na Webech se registrují většinou stálejší zákazníci nebo návštěvníci, což vede k získání jejich podrobnějších a přehlednějších údajů. Nelze však zaručit, že se bude ten samý zákazník pohybovat na Webu pouze pod svým uživatelským účtem. Jednou se může pohybovat po Webu jako přihlášený, podruhé jako nepřihlášený. Z tohoto důvodu nelze zjistit o určitém registrovaném zákazníkovi přesné údaje o počtu návštěv, počtu shlédnutých stránek, apod.

Každý majitel stránek, sledující návštěvnost musí brát v potaz také roboty, kteří znepřesňují skutečný počet návštěvníků Webu. Robota lze poznat např. podle času stráveného na stránce. Ten bývá výrazně kratší než v případě skutečného návštěvníka. Robot dokáže také projít větší množství stránek v malém časovém intervalu. Na základě těchto skutečností lze rozeznat skutečné přístupy na Web oproti přístupům robota a pravé přístupy pak vyfiltrovat.

1.5.2 Využití předzpracovaných dat

K využití získaných dat pro konkrétní potřebu se používá jejich zpracování pomocí různých metod a technik. Některé metody, převzaté z jiných oblastí musí brát v úvahu strukturu Webových dat. Existuje několik základních metod: Statistické analýzy, asociační pravidla (asociace a korelace), shlukování, klasifikace, případně sekvenční analýza a modelování závislostí.

Mezi metody statistické analýzy patří zjišťování hodnot stejných jako v klasické popisné statistice. Z předzpracovaných dat (např. údaje z logů) se určuje průměr, modus, četnost, maximum, minimum a další hodnoty dle konkrétní potřeby. Například

tak lze z dat určit nejčastěji otevírané stránky, průměrnou dobu zákazníka strávenou u nějakého produktu v internetovém obchodě, apod. Využití statistické analýzy může být např. v posílení konektivity nejvytíženějších částí Webu nebo v umístění specifické nabídky produktů na hlavní stránce internetového obchodu.

Asociační pravidla jsou metody zjišťující souvislost událostí uskutečněných ve Webovém prostředí. Výsledkem může být např. zjištění, že uživatel prohlížející si oddělení internetového obchodu s mobilními telefony si v rámci jedné unikátní návštěvy prohlíží taky oddělení obchodu s počítači. Používají se také korelace např. ke zjištění vztahu uživatelů navštěvujících určitý typ stránek k uživatelům navštěvujících tématicky jiný typ stránek. Příkladem asociačního pravidla je tvrzení: „Pokud zákazník navštíví oddělení obchodu s mobilními telefony, tak navštíví i oddělení obchodu s počítači“. K asociačním pravidlům se vztahují dva pojmy: Podpora pravidla a spolehlivost pravidla. Na výše uvedeném případu je podpora pravidla pravděpodobnost, že zákazník navštíví obě zmíněné oddělení obchodu, vzhledem ke všem návštěvám obchodu. Spolehlivost pravidla je na výše uvedeném příkladu pravděpodobnost, že zákazník navštíví oddělení obchodu s počítači vzhledem ke všem návštěvám v oddělení s mobilními telefony. Využití asociačních pravidel je hlavně pro marketingové účely (rozmístění reklamy) nebo pro vhodnou strukturu Webu (např. odkazy na části Webu které zákazníci nejčastěji navštěvují umístěné blízko sebe). Asociační pravidla můžou také sloužit jako heuristika pro předběžné načítání dokumentů za účelem snížit uživateli dobu čekání při načítání stránky z jiného vzdáleného místa [8].

Shlukování (Clustering) je metoda seskupování objektů se společnými charakteristikami. Jedná se o učení bez učitele a seskupování objektů probíhá pouze na základě vypočítané podobnosti. V oblasti Web usage mining se rozlišují dva typy shlukování:

- Shlukování uživatelů (vytváření skupin uživatelů, kteří se chovají na internetu podobně, navštěvují podobné typy stránek, nakupují podobné produkty, ...)
- Shlukování stránek (vytváření skupin stránek s podobným obsahem) může být přínosem pro internetové vyhledávače.

Klasifikace je na rozdíl od segmentace tzv. učení s učitelem. Při této metodě se objekty umísťují do předem definovaných tříd. Každá třída je charakteristická svými

vlastnostmi, které musí objekt spadající do této třídy splňovat. Mezi nejznámější klasifikátory patří neuronové sítě a rozhodovací stromy.

Cyklus, který začíná sběrem dat, pokračuje jejich předzpracováním a následným zpracováním pomocí výše uvedených metod končí analýzou výsledků a jejich využitím v praxi. Pro reálné využití získaných informací z některých z výše uvedených metod je potřeba výsledky správně pochopit, případně graficky znázornit. Analýza výsledků vede k rozhodnutí o provedení dalších opatření a změn. Správné využití výsledků v praxi může vést jak k příjemnější a intuitivnější práci uživatele na Webu, tak i ke konkurenční výhodě na straně firmy. S měnící se ekonomickou situací, preferencemi uživatelů nebo s variabilitou sociálních skupin uživatelů využívajících různé části internetu bude cyklus Web usage miningu stále aktuální otázkou.

2 Neuronové sítě a Support Vector Machines

2.1 Základní charakteristika neuronových sítí

Myšlenka a struktura neuronových sítí vychází z tendence napodobování činnosti lidského mozku. Stejně jako jsou v mozku hlavními zpracovateli informací nervové buňky (neurony), tak je i v neuronových sítích základním prvkem *neuron*. Neurony mají vstup nebo více vstupů a výstup nebo více výstupů sloužících k přenosu informace. Kromě výstupních neuronů jsou výstupy každého neuronu vstupem do jiných neuronů. Každému spojení mezi dvěma neurony se říká *synapse*. Ty jsou ohodnocené váhovými koeficienty. Formálně je neuronová síť pojatá jako orientovaný graf [9]. Skládá se z jedné vstupní vrstvy, jedné nebo více skrytých vrstev a jedné výstupní vrstvy. Hlavní předností neuronových sítí je schopnost učit se. Dále také schopnost generalizace (zevšeobecnění) a práce s daty, které obsahují šum. Zevšeobecnění znamená, že síť správně reaguje výstupní hodnotou na nová, neznámá vstupní data.

Neuronové sítě nemohou plnohodnotně napodobit funkci lidského mozku už minimálně z jedno základního důvodu. Lidský mozek obsahuje obrovské množství nervových buněk, kde jeden neuron má stovky, tisíce a někdy až několik desítek tisíc synapsí [9]. I přesto je neuronová síť meotdou s velkým potenciálem.

Neuronové sítě se používají pro dvě základní činnosti: Klasifikaci a predikci. Klasifikace znamená zařazení objektů do předem definovaných tříd. Predikce znamená předpověď budoucího vývoje nějakého ukazatele na základě jeho dosavadního vývoje. Často se používá predikce pro budoucí vývoj časových řad. Tyto dvě metody spadají do tzv. kontrolovaného učení, nebo-li učení s učitelem. Učení s učitelem je charakteristické tím, že se neuronová síť “naučí” na nějakých známých datech (trénovací množina) a následně je schopna predikovat nové hodnoty (v případě predikce) nebo zařazovat správně objekty do předem definovaných tříd (v případě klasifikace). Neuronová síť se může naučit správně klasifikovat nebo predikovat s určitou přesností, může být také špatně naučená, pokud je přesnost neuspokojivá nebo může být tzv. přeučená. Pokud je neuronová síť přeučená, pak se naučila trénovací množinu dat “nazpaměť” a bude mít problém při klasifikaci nebo predikci nových neznámých dat. Druhou skupinou je tzv.

nekontrolované učení nebo-li učení bez učitele, kde jsou objekty řazeny do tříd, které nebyly uživatelem definovány, ale metoda si je zvolila sama (na základě podobnosti objektů). Zástupcem nekontrolovaného učení jsou např. Kohonenovy (samoorganizující-se) mapy.

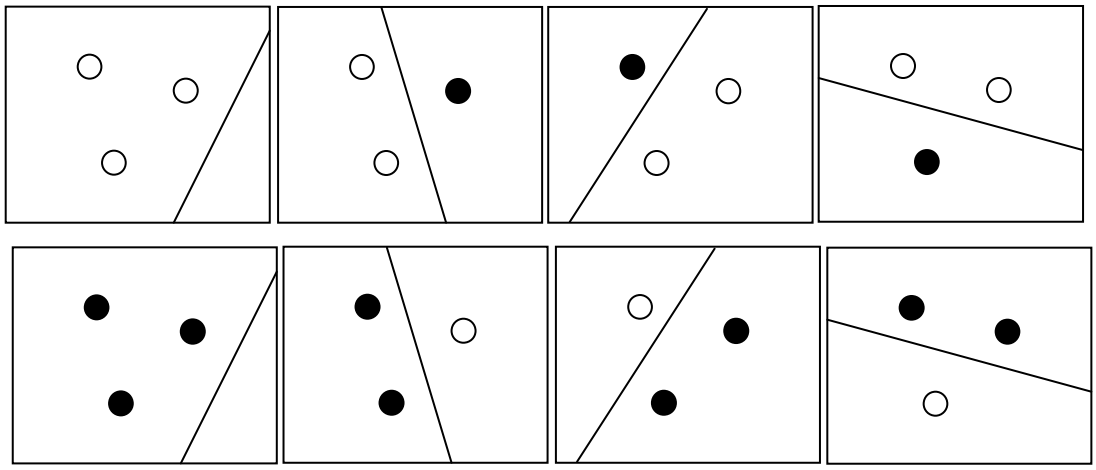
2.2 Support Vector Machine

2.2.1 VC dimenze

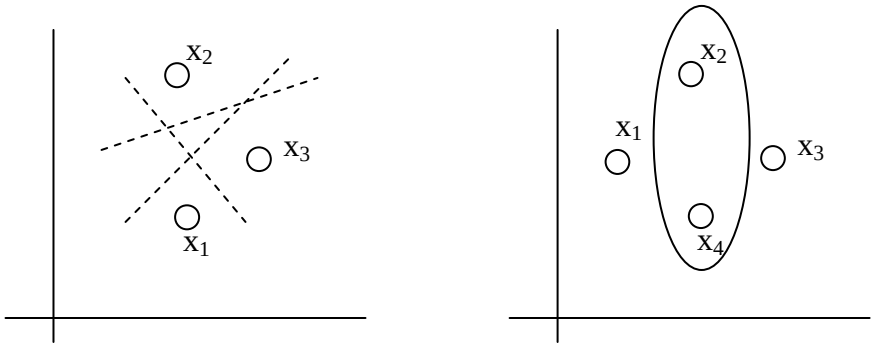
VC dimenze (Vapnik Chervonenkis dimenze) je pojem využívaný v metodě Support Vector Machines. Definice VC dimenze množiny indikačních funkcí je následující:

„VC dimenze množiny indikačních funkcí $F(x,w), w \in W$ *je maximální počet h, bodů* $x_1, ..., x_h$*, které můžou být separovány do dvou tříd všemi možnými způsoby* 2^h*. Jinak řečeno, je to maximální počet bodů, které můžou být funkcí rozděleny [14].“*

Indikační funkce je funkce, která rozděluje danou množinu bodů na dvě podmnožiny 0 a 1. Na obr.5 je ukázáno 2^3 = 8 možností rozdělení tří bodů v dvourozměrném prostoru do dvou tříd. VC dimenze dvourozměrného prostoru při separaci pomocí přímky je 3. Na obr.6 je názorně zobrazeno, že v případě čtyř bodů již nelze klasifikovat pomocí přímky body x_1 a x_3 do jedné třídy a body x_2 a x_4 do třídy druhé. Více viz. [16].



Obr.5: Rozdělení tří bodů v dvourozměrném prostoru [14]



Obr.6: Tři a čtyři body v dvourozměrném prostoru [16]

2.2.2 Minimalizace Strukturálního Rizika

Minimalizace Strukturálního rizika poskytuje kompromis mezi VC dimenzí aproximačních funkcí a kvalitou vzorku tréninkových dat (empirickou chybou). Postup je přibližně následující [17]:

- Volba třídy funkcí jako např. neuronové sítě s *n* skrytými vrstvami neuronů.
- Rozdělení třídy funkcí na hierarchii vnořených podmnožin s rostoucí složitostí.
- Provést minimalizaci empirického rizika pro každou podmnožinu.
- Vybrat model, který vykazuje minimální součet empirických rizik a minimální „VC confidence“.

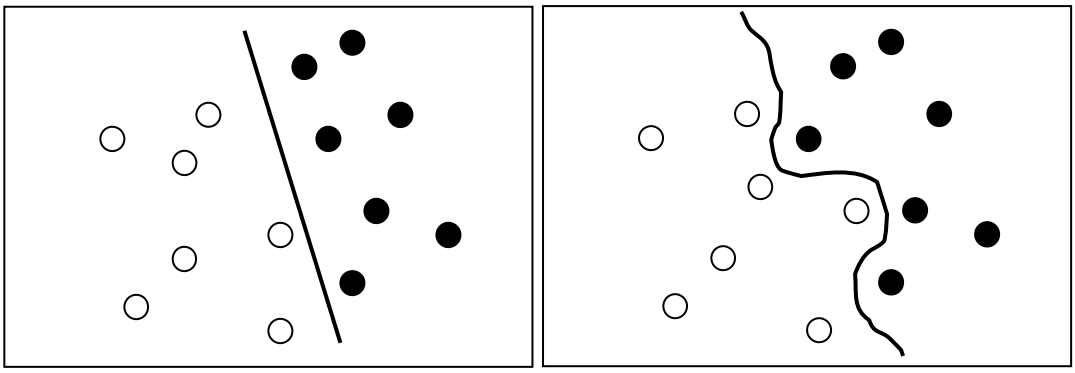
2.2.3 Charakteristika metody Support Vector Machine

Support Vector Machines (dále jen SVM) v překladu znamená algoritmy podpůrných vektorů. Jsou to metody velmi blízké neuronovým sítím. Spadají do skupiny nazývané jádrové algoritmy (kernel machines). Jedná se o metody s učením s učitelem. První zmínka o SVM byla ve (Vapnik, 1979), ale první hlavní písemný dokument byl napsán v roce 1995 [12].

Základní původní využití SVM je binární klasifikace (rozdělení datových vzorů do dvou tříd). Později byla metoda SVM rozšířena i k řešení regresních úloh. Klasické jednovrstvé neuronové sítě mají schopnost pouze lineárního oddělování dat pomocí přímek nebo rovin a u vícevrstvých sítí je riziko při nelineární funkci uvíznutí v jejím lokálním minimu, což zhoršuje náročnost učení. Metody SVM řeší tyto problémy svým specifickým způsobem.

„*Snaží se využít výhody poskytované efektivními algoritmy pro nalezení lineární hranice a zároveň jsou schopny reprezentovat vysoce složité nelineární funkce [11].*“

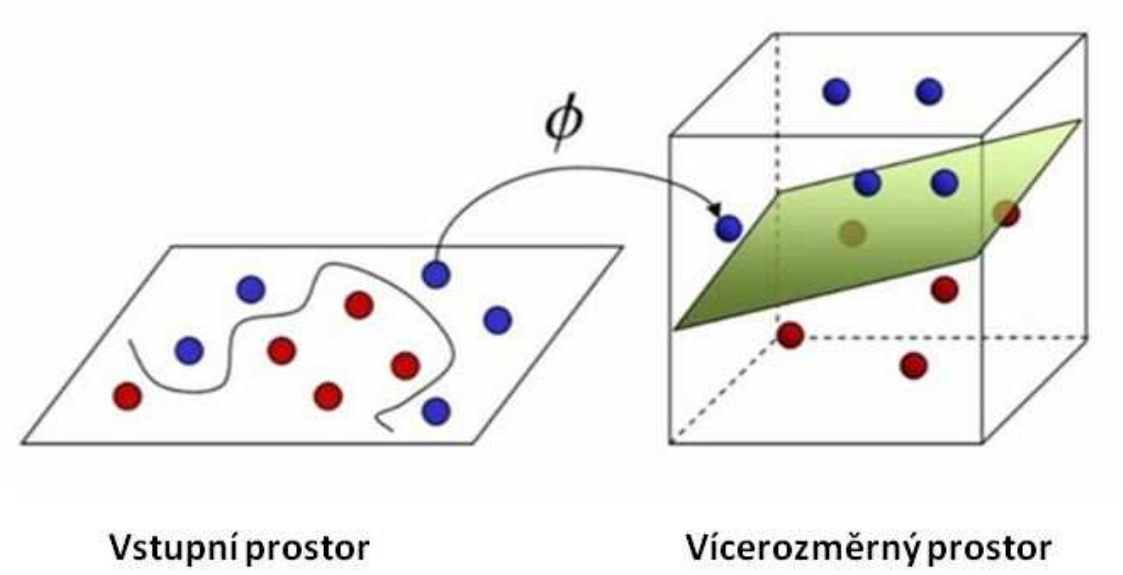
Základní myšlenkou SVM je vytvořit nadrovinu, která je rozhodovací rovinou a to takovým způsobem, že se maximalizuje hranice separace mezi dvěma třídami datových vzorů. Jinak řečeno je potřeba dosáhnout toho, aby datové vzory obou tříd, které jsou k sobě nejblíže měly maximální vzdálenost od hranice separace (maximalizování minimální vzdálenosti). K problému lineárního oddělení nelineárně oddělitelných datových vzorů se využívá převodu dat z klasického vstupního prostoru do charakteristického vícerozměrného prostoru (feature space), v němž je možné lineární separaci dat provést. Na obr.7 je v levé části zobrazen případ lineárně oddělitelných dat, v pravé části je případ lineárně neoddělitelných dat.



