



Vzhledem k vzrůstající popularitě neuronových sítí jsme se rozhodli Vám je v tomto článku představit a říci si něco o jejich využití. Co si tedy představit pod pojmem neuronová síť?

Neuronová síť

Neuronová síť je algoritmus, který si bere za vzor činnost lidského mozku. Již v dřívějších dobách bylo zjištěno, že mozek je tvořen velkým množstvím vzájemně propletených buněk, které nazýváme neurony, jež spolu komunikují pomocí elektrických impulzů.

Od vzniku prvních počítačů se programátorské kapacity snaží vytvořit algoritmus, který bude činnost lidského mozku napodobovat. Vzniká tak pojem umělé inteligence, který výsledky takového snažení někdy více a někdy méně reprezentuje. Princip neuronových sítí je v dnešní době implementován v řadě dostupných analytických a rozhodovacích softwarů a v různých oborech lidské činnosti podává extrémně dobré výsledky, tedy ve srovnání se „standardními“ typy rozhodovacích algoritmů.



Princip sítě

Princip softwarové neuronové sítě, jak už bylo řečeno a je odhadnutelné z názvu, se inspiruje biologickou neuronovou sítí, kde je základním stavebním kamenem (zjednodušeně řečeno) nervová buňka – neuron.

Jednotlivé neurony jsou vzájemně propojeny spoji ohodnocenými vahami. Takovéto propojení a schopnost tyto váhy adaptovat (učit se) na základě trénovacích vzorů v datech, dává neuronové síti nové široké možnosti v oblasti analýzy dat.

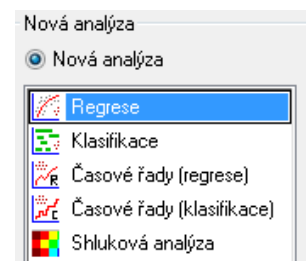
Jak již bylo řečeno, hlavní předností neuronové sítě je schopnost učit se, tedy zapamatovat si kombinace, které vedly k požadovanému výstupu a u nových vstupů se potom obracet na „svou“ paměť a na základě zkušeností odhadovat nový výsledek. V tomto případě mluvíme o generalizaci (zevšeobecnování), která je další velkou předností algoritmu neuronových sítí. Zjednodušeně řečeno, jde o přiměřenou dovednost správně zareagovat i na vstupy, které nebyly součástí trénovacích dat, a vyvodit z nich obecné závěry o datech. Schopnost učit se bývá někdy dokonce považována za definici umělé inteligence.

Často v praxi nemáme ani dostatek apriorních znalostí o tom, co způsobuje konkrétní variabilitu zkoumané proměnné, a není proto jiná možnost než se vlivy snažit popsat na základě historický naměřených dat. Učení se z historických dat je dnes součástí mnoha oborů, včetně Data Miningu, do kterého v našem pojetí neuronové sítě řadíme.

Další vlastností neuronových sítí je schopnost řešit i silně nelineární úlohy. Využití neuronové sítě v analýze dat má smysl všude tam, kde selhávají „klasické“ modely jako například regrese. V některých případech se stává, že není možné najít nějakou jednoduchou matematickou funkci, která by vhodně postihla všechny vlivy, které variabilitu sledované proměnné ovlivňují. Potom jsou neuronové sítě vhodná alternativa. Neuronové sítě jsou také do jisté míry schopné pracovat s nepřesnými daty a šumy. My však doporučujeme provést vždy čištění a přípravu dat.

Možností aplikace neuronových sítí je celá řada, dají se využít na následující typy úloh.

Vpravo vidíme menu softwaru *STATISTICA*, podle kterého možnosti využití shrneme.



Regrese – regresní analýza se zabývá předpovídáním spojité proměnné na základě vstupů (spojitých či kategorických prediktorů – nezávislých proměnných).

Klasifikace – neboli zařazování do tříd. Na základě úrovně cílové proměnné a kombinace vstupů, které ke konkrétnímu výsledku vedou, bude vytvořen model, který dokáže klasifikovat nová data. Typickým příkladem jsou bankovní problémy, které řeší otázky bonity klienta, schopnost jeho splácení, ale také problémy typu poslat či neposlat pacienta na podrobné vyšetření, detekce spamu apod.

Časové řady (regrese) – slouží k modelování spojitych proměnných, které prochází vývojem v čase, resp. časových řad. V této situaci můžeme vybrat buď pouze jedinou závislou proměnnou, kde model bude vycházet ze zpožděných hodnot této časové řady, nebo máme možnost zvolit další proměnné, které budou tuto řadu vysvětlovat.