Obr.7: Lineárně separovatelná a lineárně neseparovatelná data

Problém je v tom, že v reálu jsou data často lineárně neoddělitelná. Převod do charakteristického vícerozměrného prostoru názorně zobrazuje obr.8. Jedná se o nejjednodušší případ, kdy byly datové vzory převedeny z dvourozměrného prostoru do třírozměrného prostoru.

„Obecně platí, že N datových bodů je možno vždy (kromě některých speciálních případů) lineárně oddělit v prostoru s N-1 nebo více dimenzemi [11].“



Obr.8: Transformace ze vstupního do vícerozměrného prostoru a způsob oddělení dat nadrovinou ve vícerozměrném prostrou [13]

Ke splnění maximalizace separační hranice mezi dvěma třídami SVM využívá přístupu založeném na teorii statistického učení (viz. [10]). SVM je přibližná implementace metody minimalizace strukturálního rizika. Tento princip je založen na tom, že chyba učení SVM na testovacích datech je ohraničena součtem tréninkových chyb na termu, který závisí na VC (Vapnik-Chervonenkis) dimenzi. Pokud jsou vzorová data oddělená, SVM produkuje nulovou hodnotu pro první term a zároveň minimalizuje druhý term [10].

Název Support Vector, nebo-li podpůrný vektor vzniknul z toho, že datové vzory na každé straně oddělovací roviny, které jsou této rovině nejblíže tvoří právě podpůrné vektory, viz. další části práce. Datových vzorů tvořících podpůrný vektor bývá výrazně méně než celkový počet vstupních vzorů a jsou vybrány algoritmem. Hlavní myšlenka k tvorbě učícího algoritmu SVM je vnitřní produkt jádra mezi podpůrným vektorem x_i a vstupním vektorem x. Každá učící metoda (polynomická, RBF, ...) má své charakteristické nelineární rozhodovací roviny. Podle toho jak je jádrová funkce generovaná, lze sestrojit příslušné odlišné učící metody. Důležitou vlastností SVM je dobrá schopnost zevšeobecnění (i pro menší počet trénovacích dat než u běžné NS) a díky tomu i široká použitelnost.

2.3 Optimální nadrovina pro lineárně separovatelné datové vzory

Na úvod je nutno zmínit, že informace z odborné problematiky SVM byly v této práci čerpány především z literatury [10].

Je daný vzor trénovacích dat $\{(\mathrm{x}_i, d_i)\}_{i=1}^{N}$, kde x_i je vstupní datový vzor pro i-tý příklad a d_i je odpovídající požadovaná odezva (výstupní hodnota), přičemž $d_i \in \{-1,1\}$. Je nutno předpokládat, že vzor reprezentovaný podmnožinou $d_i = +1$ (pozitivní vzor) a vzor reprezentovaný podmnožinou $d_i = -1$ (negativní vzor) jsou "lineárně separovatelné". Rovnice rozhodovací roviny provádějící separaci má tvar

$$\mathrm{w}^{\mathrm{T}}\mathrm{x} + \mathrm{b} = 0, \tag{2.1}$$

kde x je vstupní vektor, w je regulovatelný váhový vektor a b je bias. Lze tedy psát

$$w^T x_i + b \geq 0 \quad pro\ d_i = +1,$$

$$w^T x_i + b < 0 \quad pro\ d_i = -1. \tag{2.2}$$

Separace mezi nadrovinou definovanou v (2.1) a nejbližším datovým vzorem pro daný váhový vektor w a bias b je nazývána rozpětí separace (margin) a označuje se ρ. Jak bylo napsáno výše, cílem je najít maximální hodnotu ρ. Pokud je nalezená hodnota ρ maximální, potom je rozhodovací rovina nazývána optimální. Optimální nadrovina pro lineárně separovatelné vzory je znázorněna na obr. 9. Odděluje v levé části negativní vzory dat a v pravé části pozitivní vzory dat.

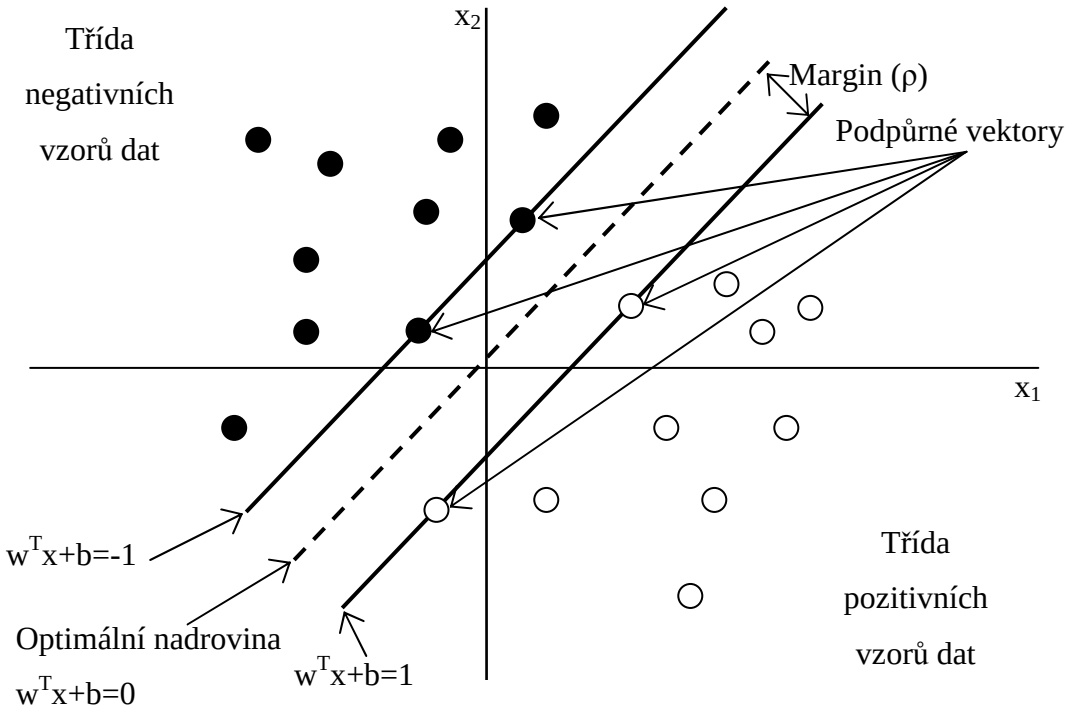
Nechť w_0 a b_0 označují optimální hodnoty váhového vektoru w a biasu b. Odpovídající optimální nadrovina reprezentující vícerozměrnou lineární rozhodovací rovinu ve vstupním prostoru je definována jako [10]

$$w_0^T x + b_0 = 0, \tag{2.3}$$

což je přepsaná rovnice (2.1). Diskriminační funkce

$$g(x) = w_0^T x + b_0 \tag{2.4}$$

dává algebraický rozsah vzdálenosti z x k optimální nadrovině.



Obr.9: Optimální nadrovina pro lineárně separovatelné datové vzory [10]

Pro jednoduché zobrazení lze vyjádřit *x* takto

$$x = x_p + r\frac{w_0}{\|w_0\|}, \qquad (2.5)$$

kde x_p je normální zobrazení *x* na optimální nadrovinu a *r* je požadovaná algebraická vzdálenost. Když je *x* na kladné straně optimální nadroviny tak je *r* kladné a když je *x* na záporné straně tak je *r* záporné. Z definice $g(x_p)=0$ vyplívá, že

$$g(x) = w_0^T x + b_0 = r\|w_0\| \qquad (2.6)$$

nebo

$$r = \frac{g(x)}{\|w_0\|}. \qquad (2.7)$$

Když $b_0>0$, vzor je na kladné straně optimální nadroviny. Když $b_0<0$, tak je na záporné straně. Když $b_0=0$, optimální nadrovina prochází skrz vzory [10].

Je daná trénovací množina $\tau = \{(x_i, d_i)\}$ a k ní příslušná optimální nadrovina. Úkolem je najít parametry w_0 a b_0 pro tuto optimální nadrovinu. Pár (w_0, b_0) musí splňovat následující omezení

$$w_0^T x_i + b_0 \geq 1 \quad pro\ d_i = 1,$$

$$w_0^T x_i + b_0 \leq -1 \quad pro\ d_i = -1. \tag{2.8}$$

Datové vzory (x_i, d_i), pro které je první nebo druhý řádek rovnice (2.8) splněn se znaménkem „=“ se nazývají podpůrné vektory. Datové vzory tvořící podpůrné vektory jsou nejobtížněji klasifikovatelné. Mají také vliv na optimální polohu rozhodovací roviny.

Nechť je daný vektor $x^{(s)}$ pro který $d^{(s)} = +1$. Podle předchozí definice platí

$$g(x^{(s)}) = w_0^T x^{(s)} \pm b_0 = \pm 1 \quad \text{pro d}^{(s)} = \pm 1. \tag{2.9}$$

Z rovnice (2.7) lze určit, že algebraická vzdálenost od podpůrného vektoru $x^{(s)}$ k optimální nadrovině je [10]

$$r = \frac{g(x^{(s)})}{\|w_0\|}$$

$$= \frac{1}{\|w_0\|} \quad když\ d(s) = 1$$

$$= -\frac{1}{\|w_0\|} \quad když\ d(s) = -1. \tag{2.10}$$

Pro kladné znaménko leží $x^{(s)}$ na kladné straně optimální nadroviny, pro záporné znaménko na záporné straně. Nechť ρ označuje optimální hodnotu hranice separace mezi dvěma třídama, představujícíma trénovací množinu τ, potom z rovnice (2.10) vyplývá, že

$$\rho = 2r = \frac{2}{\|w_0\|}. \tag{2.11}$$

Z rovnice (2.11) vyplývá, že maximalizování hranice separace mezi třídama je rovna minimalizaci Euklidovské normy váhového vektoru w [10].

2.4 Kvadratická optimalizace pro hledání optimální nadroviny

Cílem je vytvoření výpočetně efektivní procedury pro použití trénovacího vzoru $\tau = \{(x_i, d_i)\}_{i=1}^N$ k nalezení optimální nadroviny s následujícím omezením

$$d_i(w^T x_i + b) \geq 1 \quad pro\ i = 1,2,...,N. \tag{2.12}$$

Toto omezení je kombinací dvou řádků rovnice (2.8) s použitím w místo w_0. Formulace omezeného optimalizačního problému může být následující [10]:

Pro daný tréninkový vzor $\{(x_i, d_i)\}_{i=1}^{N}$ *se hledají optimální hodnoty váhového vektoru w a biasu b takové, že je splněno omezení*

$$d_i(w^T x_i + b) \geq 1 \quad pro\ i = 1,2,...,N$$

a váhový vektor w minimalizuje ztrátovou funkci

$$\Phi(w) = \frac{1}{2} w^T w.$$

Toto omezení optimalizačního problému je charakterizováno následujícími dvěma body:

- Ztrátová funkce *Φ(w)* je konvexní funkcí *w*.
- omezení jsou lineární pro *w*.

Omezený optimalizační problém lze vyřešit pomocí Lagrangeových multiplikátorů, viz. [10].

Pokud jsou známy optimální Lagrangeovy multiplikátory označené $\alpha_{0,i}$, lze spočítat optimální váhový vektor w_0 jako

$$w_0 = \sum_{i=1}^{N} \alpha_{0,i} d_i x_i. \qquad (2.13)$$

Optimální bias b_0 je vypočítán s využitím vztahu (2.9) následujícím způsobem

$$b_0 = 1 - w_0^T x^{(s)} \quad pro\ d(s) = 1. \qquad (2.14)$$

2.5 Optimální nadrovina pro neseparovatelné datové vzory

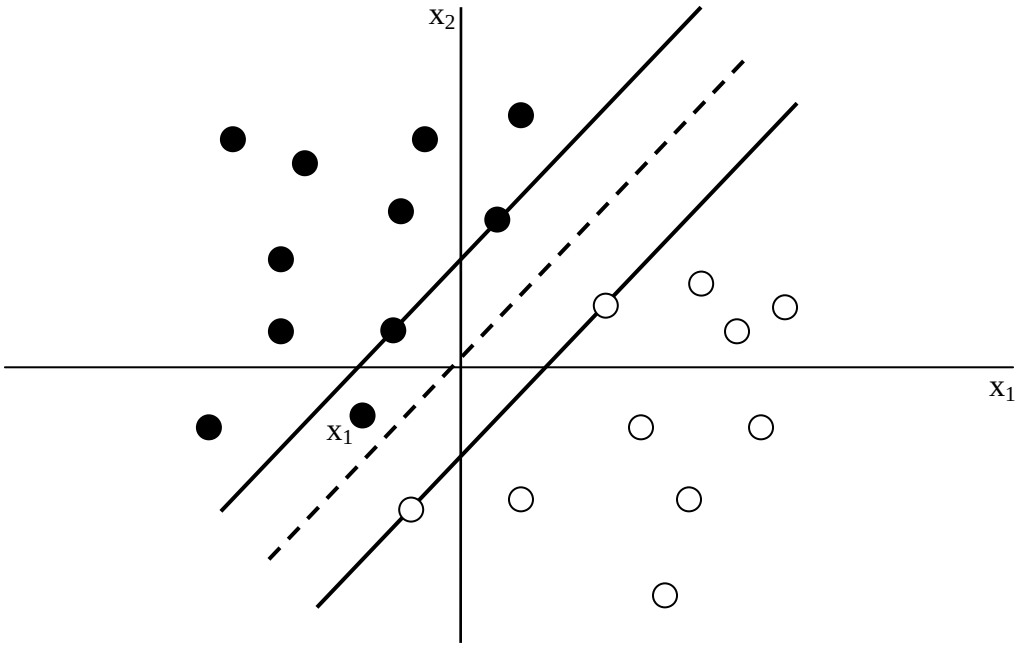
V kapitole pro lineárně separovatelnou množinu dat (datové vzory) se uvažoval případ z obr.7 vlevo, kdy bylo možné tyto data rozdělit lineárním oddělovačem. V případě neseparovatelných datových vzorů je cílem nalezení optimální nadroviny s minimalizováním pravděpodobnosti klasifikační chyby nad těmito daty.

Hranici separace mezi třídami se říká měkká (soft), pokud datové body (x_i,d_i) porušují předpoklad ze vztahu (2.12) [10]

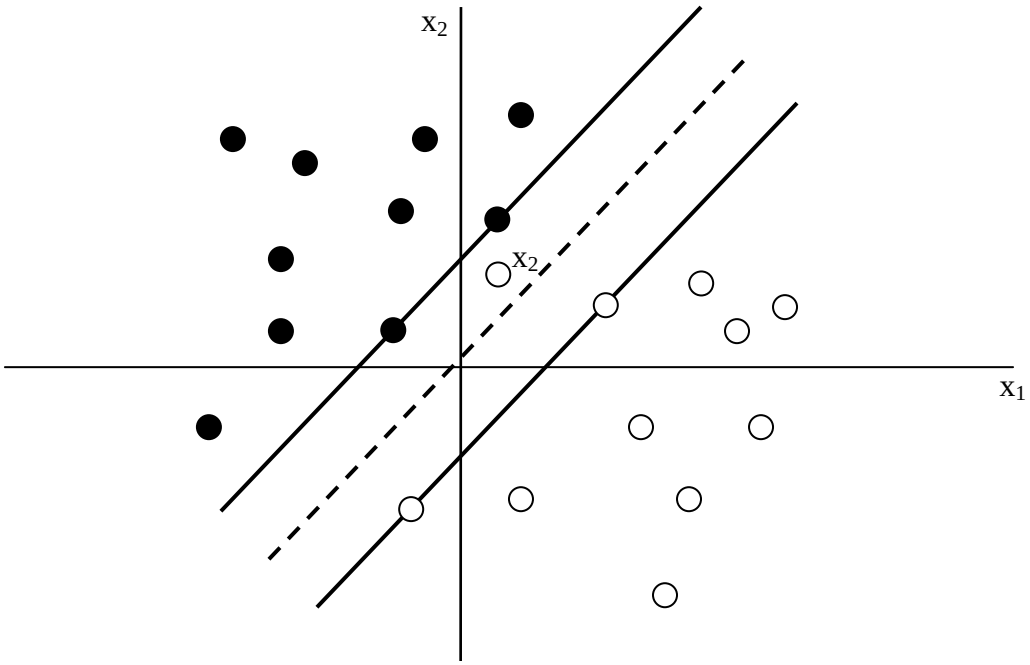
$$d_i(w^T x_i + b) \geq 1 \quad pro\ i = 1,2,...,N.$$

Porušení předpokladu může být důsledkem některé z následujících možností:

- Datový bod (x_i,d_i) spadá dovnitř oblasti separace (mezi podpůrný vektor a rozhodovací nadrovinu), ale na správnou stranu rozhodovací roviny, tzn. datový bod je správně klasifikován, viz. obr.10, bod x_1.
- Datový bod (x_i,d_i) spadá na špatnou stranu rozhodovací roviny, tzn. že je špatně klasifikován, viz obr.11, bod x_2.



Obr.10: Správně klasifikovaný datový bod x_1 uvnitř regionu separace [10]



Obr.11: Nesprávně klasifikovaný datový bod x_2 [10]

K formálnímu zpracování neseparovatelných datových bodů je potřeba zavést do definice separační roviny množinu nezáporných skalárních proměnných $\{\xi_i\}_{i=1}^{N}$ nazývaných volné proměnné (slack variables). Upravením vztahu (2.12) je získán následující vztah

$$d_i(w^T x_i + b) \geq 1 - \xi_i \quad pro\ i = 1,2, ...,N. \tag{2.15}$$

Volné proměnné ξ_i měří odchylku datových bodů od ideálně separovatelných datových bodů. Pokud platí $0 \leq \xi_i \leq 1$, je to první případ zobrazený na obr.8, pro $\xi_i > 1$ nastává druhý případ zobrazený na obr.9.

Cílem je nalézt pro tréninkový vzorek dat oddělující nadrovinu, s minimální průměrnou klasifikační chybou na tomto vzorku. To lze provést prostřednictvím minimalizování funkce [10]

$$\Phi(\xi) = \sum_{i=1}^{N} I(\xi_i - 1). \tag{2.16}$$

S omezením, které je dané vztahem (2.15) a s omezením $\|w\|^2$ pro vektor *w*. Funkce $I(\xi)$ se nazývá indikační funkce a nabývá dvou hodnot podle následujících podmínek

$$\text{k}dyž\ \xi \leq 0\ \ potom\ I(\xi) = 0,$$
$$\text{k}dyž\ \xi > 0\ \ potom\ I(\xi) = 1. \tag{2.17}$$

Protože minimalizace $\Phi(\xi)$ s ohledem na vektor *w* je nekonvexní optimalizační problém, je třeba provést aproximaci následovně

$$\Phi(\xi) = \sum_{i=1}^{N} \xi_i . \tag{2.18}$$

Pro zjednodušení výpočtu je provedena následující formulace minimalizované ztrátové funkce s ohledem na vektor *w*

$$\Phi(w, \xi) = \frac{1}{2} w^T w + C \sum_{i=1}^{N} \xi_i . \tag{2.19}$$

Kde *C* je uživatelem zvolený regulační parametr. Tato formulace plně odpovídá principu minimalizace strukturálního rizika. Parametr *C* je zvolen jedním ze dvou způsobů [10]:

- Parametr *C* je určen experimentálně použitím klasické trénovací, validační a testovací množiny.
- Parametr *C* je určen analyticky odhadováním VC dimenze a následně je použito ohraničení na zevšeobecnění algoritmu založené na VC dimenzi.

Nastavením $\xi i = 0$ pro všechna i ve vztazích (2.15) a (2.19) dělá z uvedeného optimalizačního problému optimalizační problém pro lineárně separovatelné vzory. Aplikováním metody Lagrangeových multiplikátorů lze formulovat duální problém pro neseparovatelné vzory, více viz. (Haykin, kap.6).

2.6 Princip SVM

2.6.1 Definice a vlastnosti jádrových funkcí

Návrh SVM závisí na nelineární transformaci vstupního prostoru do vícerozměrného prostoru [18]. Hledání lineárních oddělovačů v charakteristickém vícerozměrném prostoru se v případě neseparovatelných dat prování pomocí tzv. jádrových funkcí. Proto také SVM spadají do kategorie jádrových algoritmů.

„Jádrová funkce je v souvislosti se Support Vector Machine funkce, která může být aplikována na dvojice vstupních dat k vyhodnocení skalárního součinu v nějakém odpovídajícím prostoru [11].“

Nechť $\{\varphi_j(x)\}_{j=1}^{m_1}$ označuje množinu nelineárních transformací ze vstupního prostoru do vícerozměrného prostoru, kde *x* je vektor ze vstupního prostoru. Dimenze vstupního prostoru je m_0 a dimenze charakteristického prostoru je m_1. Rozhodovací nadrovina je potom definována následovně [10]

$$\sum_{j=1}^{m_1} w_j \varphi_j(x) + b = 0, \qquad (2.20)$$

kde $\{w_j\}_{j=1}^{m_1}$ je množina lineárních vah spojujících vícerozměrný prostor s výstupním prostorem a *b* je bias. Tuto rovnici lze zjednodušit stanovením $\varphi_0(x) = 1$ a předpokladem, že w_0 označuje bias. Za těchto předpokladů má nová rovnice tvar

$$\sum_{j=0}^{m_1} w_j \varphi_j(x) = 0. \qquad (2.21)$$

Na základě právě uvedených skutečností je definován vektor $\varphi(x)$, který určuje zobrazení v charakteristickém vícerozměrném prostoru na základě vstupního vektoru x

$$\varphi(x) = \left[1, \varphi_1(x), ..., \varphi_{m_1}(x)\right]^T . \qquad (2.22)$$

Rozhodovací rovina vypadá potom následovně

$$w^T \varphi(x) = 0 . \qquad (2.23)$$

S využitím Lagrangeových multiplikátorů (více např. [10]) a substitucí w ve vztahu (2.23) lze definovat rozhodovací rovinu vypočítanou ve vícerozměrném charakteristickém prostoru následujícím způsobem

$$\sum_{i=1}^{N} \alpha_i d_i \varphi^T(x_i)\varphi(x) = 0 , \qquad (2.24)$$

kde $\varphi(x_i)$ odpovídá vstupnímu datovému bodu x_i a výraz $\varphi^T(x_i)\varphi(x)$ odpovídá vnitřnímu produktu vektorů vícerozměrného prostoru (s ohledem na vstupní vektor x a vstupní vzorek x_i). Proto lze definovat vnitřní produkt jádra nebo-li jádrovou funkci jako

$$K(x, x_i) = \varphi^T(x)\varphi(x_i) = \sum_{j=0}^{m_1} \varphi_j(x)\varphi_j(x_i) \quad pro\ i = 1, 2, ..., N. \qquad (2.25)$$

Jádrová funkce je symetrická. Dosazením (2.25) do (2.24) je optimální nadrovina definována ve tvaru

$$\sum_{i=1}^{N} \alpha_i d_i K(x, x_i) = 0 . \qquad (2.26)$$

2.6.2 Jádrové funkce SVM

Metoda SVM používá několik základních typů jádrových funkcí. Jsou to: Lineární, polynomická, RBF a sigmoidální (viz. tab.1). Každá z těchto funkcí používá specifický způsob transformace do charakteristického vícerozměrného prostoru. Dimenzionalita tohoto prostoru je určena počtem podpůrných vektorů vybraných ze vzorku trénovacích dat. Požadavek na jádro je tzv. Mercerův teorém (více viz. Haykin, kap.6). Polynomické a RBF jádro vždy splňuje tento teorém, Sigmoidální pouze pro některé parametry (*koef.* a γ).

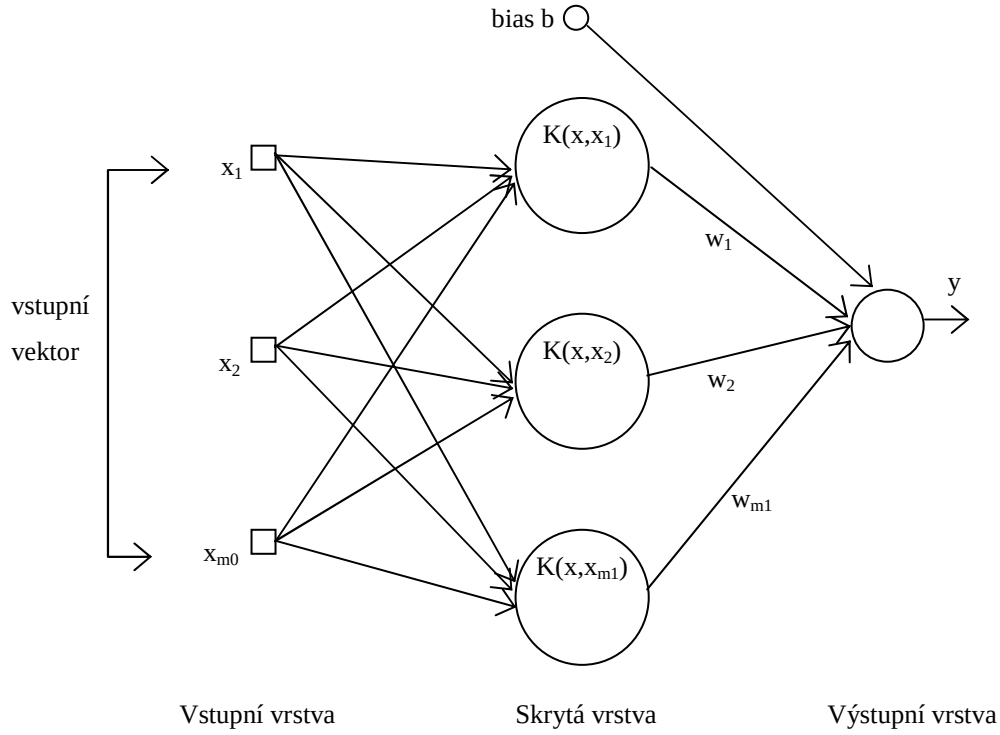
Typ SVM	$K(x,x_i)$, i=1, 2, ..., N
Lineární	$x^T x_i$
Polynomický	$(\gamma x^T x_i + koeficient)^{\beta}$
RBF	$\exp(-\gamma \|x - x_i\|^2)$
Sigmoidální	$\tanh(\gamma x^T x_i + koeficient)$

Tab.1: Typy jádrových funkcí SVM [19], [10]

Metoda SVM nabízí oproti neuronovým sítím méně parametrů. V lineární jádrové funkci se žádný parametr nenastavuje. V ostatních uvedených jádrových funkcích mění uživatel parametr γ a v polynomickém jádru i stupeň β. V sigmoidálním a polynomickém jádru lze měnit i koeficient. Při řešení konkrétního problému je možno volit z různých jádrových funkcí tu nejvhodnější a měnit konkrétní parametry tak, aby výsledky byly co nejpřesnější.

2.6.3 Architektura SVM

V přirovnání SVM k neuronovým sítím lze hovořit o dopředné síti s jednou skrytou vrstvou, jak je zobrazeno na obr.12. Architektura je grafickým zobrazením dosud popsaných poznatků. Vstupní vektor tvořící vstupní vrstvu se skládá z jednotlivých datových případů (parametrů). Neurony skryté vrstvy obsahují jádrové funkce vzhledem k vektoru a určitému datovému případu. Jako *w* jsou označeny váhy synapsí mezi vícerozměrným prostorem a výstupem. Podle předchozích úvah by mohla být synapse biasu *b* označena w_0. Výstupní neuron je označen *y*.



Obr.12: Architektura SVM [10], [18]

2.7 Support Vector Regression

Jak již bylo naznačeno v předchozích částech této práce, metoda SVM původně se zabývající klasifikací datových bodů do tříd byla rozšířena i o možnost řešení nelineárních regresních problémů. Díky tomu lze metodu SVM využít pro predikci. Metoda SVM pro řešení regresních problémů se nazývá Support Vector Regression (dále jen SVR).

Nejprve je nutno zvolit vhodné optimalizační kriterium. Požadavkem je stabilní odhad procedury necitlivý k malým změnám v modelu. Pro případ nelineární regrese se používá minimaxová (minimalizuje maximální pokles výkonu) procedura. Úkolem je minimalizovat absolutní chybu. Vhodným kriteriem je pro případ regrese tzv. ztrátová funkce *L(d,y)*, která má následující tvar

$$L(d, y) = |d - y| . \qquad (2.27)$$

Proměnná *d* označuje požadovanou odezvu a *y* je odhadovaný výstup.

Ke konstrukci SVM pro aproximaci požadované odezvy *d*, lze použít rozšíření ztrátové funkce, navržené v [16] následujícím způsobem [10]

$$L_\varepsilon(d,y) = |\mathrm{d} - \mathrm{y}| - \varepsilon \quad pro\ |d-y| \geq \varepsilon\,,$$

$$L_\varepsilon(d,y) = 0 \qquad v\ ostatních\ případech, \tag{2.28}$$

kde ε je stanovený parametr. Ztrátová funkce $L_\varepsilon(d,y)$ se nazývá ε-necitlivostní (ε-insensitive) ztrátová funkce.

Je dán nelineární regresní model, ve kterém je závislost skaláru *d* na vektoru *x* popsaná následovně [10]

$$d = f(x) + e\,. \tag{2.29}$$

Proměnná *e* je přidaný šum, který je statisticky nezávislý na vstupním vektoru *x*. Funkce *f()* a statistiky šumu jsou neznámé. K dispozici je množina trénovacích vzorků dat $\{(x_i, d_i)\}_{i=1}^{N}$, kde x_i je datový vzor vstupního vektoru *x* a d_i je odpovídající hodnota výstupu modelu *d*. Cílem je určit závislosti *d* na *x*.

Předpokládá se, že odhad *d*, označený *y* je rozšířen v členech množiny nelineárních bazických funkcí $\{\varphi_j(x)\}_{j=0}^{m_1}$ následovně

$$y = \sum_{j=0}^{m_1} w_j \varphi_j(x) = w^T \varphi(x)\,, \tag{2.30}$$

kde

$$\varphi(x) = [\varphi_0(x), \varphi_1(x), ..., \varphi_{m_1}(x)]^T \tag{2.31}$$

a

$$w = [w_0, w_1, ..., w_{m_1}]^T\,. \tag{2.32}$$

Stejně jako v případě rozpoznávání datových vzorů se předpokládá, že $\varphi_0(x) = 1$ a váha w_0 označuje bias *b*. Problém je řešen minimalizací empirického rizika

$$R_{emp} = \frac{1}{N} \sum_{i=1}^{N} L_\varepsilon(d_i, y_i)\,, \tag{2.33}$$

s podmínkou

$$\|w\|^2 \leq c_0\,, \tag{2.34}$$

kde c_0 je konstanta.

Zavedením volných proměnných (Slack variables) $\{\xi_i\}_{i=1}^{N}$ a $\{\xi'_i\}_{i=1}^{N}$ lze formulovat minimalizovanou chybovou funkci následovně

$$\Phi(w,\xi,\xi') = \mathrm{C}\left(\sum_{\mathrm{i}=1}^{\mathrm{N}}\left(\xi_{\mathrm{i}} + \xi'_{\mathrm{i}}\right)\right) + \frac{1}{2}w^T w\,, \qquad (2.35)$$

kde konstanta *C* je uživatelsky volený parametr a je větší než 0. Tato funkce je minimalizována s podmínkami [21], [19]

$$w^T\varphi(x_i) - d_i \le \varepsilon + \xi_i' \quad pro\ i = 1...N, \qquad (2.36)$$

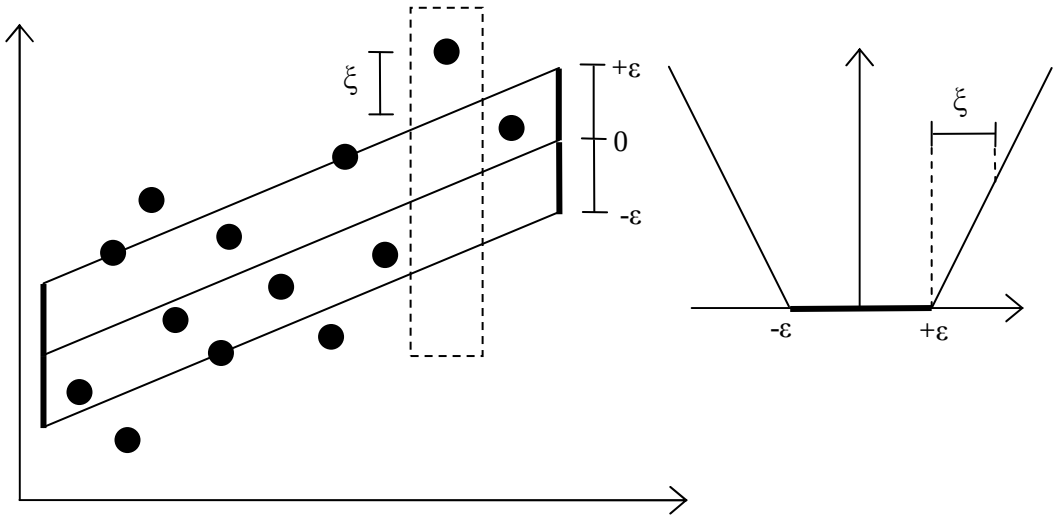
$$d_i - w^T\varphi(x_i) \le \varepsilon + \xi_i \quad pro\ i = 1...N, \qquad (2.37)$$

$$\xi_i, \xi_i' \ge 0 \quad pro\ i = 1...N\,. \qquad (2.38)$$

Parametry *C* a ε by měly být voleny současně. Ztrátová ε-necitlivostní funkce $|\xi|_\varepsilon$ je popsána následovně (graficky zobrazuje tuto situaci obr.13)

$$|\xi|_\varepsilon = 0 \text{ když } |\xi| < \varepsilon\,,$$

$$|\xi|_\varepsilon = |\xi| - \varepsilon \text{ když } |\xi| \ge \varepsilon\,. \qquad (2.39)$$



Obr.13: ε-necitlivostní ztrátová funkce (vpravo) a zobrazení ztráty odpovídající lineární SVM [21]

Primární i duální problém pro regresi lze stanovit pomocí definice Lagrangeovy funkce a zavedením Lagrangeových multiplikátorů. Tento proces je podrobněji popsán např. v [10]. Výsledná SVR funkce k řešení aproximačního problému má následující tvar

$$F(x,w) = w^T x = \sum_{i=1}^{N} (\alpha_i - \alpha'_i) K(x, x_i), \qquad (2.40)$$

kde α_i a α_i' jsou Lagrangeovy multiplikátory a $K(x,x_i)$ je jádrová funkce definovaná v souladu s Mercerovým teorémem.

Klasická SVR metoda, nebo-li regrese prvního typu je popsána vztahem (2.35) s podmínkami (2.36), (2.37) a (2.38). V této metodě je obtížné určit hodnotu parametru ε předem. Tento problém je částečně vyřešen novým algoritmem v-SVR, kde je parametr ε v procesu optimalizace a je řízen jiným parametrem $v \in (0,1)$. Parametr v je horní mez části chybných bodů nebo dolní mez části bodů uvnitř oblasti ε. Dobré ε může být automaticky nalezeno volbou v. Parametr v je tak vhodným volitelným parametrem. Regresní metoda druhého typu, která používá parametr v je popsána následující minimalizovanou chybovou funkcí

$$\Phi(w,\xi,\xi',\varepsilon) = -C\left(v\varepsilon + \frac{1}{N}\sum_{i=1}^{N} (\xi_i + \xi'_i) \right) + \frac{1}{2} w^T w . \qquad (2.41)$$

Tato funkce je minimalizována s podmínkami

$$w^T \varphi(x_i) - d_i \le \varepsilon + \xi_i \quad pro\ i = 1...N, \qquad (2.42)$$

$$d_i - w^T \varphi(x_i) \le \varepsilon + \xi_i' \quad pro\ i = 1...N, \qquad (2.43)$$

$$\xi_i, \xi_i' \ge 0, \varepsilon \ge 0 \quad pro\ i = 1...N . \qquad (2.44)$$

Stejně jako v případě SVM, tak i SVR lze implementovat s jádrovými funkcemi zobrazenými v tab. 1.

3 Data pro modelování

3.1 Charakteristika dat

Jako data pro praktickou část této práce byly použity statistiky návštěvnosti webových stránek Univerzity Pardubice (www.upce.cz). Tyto statistiky byly získány pomocí nástroje Google Analytics.

Google Analytics je bezplatný nástroj pro sledování Web miningových ukazatelů. Po registraci na stránkách Google Analytics je možno využívat jeho služeb. Umístěním Java Scriptového kódu na webových stránkách lze sledovat příslušné hodnoty ukazatelů. Google Analytics nabízí měření mnoha zajímavých ukazatelů jako jsou počty návštěv, unikátní návštěvy, operační systém a typ prohlížeče návštěv, směry příchodů a odchodů uživatelů, hledaná slova uživatelů, nové návštěvníky a mnoho dalších. Lze zjistit i národnost návštěvníků a na základě toho např. rozšířit Web o vhodnou jazykovou verzi. Google Analytics umožňuje naměřené hodnoty ukazatelů zobrazit v přehledné grafické formě.

K predikci návštěvnosti Webu Univerzity Pardubice je nutné zaznamenávat návštěvy během dané časové periody. Návštěva je zde definována jako neopakující se kombinace IP adresy a cookies.

Informace získané prostřednictvím Google Analytics (Květen 2009) jsou následující [23]:

- Celková návštěvnost během daného měsíce. Návštěvnost má od pondělí do konce týdne klesající charakter, sobota je den s nejmenší návštěvností.
- Průměrný počet navštívených stránek je více než tři.
- Návštěvník zůstává na určité stránce průměrně pět a půl minuty.
- Počet návštěvníků, kteří stránku po otevření ihned opustí (bounce rate) je kolem 60%.
- Návštěvníci přicházejí na web především přímo (ne z jiných stránek).
- Nejoblíbenější stránkou je hlavní strana Univerzity, následuje stránka Fakulty Ekonomicko-Správní a Fakulty Filozofické.
- Většina návštěvníků stránek Univerzity používá prohlížeč Internet Explorer (60%), dále Firefox (33%) a Operu (5%).

Data pro predikci byla naměřená od 21.srpna 2007 do 20. září 2009 a jeden datový údaj je hodnota počtu návštěv za jeden den. Data jsou rozdělena na tři různě dlouhé časové řady a jednotlivé časové řady se částečně překrývají:

- Krátká časová řada, její počet je 264 záznamů (dnů)
- Střední časová řada s počtem 480 záznamů (dnů)
- Dlouhá časová řada s počtem 752 záznamů (dnů)

Dlouhou časovou řadu tvoří záznamy ve výše uvedeném časovém intervalu, krátká a střední časová řada je podmnožinou dlouhé časové řady.

Co se týká dalších charakteristik dat, tak lze zmínit, že návštěvnost Webu Univerzity Pardubice se pohybuje od 1000 do 7000 unikátních návštěv za den, denní průměrný počet návštěv je přibližně 3200 a směrodatná odchylka těchto dat má hodnotu přibližně 1100.

3.2 Předzpracování dat

Všechny tři typy dat byly předzpracovány matematicko-statistickými metodami uvedenými v tab.2.

Metoda	Charakteristika
Jednoduchý klouzavý průměr (SMA)	5, 7, 9 dní
Centrovaný klouzavý průměr (CMA)	4, 6, 8 dní
Klouzavý medián (MM)	5, 7, 9 dní
Jednoduché exponenciální vyrovnání (SES)	pro $\alpha = 0{,}1$ a $\alpha = 0{,}2$
Dvojité exponenciální vyrovnání (DES)	pro $\alpha = 0{,}7$ a $\alpha = 0{,}9$

Tab.2: Metody předzpracování časových řad [23]

Tato data bylo potřeba standardizovat, aby na vstupu byly jednotné hodnoty parametrů se stejnou váhou. Standardizace dat slouží k odstranění rozdílů (např. různé jednotky, různé řády dat) mezi vstupními daty. Vztah pro standardizování dat je následující [26]

$$z_{ij} = \frac{x_{ij} - \overline{x_j}}{s_j}, \qquad (3.1)$$

kde z_{ij} je standardizovaná i-tá hodnota j-tého parametru, x_{ij} je původní i-tá hodnota

j-tého parametru, $\overline{x_j}$ je průměrná hodnota j-tého parametru a s_j je směrodatná odchylka j-tého parametru.

V tab.3 jsou uvedeny některé statistické údaje pro jednotlivé typy dat po standardizaci a předzpracování. Všechna data mají společný průměr a směrodatnou odchylku. Po standardizaci je průměr všech dat vždy rovný 0 a směrodatná odchylka je pro všechna data rovna 1.

Parametr	Krátká data		Střední data		Dlouhá data		Průměr	Směrodatná odchylka
	MIN	MAX	MIN	MAX	MIN	MAX		
SMA	-2.88	3.25	-3.15	2.94	-2.65	2.48	0	1
CMA	-2.8	3.19	-3.07	2.94	-2.47	2.47	0	1
MM	-2.74	2.78	-2.97	2.69	-2.71	2.04	0	1
SES	-1.96	2.69	-2.24	2.62	-2.11	2.41	0	1
DES	-2.96	2.99	-3.16	2.83	-2.98	2.95	0	1
y	-2.47	3.47	-2.61	2.83	-2.49	3.36	0	1

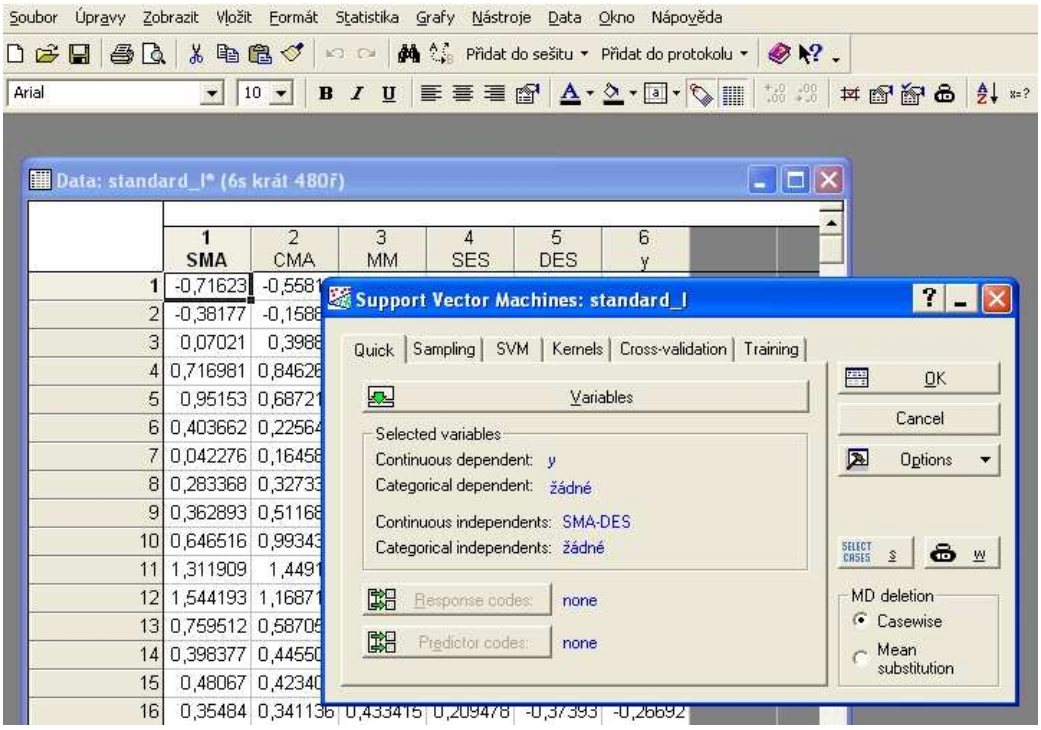
Tab.3: Základní statistické údaje standardizovaných předzpracovaných dat

4 Modelování

4.1 Modelovací prostředí

Návrh vhodného modelu SVR a veškeré s tím spojené experimenty byly prováděny v prostředí programu Statistica 7.0. Statistica je nástroj vhodný k práci s daty, k provádění statistických operací nad daty a k zobrazení dat pomocí různých typů grafů. Tento nástroj je snadno a intuitivně ovladatelný, podporuje načítání velkého množství typů souborů a mezi nimi i formát Microsoft Excel.

V nabídce hlavního menu je Statistika, pod ní se nachází položka vytěžování dat a ve vyvolaném seznamu metod a modelů se nachází i položka strojové učení. Mezi třemi nabízenými metodami je zde k dispozici i metoda SVM, viz. obr.14. Nastavení SVM se provádí prostřednictvím sedmi záložek. Na první se definuje zda jde o

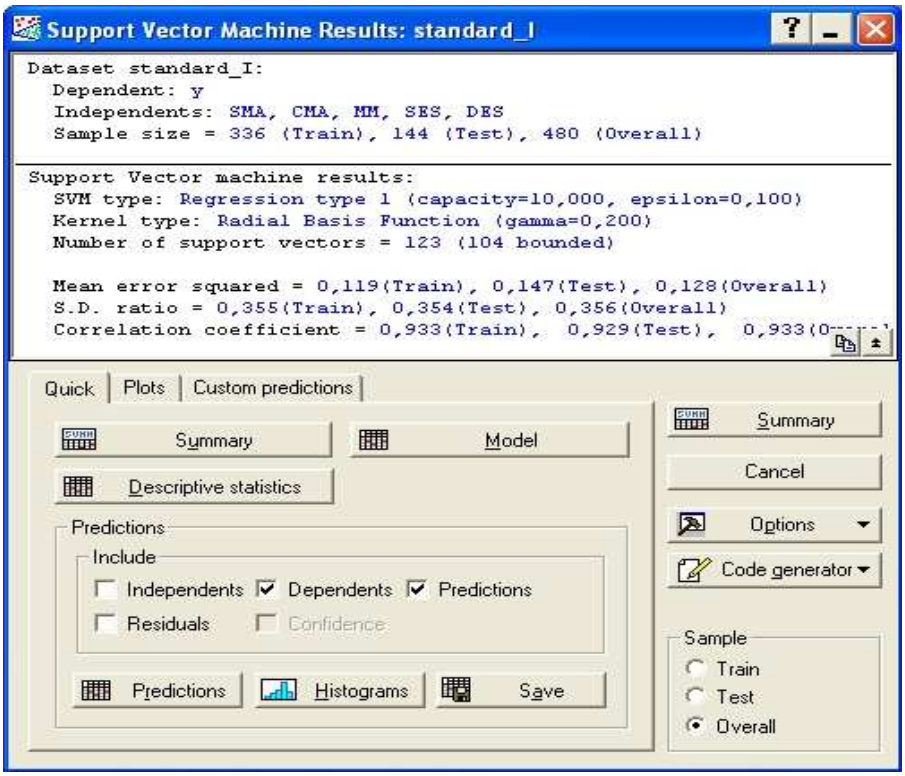


Obr.14: Modelování SVM v prostředí Statistica

klasifikaci nebo predikci a volí se zde příslušné vstupní hodnoty a výstupní hodnota. Na druhé záložce se nastavují vlastnosti rozdělení dat na tréninkovou a testovací množinu. Nastavuje se zde také, jakým způsobem mají být data rozdělena. Na třetí záložce se nastavuje typ regrese nebo klasifikace a k tomu příslušné měněné parametry. Čtvrtá

záložka slouží k nastavení jádrových funkcí a jejich příslušných parametrů. V další záložce se aktivuje křížová validace a v poslední je nejdůležitější položkou maximální počet iterací.

Po aktivaci tlačítka OK se provede učení pro nastavené parametry a otevře se okno s výsledky (viz. obr.15). V tomto okně je souhrn všech důležitých výsledků. Jsou zde zobrazeny vstupní hodnoty, výstupní hodnota, velikost vzorku (trénovacího, testovacího a celkem), typ SVM a příslušné parametry, typ jádrové funkce a příslušné parametry, počet podpůrných vektorů, apod. Pod tlačítkem „Deskriptivní statistiky“ se nachází požadované výsledky, především chyby a nejžádanější položka „průměr čtvercových chyb“. Tyto výsledky lze zobrazit pro trénovací, testovací nebo souhrnně pro všechna data. Pod tlačítkem „Histograms“ se nachází histogramy pozorovaných a předpovídaných hodnot. Pod tlačítkem „Predictions“ se nachází tabulka se dvěma sloupci. Jeden uvádí skutečné hodnoty výstupní proměnné a druhý predikované hodnoty výstupní proměnné, opět pro trénovací, testovací nebo celou množinu dat.



Obr.15: Okno zobrazení výsledků v prostředí Statistica

Na záložce „Plots“ lze vykreslovat požadované grafy. Nejzajímavější je nejspíš nastavení osy X jako původních hodnot a osy Y jako predikovaných hodnot. Čím více se graf blíží přímce, neboli čím je pásmo zobrazených bodů užší, tím je predikce

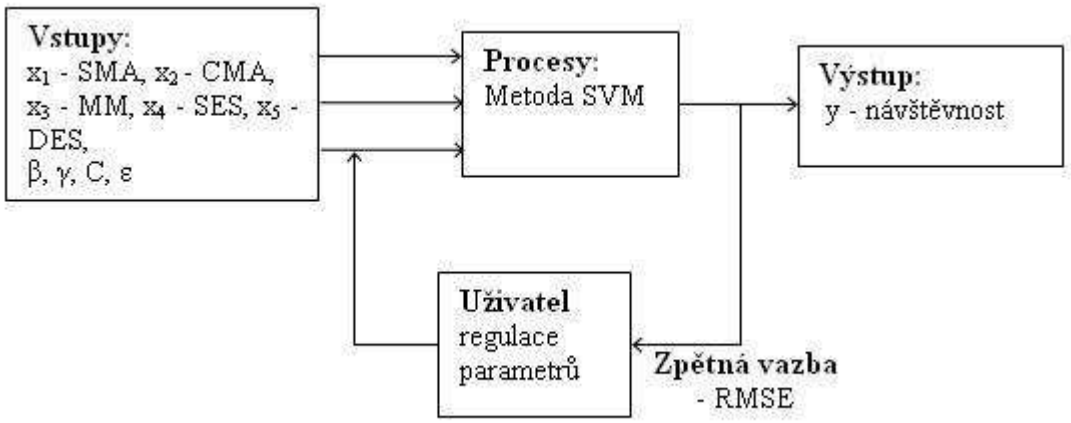
přesnější. Lze zde vykreslovat i 3D grafy. Na záložce „Custom predictions“ lze provést predikci výstupní hodnoty na základě zvolených vstupních parametrů.

Veškeré výsledky byly přeneseny do prostředí Microsoft Excel 2003 a zde byly výsledky zpracovány do tabulkové a grafické podoby.

4.2 Vstupní a výstupní data

Jako vstupní data do modelu byla použita standardizovaná data předzpracována jednotlivými metodami. Vstupní vektor x tedy tvoří 5 hodnot $x=(x_1,x_2,x_3,x_4,x_5)$, kde x_1 je klouzavý průměr, x_2 je centrální klouzavý průměr, x_3 je klouzavý medián, x_4 je jednoduché exponenciální vyrovnání a x_5 je dvojité exponenciální vyrovnání. Výstupní hodnotou jsou standardizovaná původní data y. Predikována je hodnota y v čase $t+1$ na základě hodnoty x v čase t.

Celý proces modelování by mohl být popsán jako uzavřený zpětnovazební systém s více vstupy a jedním výstupem (Multiple Input Single Output, zkratkou MISO). Pouze ale v případě, není-li uvažováno předzpracování dat a jako vstupní hodnoty jsou pevně určená data, předzpracována výše popsanými metodami. Toto schéma je znázorněno na obr.16.



Obr.16: Proces modelování

4.3 Nastavení parametrů modelu

V každém ze třech typů testovaných dat byly provedeny experimenty pro různé rozdělení dat na trénovací a testovací část a pro různá jádra. V případě rozdělení dat na trénovací a testovací část nabízí prostředí Statistica tři možnosti:

- Náhodný výběr testovacích a trénovacích dat z celkového množství dat s určením procentuálního zastoupení trénovací podmnožiny.
- Uživatelem zvolená vzorová proměnná, která je nominálního typu a udává které případy z množiny dat zařadit do testovací a které do trénovací podmnožiny.
- Výběr prvních N vzorků dat jako tréninkovou podmnožinu.

Pro účel experimentů byl pokaždé zvolen náhodný výběr dat. Zkoumány byly výsledky pro různé rozdělení trénovací a testovací podmnožiny dat. Zvoleno bylo následujících pět rozdělení:

- trénovací část : testovací část, 50% : 50%
- trénovací část : testovací část, 60% : 40%
- trénovací část : testovací část, 70% : 30%
- trénovací část : testovací část, 80% : 20%
- trénovací část : testovací část, 90% : 10%

Ze dvou možných typů regrese popsaných v teoretické části práce byl vybrán první typ (tzn., že měněny byly parametry C a ε). Experimenty byly prováděny na dvou jádrech a to RBF jádru a Polynomickém jádru. V případě RBF jádra byl měněn pouze parametr γ, v případě Polynomického jádra, lze měnit parametr γ, stupeň β a *koeficient*. V modelu byly však experimentálně měněny pouze parametry γ a β. Hodnota koeficientu byla vždy rovna nule. Počet cyklů byl pevně nastaven pro všechny prováděné experimenty na 600. Prostředí Statistica nabízí hledání optimálního modelu pomocí tzv. křížové validace. Tento algoritmus nebyl při modelování použit a optimální parametry modelu byly hledány experimentálně podle zvolené vhodné metodiky.

4.4 Sledovaná chyba modelu

Prostředí Statistica vrací po naučení sítě jako ukazatel kvality naučení průměr čtvercových chyb, zkráceně MSE (Mean Squared Error). K vyjádření MSE je potřeba nejprve definovat SSE, což je součet čtvercových chyb (Sum Squared Error). SSE je daný následujícím vztahem

$$SSE = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 , \qquad (4.1)$$

kde y_i je naměřená hodnota, $\hat{y}_i$ je predikovaná hodnota a n je počet vzorků dat. Potom lze MSE vyjádřit jako

$$MSE = \frac{SSE}{n}, \qquad (4.2)$$

kde n je počet vzorků dat. Hledanou chybou je však směrodatná chyba RMSE (Root Mean Squared Error). Chyba RMSE se vypočítá z MSE následovně

$$RMSE = \sqrt{MSE}\,. \qquad (4.3)$$

Hledaná nejmenší chyba v prostředí Statistica je tedy MSE, ale model se analyzuje pomocí ukazatele RMSE, který se z MSE vypočítá.

4.5 Postup při provádění experimentů

Hledání ideálních parametrů pro výsledný model s sebou přineslo několik klíčových problémů. Bylo potřeba zvolit vhodný postup při změnách parametrů. Nejprve byly parametry měněny jednotlivě a sledována odezva (průměr čtvercových chyb) při každé změně. Na základě toho byl vypozorovaný vliv změny některých parametrů na změnu výstupu. Sledován byl ukazatel průměr čtvercových chyb pro trénovací i testovací množinu dat, především ale pro testovací množinu, na které se testuje kvalita a přesnost naučení sítě. Pro další modelování vyvstalo několik otázek:

- Jaké zvolit výchozí parametry modelu?
- Jak postupovat při hledání optimálního modelu?
- Kolik dalších změn parametru provést pokud se při změnách chyba stále zvyšuje?
- S jakou přesností měnit které parametry?
- V jakém rozsahu měnit které parametry?
- Jakým způsobem ukládat výsledky?

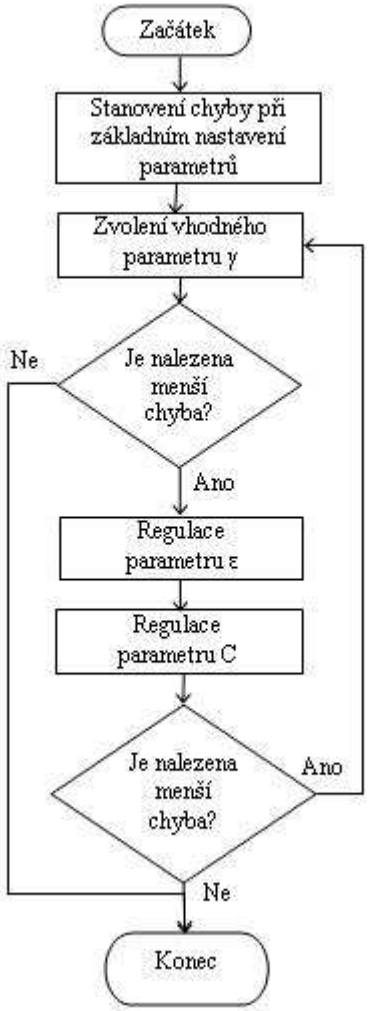
Jako výchozí parametry byly v případě RBF jádra nastaveny: $C = 1$, $\varepsilon = 0.1$, $\gamma = 0.1$ a v případě Polynomického jádra byly výchozí parametry modelu následující: $\beta = 1$, $C = 1$, $\varepsilon = 0.1$, $\gamma = 0.1$. Jak bylo později zjištěno pokud je velká chyba modelu s těmito základními parametry, bude i poměrně velká chyba optimálního modelu. Pokud model se základními parametry vykazuje menší chybu, bude většinou i menší chyba v optimálně navrženém modelu.

Při hledání nejmenší chyby modelu bylo potřeba zvolit vhodný systém hledání optimálních parametrů modelu. V případě polynomického jádra byly vyloučeny z regulovatelných parametrů dva jádrové parametry a to

- koeficient (viz. kapitola nastavení parametrů modelu)

- β (experimentálně byla zjištěna zvyšující se chyba s nastavením větší hodnoty tohoto parametru) – tento parametr byl pokaždé nastaven na hodnotu 1.

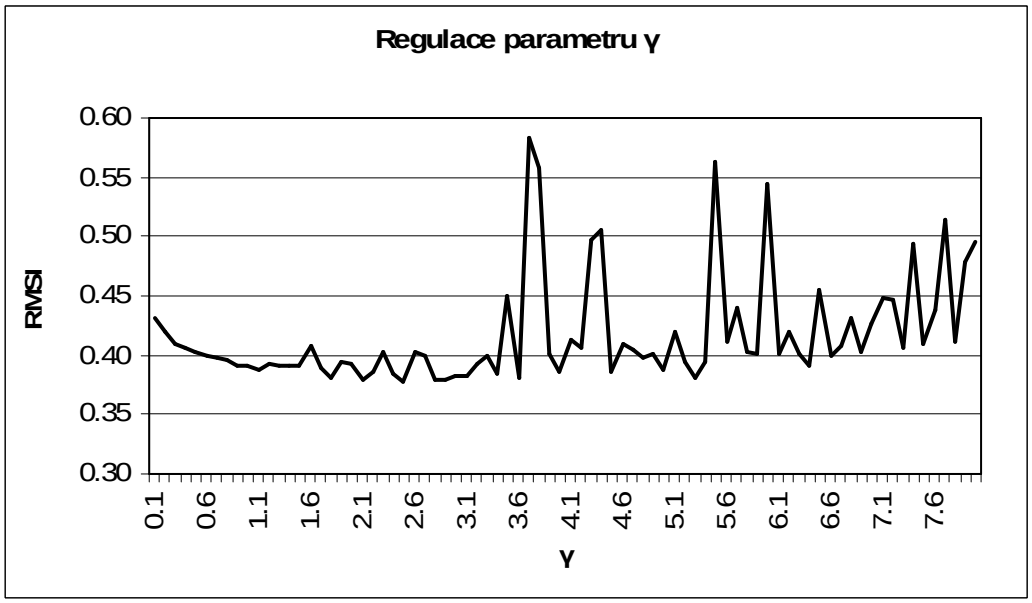
Z tohoto důvodu byly měněné pouze SVR parametry *C* a ε a jádrový parametr γ v polynomickém i RBF jádru. Zapisována byla každá nová nejmenší nalezená chyba modelu. Na obr.17 je znázorněn postup při hledání nejmenší chyby modelu.



Obr.17: Postup při hledání optimálního modelu

Otázkou bylo, do jaké míry měnit velikost určitého parametru. Pokud se chyba zvyšovala po určitém množství změn nastavení parametru, byly tyto regulace parametru ukončeny. Muselo také dojít k situaci, kdy bylo z dosažených výsledků velmi nepravděpodobné další snížení chyby. Jako příklad je v grafu 1 ukázka regulace parametru γ s krokem 0.1, kdy nejlepší chyby bylo dosaženo při hodnotě γ=2.5, ale

např. od hodnoty $\gamma=3.5$ a $\gamma=5.2$ se chyba znovu snižovala, trendově však rostla. Regulace tak byla ukončena až při hodnotě $\gamma=8$.



Graf 1: Regulace parametru γ

Každý parametr byl regulován s určitou zvolenou přesností. Pro parametr β (před tím než byl vyloučen z množiny regulovaných parametrů) byl zvolen krok 1, pro parametr γ byl zvolen krok 0.1 v některých případech i 0.01, pro parametr C byl zvolen krok 1 a pro parametr ε to byl krok 0.01. U parametru ε byly vyzkoušeny i hodnoty v řádu desetin, ale chyba byla při těchto hodnotách příliš velká.

Z časových důvodů nebyla zaznamenávána každá nalezená chyba modelu, ale bylo zvoleno ukládání každé další nejmenší nalezené chyby modelu a k ní příslušné hodnoty parametrů.

5 Analýza výsledků

5.1 Charakteristiky parametrů modelu SVR

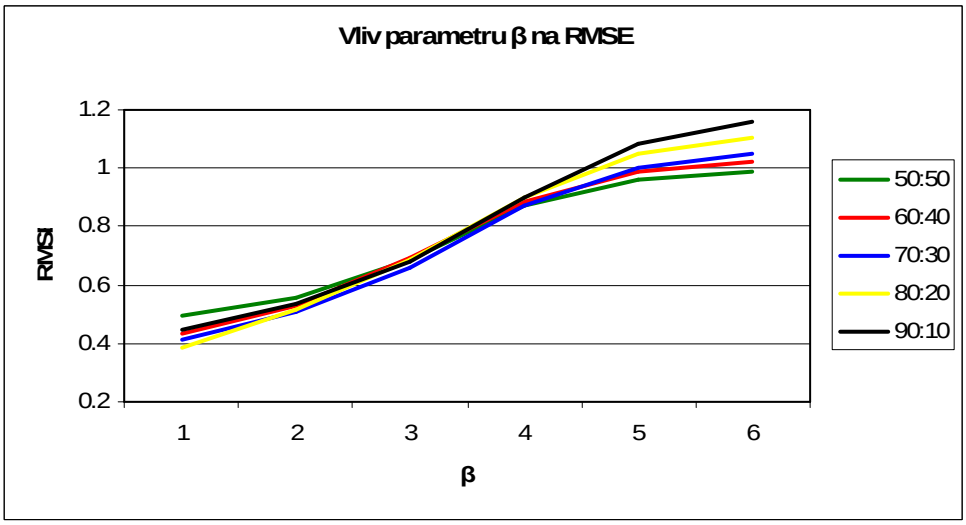
V průběhu modelování byly zjišťovány vlastnosti parametrů a to, jaké mají vlivy na SVR model. Především bylo zjišťováno, jak jednotlivé parametry ovlivňují chybu RMSE. Pro RBF i polynomické jádro byly sledovány změny parametrů C a ε, dále byl pro RBF jádro sledovaný jádrový parametr γ a pro polynomické jádro parametry γ a β. Zjišťováno bylo také, jak ovlivňuje změna jednoho parametru následnou změnu jiného parametru za účelem snižování RMSE chyby. Sledována byla především RMSE chyba testovací množiny dat ($RMSE_{test}$), ale zaznamenávána byla i chyba trénovací množiny ($RMSE_{train}$).

5.1.1 Vlivy parametrů na RMSE

Experimentálně bylo zjištěno, že průběh určitého parametru na $RMSE_{test}$ závisí i na stavu ostatních parametrů. V této části budou uvedeny vlivy parametrů na chybu pro základní stanovené nastavení ostatních parametrů. Budou zkoumány následující závislosti pro polynomickou jádrovou funkci:

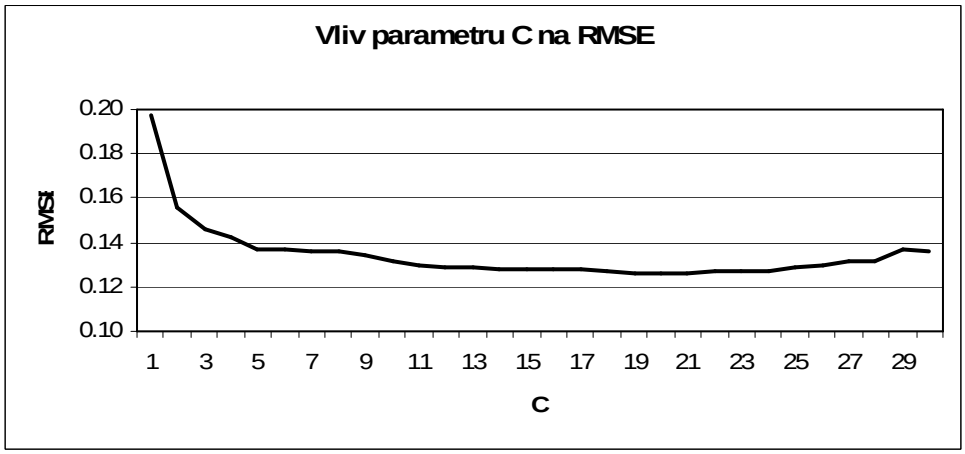
- Vliv parametru β při nastavení $C = 1, \varepsilon = 0.1, \gamma = 0.1$
- Vliv parametru C při nastavení $\beta = 1, \varepsilon = 0.1, \gamma = 0.1$
- Vliv parametru ε při nastavení $\beta = 1, C = 1, \gamma = 0.1$
- Vliv parametru γ při nastavení $\beta = 1, C = 1, \varepsilon = 0.1$

V případě SVR s polynomickým jádrem je uživatelem nastavovaný parametr β. Experimentálně bylo zjištěno, že zvyšováním tohoto parametru roste $RMSE_{test}$ a výrazně také roste $RMSE_{train}$ a to bez ohledu na hodnotách ostatních parametrů. V grafu 2 je zobrazen průběh chyby $RMSE_{test}$ při různých hodnotách parametru β. Graf je vytvořen z experimentů na krátké časové řadě pro různé rozdělení trénovací a testovací množiny dat.



Graf 2: Vliv parametru β *na* $RMSE_{test}$

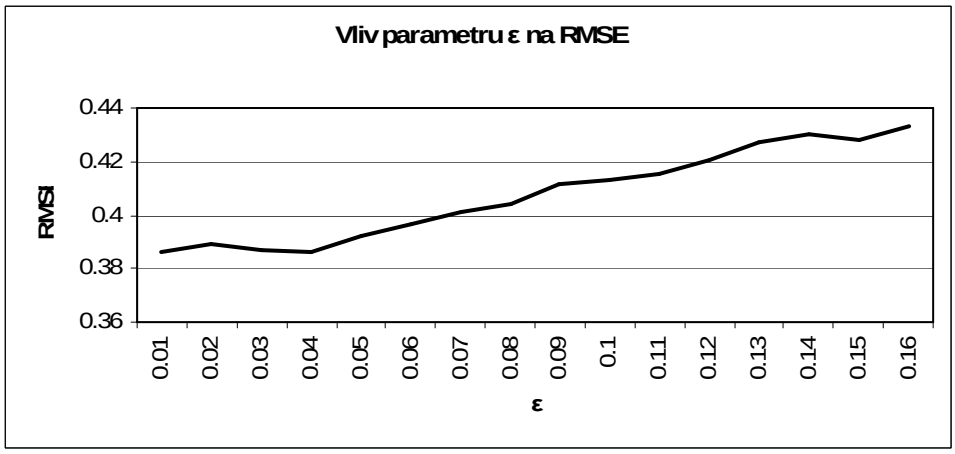
Vliv parametru C na $RMSE_{test}$ je znázorněn v grafu 3. Při základních nastavení ostatních parametrů chyba klesá až k určité hodnotě C a dále potom roste. Parametr C je však regulován spolu s ε a změna samotného C tak nemá až takový význam. Bylo dále vypozorováno, že při různém nastavení parametru γ se chyba modelu začne zvyšovat při různých hodnotách *C*. Většinou při hodnotě okolo *C=5*. Protože je při nejlepším nalezeném modelu hodnota γ většinou zcela odlišná od hodnoty 0.1, podoba křivky znázorněné v grafu 3 se může velmi lišit. $RMSE_{train}$ má v závislosti na parametru C podobný průběh jako $RMSE_{test}$. Graf 3 je vytvořen z krátkých dat a poměr trénovací a testovací množiny je zde 90:10.



Graf 3: Vliv parametru C *na* $RMSE_{test}$

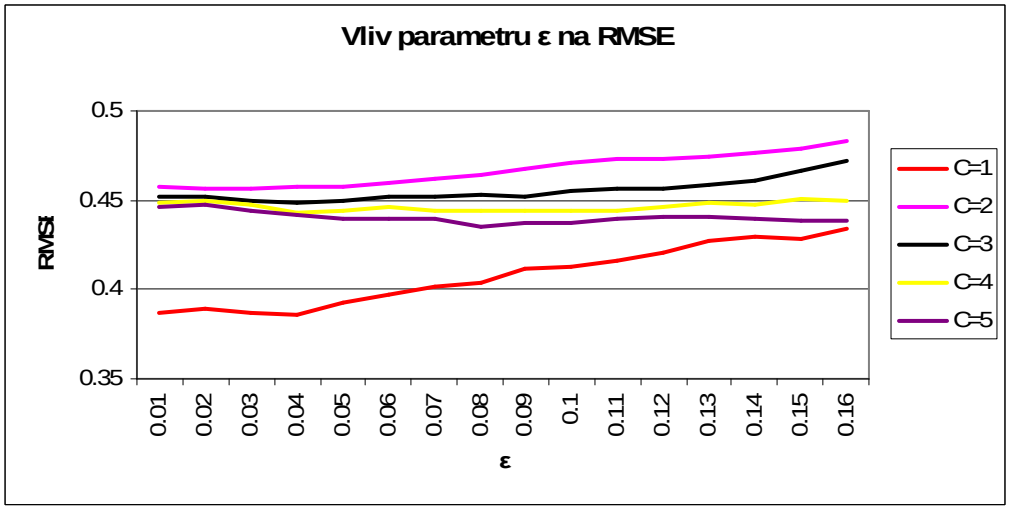
V případě parametru ε a jeho změn platí podobné poznatky jako u parametru C. Tyto dva parametry se regulují současně a na jejich vliv má i nastavení γ. Bylo vypozorováno, že chyba pro zvyšování ε od určité hodnoty má rostoucí průběh. Tato

hodnota je dána typem dat a nastavením ostatních parametrů. Z grafu 4 je patrné, že v uvedeném případě se chyba zvyšuje od hodnoty *ε=0.05*. U většiny nalezených ideálních modelů se nejvhodnější hodnota parametru pohybuje kolem *ε=0.1* a níže. Více o tom při hodnocení výsledků jednotlivých typů dat. Graf 4 je vytvořen z krátkých dat a poměr trénovací a testovací množiny je 50:50. Pro stejná data a stejné rozdělení množin je v grafu 5 zobrazen průběh chyby pro různá nastavení parametru *C*.



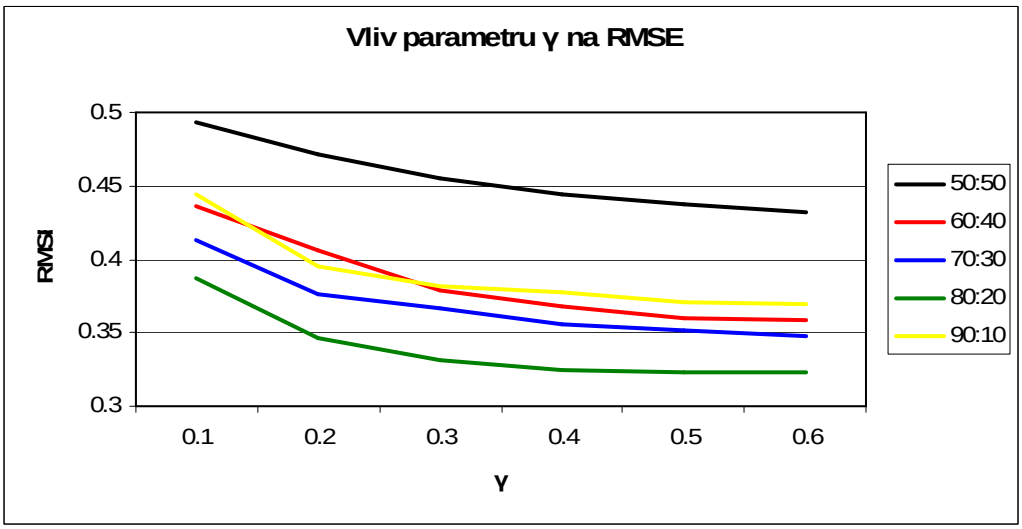
Graf 4: Vliv parametru ε na $RMSE_{test}$

Z grafu 5 je patrné, že průběh chyby při změnách *ε* se pro různé C odlišuje. Ve většině případů však chyba roste od hodnoty *ε =0.1*, v některých případech se chyba radikálně zvyšuje až při vyšších hodnotách, přibližně *ε =0.18*. Z tohoto zjištění vyplívá, že optimální hodnota *ε* je daná velmi malým rozsahem hodnot. Tento parametr tak byl správně při experimentech regulován v řádech setin.



Graf 5: Vliv parametru ε na $RMSE_{test}$ pro různé C

Posledním zkoumaným parametrem je γ. Jedná se o jádrový parametr, regulovaný v polynomickém i RBF jádru. Jak již bylo zobrazeno v grafu 1 (v předchozí kapitole), chyba má při regulaci tohoto parametru dost specifický průběh. Pokud je tento parametr měněn v kroku 0.1, má chyba do určité hodnoty parametru klesající tendenci a od určité hodnoty je velmi citlivá na změnu tohoto parametru. Jak je vidět ze zmiňovaného grafu, chyba se vrací k minimu při několika velmi odlišných hodnotách parametru γ. Až od určité hodnoty je chyba trendově rostoucí. Graf 6 zobrazuje jakýsi výřez ze zmiňovaného grafu1 předchozí kapitoly. Ukazuje průběh chyby při hodnotách $\gamma=0.1$ až $\gamma=0.6$ a to pro různé rozdělení trénovací a testovací množiny dat. Jak lze očekávat průběhy jsou při různých rozděleních velmi podobné. Chyba $RMSE_{test}$ je při dalším zvyšování γ stále menší až do určité hodnoty a poté je průběh podobný průběhu zobrazenému v grafu 1.



Graf 6: Vliv parametru γ na $RMSE_{test}$

Chyby pro zmíněné změny parametrů C, γ, i ε mají podobné průběhy pro SVR model s polynomickou i RBF jádrovou funkcí. Chyba $RMSE_{train}$ se pro RBF jádrovou funkci i polynomickou jádrovou funkci snižuje se zvyšujícím se γ. Dalo by se říci že konverguje k nule.

5.1.2 Vzájemné vlivy parametrů

Během provádění experimentů bylo na základě algoritmu zobrazeného v předchozí kapitole zjišťováno, jestli je při regulacích parametrů chyba modelu menší než poslední nalezená minimální chyba. Při těchto experimentech bylo náhodně

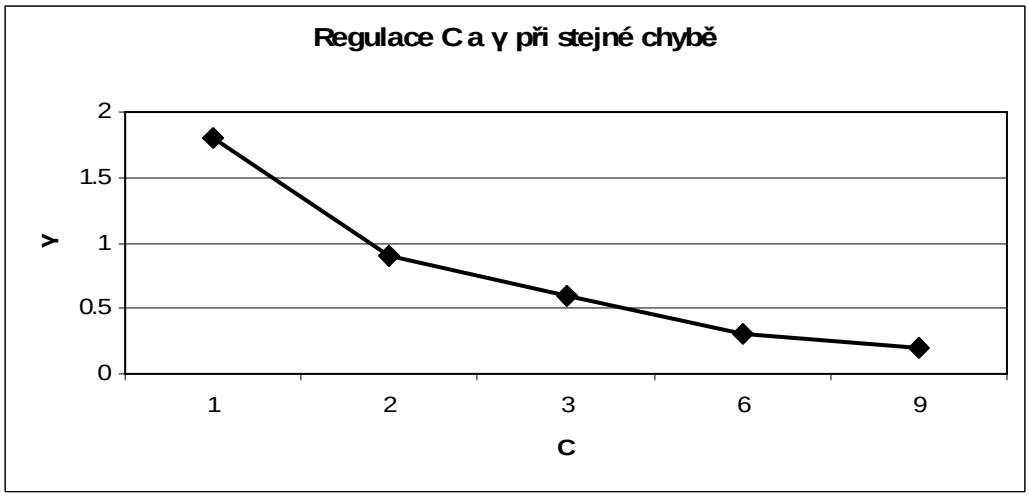
zjištěno, že při různých nastaveních parametrů několikrát vycházela stejná chyba. Bylo zjištěno pouze pro polynomickou jádrovou funkci, že pro konstantní hodnotu ε a vzájemnou regulaci parametrů C a γ vrací model stejnou chybu.

β	ε	C	γ	$RMSE_{test}$
1	0.01	1	1.8	0.313765
1	0.01	2	0.9	0.313765
1	0.01	3	0.6	0.313765
1	0.01	6	0.3	0.313765
1	0.01	9	0.2	0.313765

Tab.4: Hodnoty parametrů C a γ při chybě $RMSE_{test}=0.3137$

V tab.4 je zobrazeno, pro které C a γ vyšla stejná chyba. Vynásobením parametrů C a γ v kterémkoli řádku vychází stále stejná hodnota 1.8. Tento uvedený případ je pro střední data a rozdělení trénovací a testovací množiny v poměru 90:10. To samé pravidlo platí ale i pro jiný typ dat bez ohledu na rozdělení trénovací a testovací množiny. graf 7. je grafická interpretace tab.4. Je vidět, že pro každé snížení hodnoty parametru γ na polovinu je potřeba dvojnásobně zvýšit parametr C k udržení stejné chyby modelu.

Výše zmíněné pravidlo neplatí, pokud je parametr β různý od hodnoty 1. Protože však parametr β chybu vždy zvyšuje jak bylo výše popsané, každý ideální model musí mít hodnotu $\beta=1$. Z toho také vyplývá, že ideální model SVR s polynomickým jádrem může mít několik různých struktur parametrů.



Graf 7: Hodnoty parametrů C a γ při chybě $RMSE_{test}=0.3137$

5.2 Krátká časová řada

První ze zkoumaných dat jsou data krátké časové řady. Tato data obsahují pouze 264 záznamů, což je poměrně malé množství pro dobré naučení běžné neuronové sítě. Na těchto datech by mělo být ověřeno, zda mají sítě SVM výhodu (díky principu na kterém fungují) oproti ostatním sítím ve schopnosti zevšeobecnění i na menším množství dat. Modely s nejlepší strukturou pro RBF jádro jsou uvedeny v tab.5.

C	ε	γ	$RMSE_{train}$	$RMSE_{test}$	O_{train}:O_{test}		jádro
					(%)	(poč.)	
18	0.1	1.2	0.297235	0.381980	50:50	132:132	RBF
14	0.06	0.26	0.367723	0.333278	60:40	158:106	RBF
15	0.09	0.09	0.388117	0.336405	70:30	184:80	RBF
1	0.08	2.38	0.355886	0.307760	80:20	211:53	RBF
2	0.08	5	0.299658	0.318915	90:10	237:27	RBF

Tab.5: Nejlepší struktury parametrů pro krátkou časovou řadu a RBF jádro

Jak je vidět z tab.5, $RMSE_{test}$ je klesající až k poměru trénovací a testovací množiny 80:20 a dále je rostoucí. Je vidět, že i při rozdělení množin 50:50 a poměrně malém množství dat v trénovací množině je chyba stále přijatelná. První tři struktury parametrů se vyznačují především nezvykle velkou hodnotou parametru *C*, což je výjimka mezi všemi experimenty. Hodnota parametru ε se stabilně pohybuje mezi 0.06 a 0.1. Parametr γ byl zde regulován i v řádech setin a jeho optimální hodnota je velmi rozdílná při různých rozdělení množin.

Nejlepší chybu vykazuje struktura při rozdělení množin v poměru 80:20 a s hodnotami parametrů $C=1$, $\varepsilon=0.08$ a $\gamma=2.38$.

Další experimenty s krátkou časovou řadou byly provedeny pro polynomickou jádrovou funkci. Výsledné nejlepší struktury parametrů jsou uvedeny v tab.6. Stupeň polynomické funkce β má pokaždé hodnotu 1 jak bylo popsáno v předchozích částech práce. Optimální Parametry C jsou zde oproti RBF jádru v menším rozsahu a pohybují se v rozsahu od $C=1$ až $C=7$. Optimální hodnoty parametru ε se pohybují ve stejných mezích jako v případě RBF jádra. Hodnoty parametru γ jsou opět velmi odlišné pro jednotlivé struktury. Byly zde ale voleny pouze v řádu desetin.

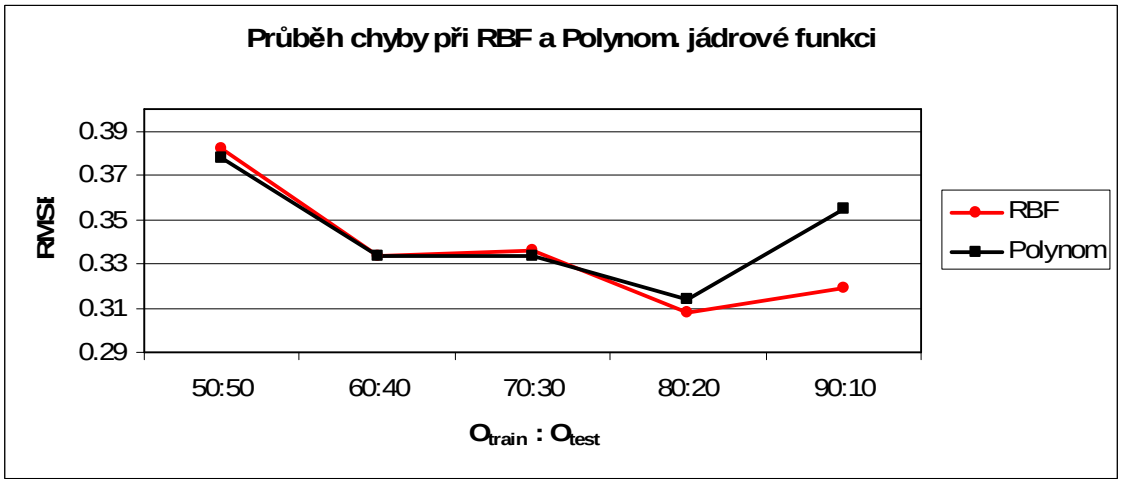
Optimální model je zde opět pro rozdělení množin v poměru 80:20 a jeho struktura parametrů má hodnoty: $\beta=1$, $C=7$, $\varepsilon=0.07$ a $\gamma=0.3$.

β	C	ε	γ	$RMSE_{train}$	$RMSE_{test}$	O_{train}:O_{test}		jádro
						(%)	(poč.)	
1	3	0.1	4.5	0.332602	0.377518	50:50	132:132	Polynom.
1	5	0.08	1.1	0.388955	0.333068	60:40	158:106	Polynom.
1	5	0.08	0.7	0.387798	0.333247	70:30	184:80	Polynom.
1	7	0.07	0.3	0.396470	0.314204	80:20	211:53	Polynom.
1	1	0.09	2	0.381650	0.354472	90:10	237:27	Polynom.

Tab.6: Nejlepší struktury parametrů pro krátkou časovou řadu a polynomické jádro

V grafu 8 je znázorněn průběh $RMSE_{test}$ pro obě jádrové funkce. Je vidět, že optimální modely vykazují téměř stejné chyby s oběma jádrovými funkcemi. Chyba se mírně liší až při rozdělení množin v poměru 90:10.

Vhodná rozdělení množin jsou pro tato data i v poměru 60:40 a 70:30, což potvrzuje skutečnost, že by velikost trénovacího vzorku dat měla být přibližně dvě třetiny ze všech vzorků dat.



Graf 8: Průběh $RMSE_{test}$ při různém rozdělení trénovací a testovací množiny a pro krátkou časovou řadu

5.3 Střední časová řada

Pro střední časovou řadu a RBF jádrovou funkci vycházel ideální parametr C v poměrně úzkém rozsahu, v hodnotách od 1 do 5. I parametr ε se v ideálních strukturách pohybuje v rozmezí 0.1 a níže podobně jako v případě dat krátké časové řady. Hodnota parametru γ se zde zvyšuje s velikostí trénovací množiny, jeho hodnoty

jsou v řádu desetin mezi číslem 1.3 a 4.5. Výjimkou je však rozdělení množin v poměru 90:10, kde má ideální γ hodnotu 12.5.

Nejlepších výsledků sice dosahuje model při rozdělení množiny v poměru 90:10, ale dá se předpokládat, že při těchto hodnotách počtu trénovacích a testovacích dat dochází k poklesu schopnosti zevšeobecnění. Jinak řečeno, velké množství trénovacích dat zapříčinilo naučení se sítě tak, že část malého množství testovacích dat už nepředstavuje pro síť nová data. Je těžké určit ve kterém bodě přesně dochází ke ztrátě zevšeobecnění. Malou $RMSE_{test}$ vykazuje i model s rozdělením dat v poměru 80:20, ale v tomto bodě již dochází také k poklesu $RMSE_{test}$. Ideální se pro to jeví model s rozdělením dat 60:40 a s hodnotami parametrů *C=5*, *ε=3.1* a *γ=3.1*. Jak je vidět z grafu 9, při zvyšování i snižování počtu trénovacích dat od 60% (288 dat) mírně roste $RMSE_{test}$. Proto se dá předpokládat, že síť má v tomto bodě stále dobrou schopnost zevšeobecnění. Výsledné struktury pro RBF jádro jsou zobrazeny v tab.7.

C	ε	γ	$RMSE_{train}$	$RMSE_{test}$	O_{train}:O_{test}		jádro
					(%)	(poč.)	
5	0.09	1.3	0.311843	0.367592	50:50	240:240	RBF
5	0.08	3.1	0.301760	0.365167	60:40	288:192	RBF
2	0.07	4.2	0.298725	0.376794	70:30	336:144	RBF
3	0.04	4.5	0.318270	0.346648	80:20	384:96	RBF
1	0.1	12.5	0.292217	0.276251	90:10	432:48	RBF

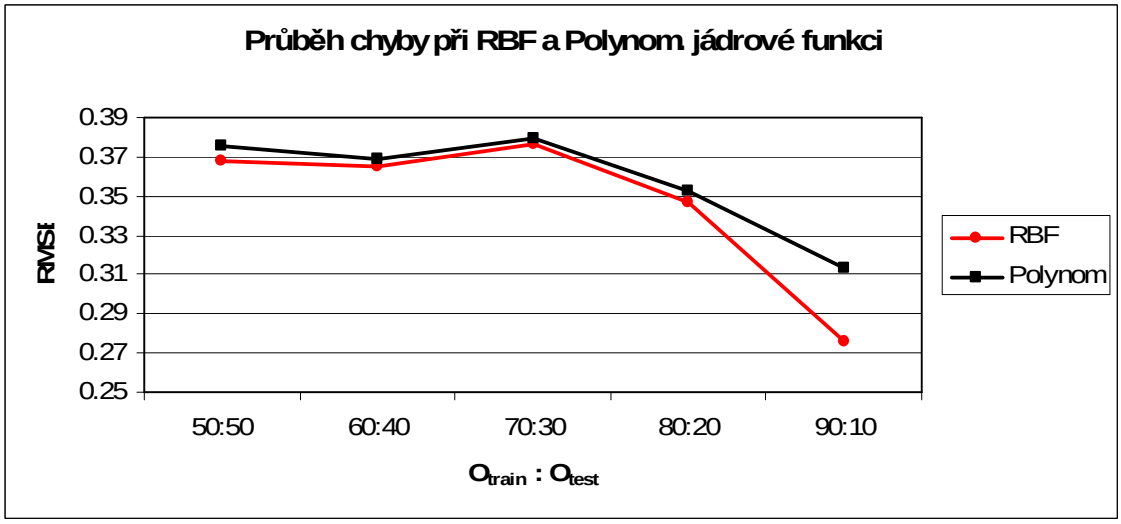
Tab.7: Nejlepší struktury parametrů pro střední časovou řadu a RBF jádro

Zhodnocení výsledků pro polynomickou jádrovou funkci je následující: Parametr *β* má opět ve všech strukturách hodnotu 1. Parametr *C* se pohybuje v podobném rozsahu jako v případě RBF jádrové funkce. Parametr *ε* má pro rozdělení množin 90:10 velmi nízkou hodnotu 0.01, pro rozdělení množin 50:50 naopak vyšší hodnotu (0.14) než je obvyklé. Parametr *γ* má o něco vyšší hodnotu při rozdělení množin 50:50, jinak jde o standardní rozsah v řádu desetin. Výsledné struktury pro polynomické jádro jsou zobrazeny v tab.8.

β	C	ε	γ	$RMSE_{train}$	$RMSE_{test}$	O_{train}:O_{test}		jádro
						(%)	(poč.)	
1	1	0.14	7.9	0.340357	0.375982	50:50	240:240	Polynom.
1	2	0.09	4.9	0.357692	0.369172	60:40	288:192	Polynom.
1	4	0.11	1.9	0.352754	0.379304	70:30	336:144	Polynom.
1	1	0.1	2.9	0.366115	0.352302	80:20	384:96	Polynom.
1	1	0.01	1.8	0.402541	0.313765	90:10	432:48	Polynom.

Tab.8: Nejlepší struktury parametrů pro střední časovou řadu a polynomické jádro

Celkové hodnocení je pro polynomickou jádrovou funkci podobné jako RBF funkci. Z grafu 9 je vidět průběh chyby $RMSE_{test}$ při různých rozdělení množin, který je pro obě jádrové funkce opět téměř stejný.



Graf 9: Průběh $RMSE_{test}$ při různém rozdělení trénovací a testovací množiny pro střední časovou řadu

5.4 Dlouhá časová řada

Posledním typem zkoumaných dat jsou data dlouhé časové řady. Pro RBF jádro jsou ideální hodnoty parametru C opět ve standardním rozsahu mezi 1 až 10. Ideální hodnoty parametru ε jsou v případě rozdělení dat v poměru 50:50 až 70:30 o něco vyšší než je běžné. Hodnoty parametru γ jsou velmi odlišné pro různé rozdělení množin dat. Některé hodnoty byly regulovány až do řádů setin, protože v případě rozdělení dat 50:50 a 70:30 má parametr γ hodnotu menší než 0.1, což není příliš obvyklé. Zajímavé je, že hodnoty tohoto parametru jsou pro různá rozdělení dat velmi podobná hodnotám

pro téže rozdělení v datech krátké časové řady. Výsledky pro RBF jádro a dlouhou časovou řadu jsou zobrazeny v tab.9.

C	ε	γ	$RMSE_{train}$	$RMSE_{test}$	O_{train}:O_{test}		jádro
					(%)	(poč.)	
3	0.15	0.07	0.402381	0.391498	50:50	376:376	RBF
1	0.17	0.21	0.401820	0.393978	60:40	451:301	RBF
6	0.17	0.03	0.397067	0.401710	70:30	526:226	RBF
3	0.1	4.5	0.384182	0.402021	80:20	601:151	RBF
1	0.1	6	0.381599	0.357901	90:10	676:76	RBF

Tab.9: Nejlepší struktury parametrů pro dlouhou časovou řadu a RBF jádro

Pro hodnocení lze použít podobný úsudek jako v případě střední časové řady. Nejlepší $RMSE_{test}$ sice vyšla pro rozdělení dat 90:10, ale při tomto rozdělení dochází k poklesu schopnosti zevšeobecnění a to ještě výraznějšímu než ve střední časové řadě. Toto rozdělení množin lze proto zanedbat. Z ostatních rozdělení dat vychází $RMSE_{test}$ velmi vyrovnaná a nejlepší struktura parametrů modelu je při rozdělení 50:50 (jen nepatrně horší je 60:40). Ideální parametry mají (mimo rozdělení 90:10) následující podobu: *C=3, ε=0.15* a *γ=0.07.*

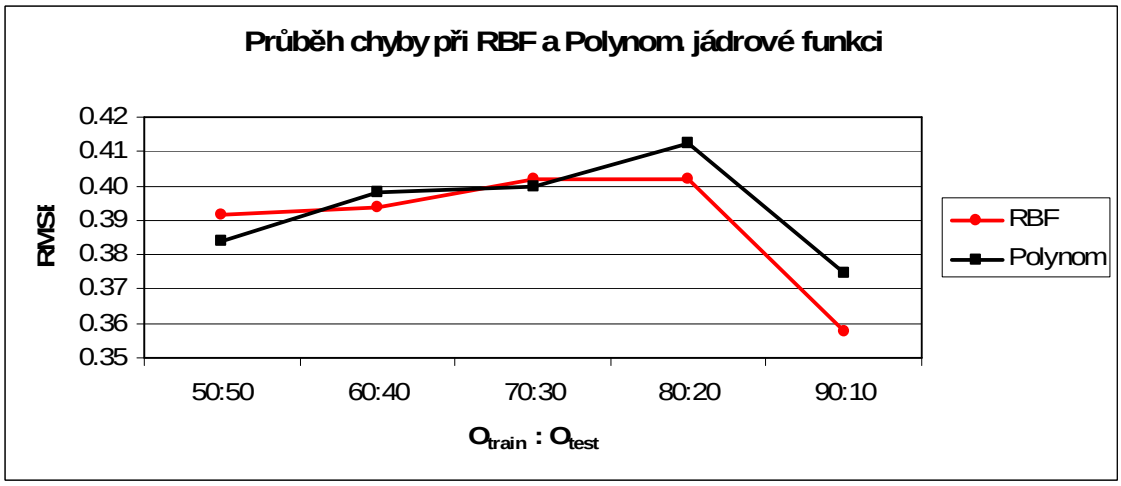
Pro data dlouhé časové řady a polynomickou jádrovou funkci vyšla ve všech případech hodnota ideálního parametru *β=1*, hodnota ideálního parametru *C=1*, hodnota ideálního parametru *ε* je většinou 0.1 a hodnota parametru *γ* je opět různá, viz. tab.10. Zajímavé jsou zde především velmi podobné hodnoty některých parametrů.

β	C	ε	γ	$RMSE_{train}$	$RMSE_{test}$	O_{train}:O_{test}		jádro
						(%)	(poč.)	
1	1	0.1	7	0.409955	0.384019	50:50	376:376	Polynom.
1	1	0.1	3.2	0.390890	0.397978	60:40	451:301	Polynom.
1	1	0.1	2.7	0.393647	0.399705	70:30	526:226	Polynom.
1	1	0.11	2	0.394954	0.412362	80:20	601:151	Polynom.
1	1	0.15	0.3	0.399837	0.374378	90:10	676:76	Polynom.

Tab.10: Nejlepší struktury parametrů pro dlouhou časovou řadu a polynomické jádro

Nejlepší struktura parametrů modelu vyšla jako v případě RBF jádra při rozdělení 50:50 a ideální hodnoty parametrů jsou: *β=1, C=1, ε=0.1* a *γ=7*. Průběh chyb obou jádrových funkcích zobrazuje graf 10. Průběhy se tentokrát mírně liší. Výrazný

pokles chyby je u obou funkcí vidět mezi rozdělením 80:20 a 90:10. Polynomická funkce dosahuje mírně lepších výsledků při rozdělení množiny dat 50:50 a 70:30, při jiných poměrech trénovací a testovací množiny vykazuje lepší výsledky model s RBF jádrovou funkcí.

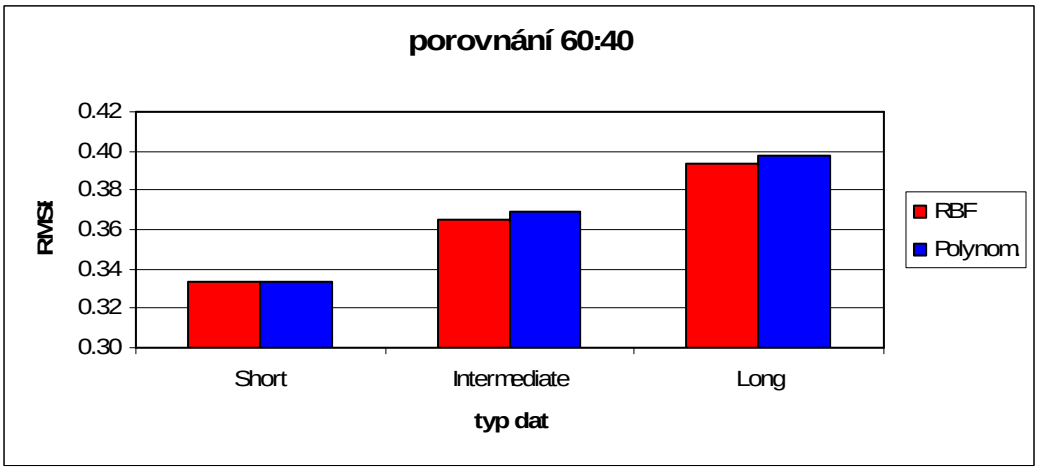


Graf 10: Průběh $RMSE_{test}$ při různém rozdělení trénovací a testovací množiny pro dlouhou časovou řadu

5.5 Porovnání výsledků

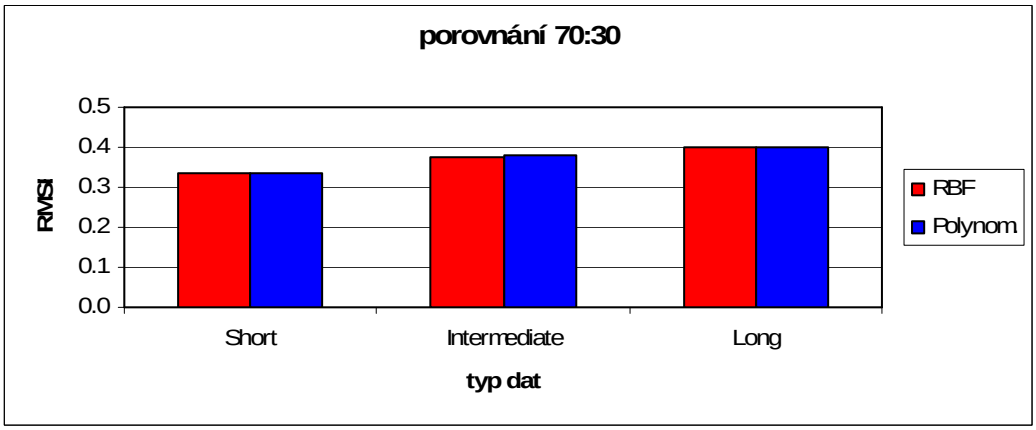
Po provedení experimentů s krátkou, střední a dlouhou časovou řadou lze ze získaných výsledků provést několik závěrů. První, který byl už zmíněn je ztráta schopnosti zevšeobecnění při velkém procentuálním množství trénovacích dat oproti testovacím datům. Z tohoto důvodu mezi porovnávané varianty není zařazeno rozdělení množin v poměru 90:10. Vynechána je i varianta rozdělení dat v poměru 50:50 a následuje porovnání zbylých tří možných rozdělení dat.

Protože byly parametry voleny experimentálně uživatelem a nebyla použita metoda cross-validation, je možné, že existuje v některém z prezentovaných nejlepších modelů jiná struktura optimálních parametrů a s ní i menší $RMSE_{test}$. Následující grafy však prezentují pro různé časové řady stejný průběh chyb při různých rozděleních dat a navíc podobnou $RMSE_{test}$ při použití dvou jádrových funkcí. To vede k závěru, že nejlepší nalezené modely jsou správné.



Graf 11: Výsledné chyby při rozdělení v poměru 60:40

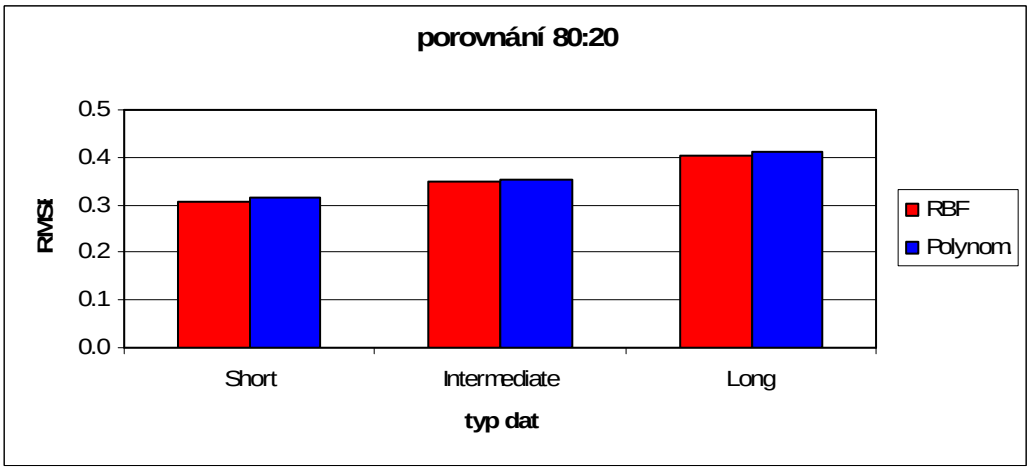
Z grafu 11 je patrný průběh chyby při rozdělení dat v poměru 60:40 a pro různě dlouhé časové řady. Modely s nejlepšími výsledky vykazují rostoucí chybu se zvyšujícím se počtem dat a to v případě obou jádrových funkcí. Při tomto rozdělení vykazuje mírně lepší výsledky model s polynomickou jádrovou funkcí.



Graf 12: Výsledné chyby při rozdělení v poměru 70:30

Pro rozdělení dat v poměru 70:30 (jak ukazuje graf 12) je průběh hodnoty chyby opět rostoucí se zvyšujícím se počtem dat. Je zajímavé, že v tomto případě vykazují modely s polynomickou i RBF jádrovou funkcí téměř stejné chyby ve všech typech dat i když nalezené struktury parametrů optimálních modelů jsou u obou jádrových funkcí zcela odlišné.

I rozdělení dat v poměru 80:20 vykazuje rostoucí chybu s rostoucím počtem dat. Zde je opět nepatrně lepší výsledek v případě polynomické jádrové funkce. Názorně je to zobrazeno v grafu 13.



Graf 13: Výsledné chyby při rozdělení v poměru 80:20

V případě zde neprezentovaných výsledků variant rozdělení dat v poměru 50:50 a 90:10 není potvrzen závěr zvyšující se chyby s rostoucím množstvím dat. Průběhy $RMSE_{test}$ v závislosti na délce dat jsou v těchto případech zcela jiné, což potvrzuje, že tyto varianty rozdělení dat nejsou příliš vhodné ke kvalitnímu naučení sítě správně reagující na neznámé vstupy.

Již bylo zmíněno, že každý parametr je regulovaný v určitém rozsahu a také to, že v některých optimálních modelech tyto parametry nabývají nezvykle vysokých nebo nízkých hodnot. S opomenutím těchto extrémních případů lze na základě experimentů shrnout hodnoty rozsahů jednotlivých parametrů pro nalezení kvalitního modelu následovně:

- Ideální hodnota parametru β je 1.
- Ideální hodnota parametru γ je v intervalu $<0.1,7>$.
- Ideální hodnota parametru C je v intervalu $<1,7>$.
- Ideální hodnota parametru ε je v intervalu $<0.01,0.15>$.

Na závěr bylo prověřeno, jestli jsou všechny vstupní parametry vhodné a jestli přispívají k menší chybě modelu. Postupně byly jednotlivé vstupní parametry odebírány a sledován vliv tohoto kroku na výsledek predikce. Při absenci kteréhokoliv vstupního parametru došlo ke zhoršení predikce a zvýšení $RMSE_{test}$. Všechny vstupní parametry proto byly zvoleny vhodně a mají kladný vliv na výsledky predikce.

6 Závěr

V diplomové práci je popsáno využití Web miningových technik v prostředí Internetu. Dále je charakterizována metoda Support Vector Machines a princip jejího fungování při klasifikaci a predikci.

Získaná data z Web miningového nástroje Google Analytics jsou analyzována. Je popsáno jakými metodami byla data předzpracována, jaké jsou jejich charakteristiky a co z nich lze vyvodit. Podařilo se pro různě velké datové vzorky dokázat, že neuronová síť Support Vector Machine je schopna se dobře naučit i při menším množství trénovacích dat. Podařilo se pro různé rozdělení trénovacích a testovacích dat dokázat, kdy síť vykazuje nejlepší zevšeobecňující schopnosti a naopak, který typ rozdělení dat je pro správné naučení sítě již nevhodný. Na základě experimentů byly zjištěny rozsahy jednotlivých parametrů, které jsou vhodné pro vytvoření správných modelů poskytujících kvalitní výsledky.

Modelování bylo provedeno pro jeden ze dvou možných typů regrese a pro dvě zvolené jádrové funkce. Byly nalezeny optimální modely pro různě velké vzorky dat, pro různé rozdělení dat na trénovací a testovací množinu a pro dvě zvolené jádrové funkce. Bylo zjištěno, že modely Support Vector Machine dosahují pro obě jádrové funkce přibližně stejných výsledků. Volba pěti metod předzpracování dat byla vybrána vhodně a to na základě dosahovaných chyb modelu a na základě zvýšení chyby modelu při absenci některého ze vstupů. Support Vector Machine se osvědčila jako vhodná metoda pro predikci tohoto typu dat.

Zdroje

[1] BING, L. *Web Data Mining : Exploring Hyperlinks, Contents, and Usage Data*. [s.l.] : Hardcover, 2007. 532 p. ISBN 978-3-540-37881-5.

[2] Cooley, R., Mobasher B., Srivastava J. *Web Mining: Information and Pattern Discovery on the World Wide Web*. [s.l.] : [s.n.], 1997. 10 p.

[3] *UCLA Anderson School of Management* [online]. 1996 [cit. 2011-02-15]. Data mining: What is Data mining?. Dostupné z WWW: <http://www.anderson.ucla.edu/faculty/jason.frand/teacher/technologies/palace/datamining.htm>.

[4] *InfoVis.net* [online]. 2005-09-18 [cit. 2011-02-15]. Web mining. Dostupné z WWW: <http://www.infovis.net/printMag.php?lang=2&num=172>.

[5] SCIME, A. *Web Mining: Applications and Techniques*. [s.l.] : Idea Group Inc, 2005. 427 p.

[6] XU, G., ZHANG, Y., LI, L. *Web Mining and Social Networking : Techniques and Applications*. [s.l.] : [s.n.], 2010. 210 p.

[7] *Website Optimisation UK* [online]. 2011 [cit. 2011-02-16]. Authority Development, Hub and Authority Model Optimisation. Dostupné z WWW: <http://www.kolberg.co.uk/authority_development.php>.

[8] SRIVASTAVA, J., et al. *Web Usage Mining : Discovery and Applications of Usage Patterns from Web Data* [online]. [s.l.] : [s.n.], 2000 [cit. 2011-02-18]. Dostupné z WWW: <http://www.sigkdd.org/explorations/issues/1-2-2000-01/srivastava.pdf>.

[9] KVASNIČKA, V., et al. *Úvod do teórie neurónových sietí*. Bratislava : IRIS, 1997. 285 s. ISBN 80-88778-30-1.

[10] HAYKIN, S. *Neural Networks : A Comprehensive Foundation*. 2nd ed. [s.l.] : Prentice Hall, 1999. 842 p. ISBN 0132733501, 9780132733502.

[11] *Support Vector Machines (SVM)* [online]. [s.l.] : [s.n.], [2005] [cit. 2011-02-19]. Dostupné z WWW: <http://is.muni.cz/el/1433/podzim2006/PA034/09_SVM.pdf>.

[12] BOSWELL, D. *Introduction to Support Vector Machines* [online]. [s.l.] : [s.n.], 2002-8-6 [cit. 2011-02-20]. Dostupné z WWW: <http://dustwell.com/PastWork/IntroToSVM.pdf>.

[13] *Institute of Microbial Technology* [online]. [2005] [cit. 2011-02-20]. Algorithm of Rice Blast Prediction. Dostupné z WWW: <http://www.imtech.res.in/raghava/rbpred/algorithm.html>.

[14] SEWELL, M. *VC Dimension* [online]. [s.l.] : [s.n.], 2008 [cit. 2011-02-25]. Dostupné z WWW: <http://www.svms.org/vc-dimension/vc-dimension.pdf>.

[15] FAIGL, J., SVOBODA, L., ŽOLDÁK, M. *Support Vector Machine* [online]. [s.l.] : [s.n.], 6.6.2001 [cit. 2011-02-25]. Dostupné z WWW: <http://cmp.felk.cvut.cz/cmp/courses/recognition/zapis_prednasky/zapis_01/8-9-11/rpz8-9-11.pdf>.

[16] VAPNIK, V. *The Nature of Statistical Learning Theory*. [s.l.] : Springer, 2000. 314 p. ISBN 0387987800.

[17] SEWELL, M. *Structural Risk Minimization* [online]. [s.l.] : [s.n.], 2008 [cit. 2011-02-25]. Dostupné z WWW: <http://www.svms.org/srm/srm.pdf>.

[18] HÁJEK, P., OLEJ, V. *Municipal Creditworthiness Modelling by Kernel-Based Approaches with Supervised and Semi-supervised Learning*. Proc. of the 11th International Conference on Engineering Applications of Neural Networks, EANN 2009, Communications in Computer and Information Science, Engineering Applications of Neural Networks, London, Palmer-Brown, D., Draganova, Ch., Pimenidis, E., Mouratidis, H., Eds., 27-29 august, Springer Berlin Heidelberg New York, 2009, pp.35-44, ISSN 1865-0929, ISBN 978-3-642-03968-3.

[19] *STATISTICA* [online]. 2007 [cit. 2011-03-01]. Support Vector Machines (SVM). Dostupné z WWW: <http://www.statsoft.com/textbook/>.

[20] *Neural Networks Forecasting* [online]. 2005 [cit. 2011-03-02]. SVM Support Vectors. Dostupné z WWW: <http://www.neural-forecasting.com/support_vector_machines.htm>.

[21] DEBASISH, B., SRIMANTA, P., PATRANABIS, D. Support Vector Regression. In *Neural Information Processing : Letters and Reviews* [online]. [s.l.] : [s.n.], 2007 [cit. 2011-03-06]. Dostupné z WWW: <http://bsrc.kaist.ac.kr/nip-lr/V11N10/V11N10P1-203-224.pdf>.

[22] CASTILLO, O., MELIN, P., ROSS, O. *Theoretical Advances and Applications of Fuzzy Logic and Soft Computing*. [s.l.] : [s.n.], 2007. 895 p. ISBN 3540724338, 9783540724339.

[23] Olej V., Hajek P., Filipova J., "*Modelling of Web Domain Visits by IF-Inference System*," WSEAS Transactions on Computers, vol. 9, no. 10, pp. 1170–1180, 2010.

[24] *Google Analytics* [online]. 2011 [cit. 2011-03-15]. Google Analytics. Dostupné z WWW: <http://www.google.com/analytics>.

[25] *ET NETERA* [online]. [2010] [cit. 2011-03-15]. Google Analytics. Dostupné z WWW: <http://www.etnetera.cz/cz/google-analytics/index.html>.

[26] RZEMPOLUCK, E. *Neural Network Data Analysis using Simulnet*. [s.l.] : [s.n.], 1998. 226 p. ISBN 0387982558.

Seznam Obrázků

Seznam Tabulek

Seznam Grafů

Seznam Zkratek

CMA	- Central Moving Average
DES	- Double Exponential Smoothing
HITS	- Hyperlink Induced Topic Search
IP	- Internet Protocol
IR	- Information Retrieval
MISO	- Multiple Input Single Output
MM	- Moving Median
MSE	- Mean Squared Error
NS	- Neuronová Síť
RBF	- Radial Basis Function
RMSE	- Root Mean Squared Error
SES	- Simple Exponential Smoothing
SMA	- Simple Moving Average
SSE	- Sum Squared Error
SVM	- Support Vector Machines
SVR	- Support Vector Regression
VC	- Vapnik Chervonenkis
WCM	- Web Content Mining
WSM	- Web Structure Mining
WUM	- Web Usage Mining

Seznam Symbolů

α - Lagrangeův multiplikátor

β - parametr jádrové polynomické funkce

γ - parametr jádrové funkce

ε - nastavovaný parametr SVR

ξ - volná proměnná (slack variable)

ρ - rozpětí separace (margin)

φ_j - nelineární transformace

τ - trénovací množina

Φ - ztrátová funkce

b - bias

b_0 - optimální bias

C - regulační parametr

d_i - požadovaná výstupní hodnota

e - šum

$f_{t,d}$ - četnost termu

I - indikační funkce

K - jádrová funkce

L - ztrátová funkce

m_0 - dimenze vstupního prostoru

m_1 - dimenze charakteristického prostoru

r - algebraická vzdálenost

R_{emp} - empirické riziko

s_j - směrodatná odchylka

t - term

v - nastavovaný parametr SVR

w - váhový vektor

w_0 - optimální váhový vektor

x - vektor vstupních dat

x_i - i-tý příklad z vektoru vstupních dat

xp - normální zobrazení vstupního vektoru na optimální nadrovinu

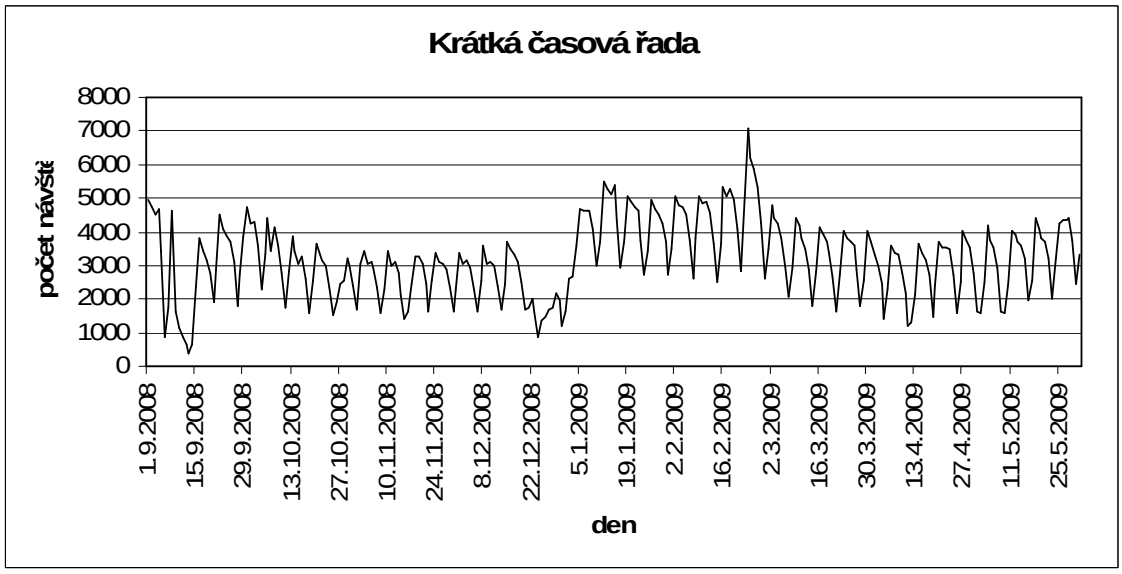
y - výstupní hodnota

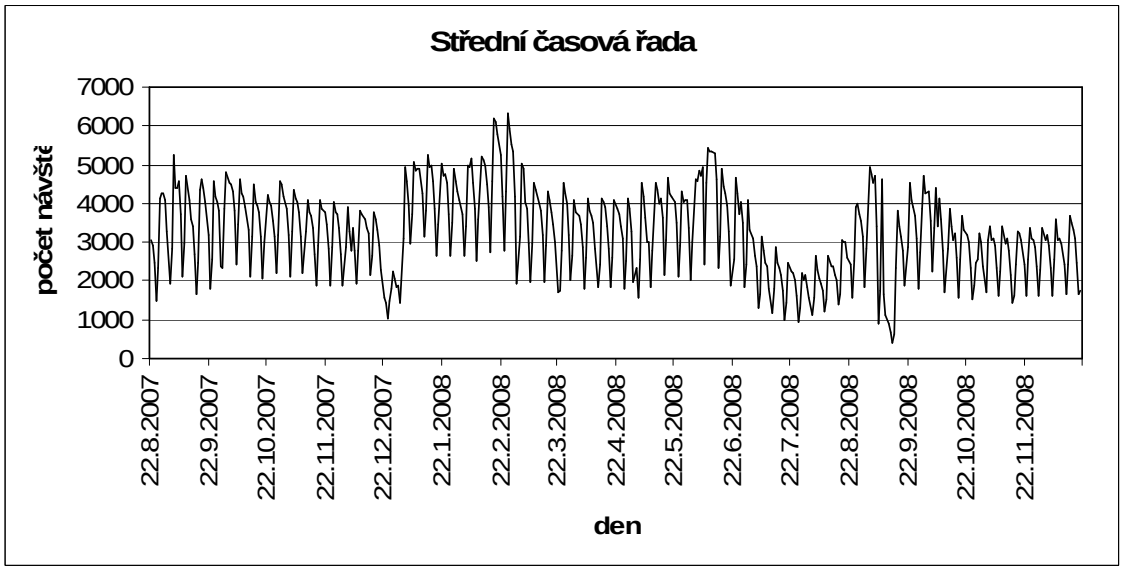
$\hat{y}$ - predikovaná hodnota

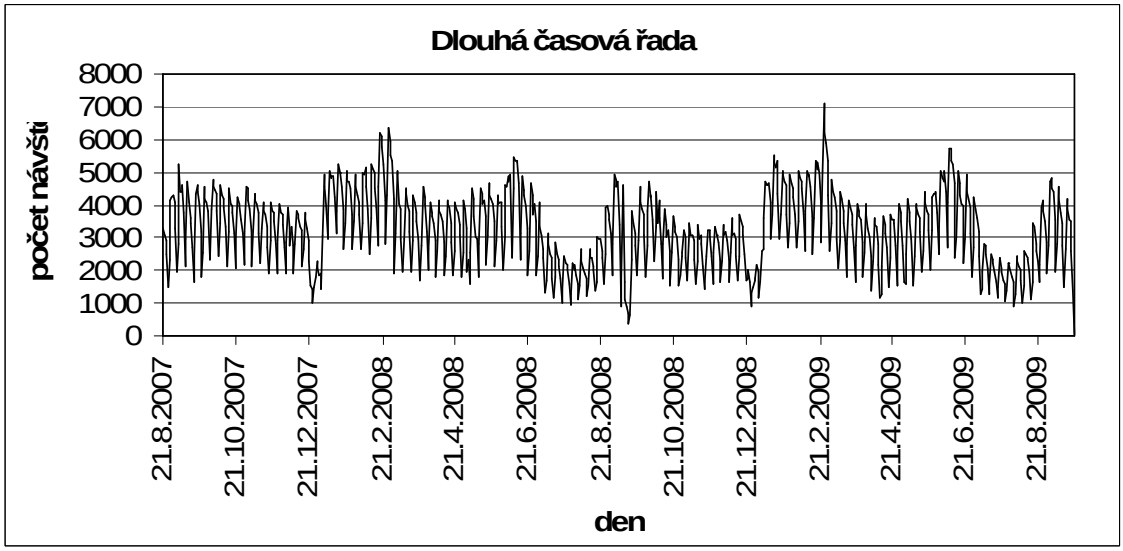
z_{ij} - standardizovaná hodnota

Přílohy

Příloha 1: Průběhy časových řad






Dlouhá časová řada
počet návštěv
8000
7000
6000
5000
4000
3000
2000
1000
0
21.8.2007
21.10.2007
21.12.2007
21.2.2008
21.4.2008
21.6.2008
21.8.2008
21.10.2008
21.12.2008
21.2.2009
21.4.2009
21.6.2009
21.8.2009
den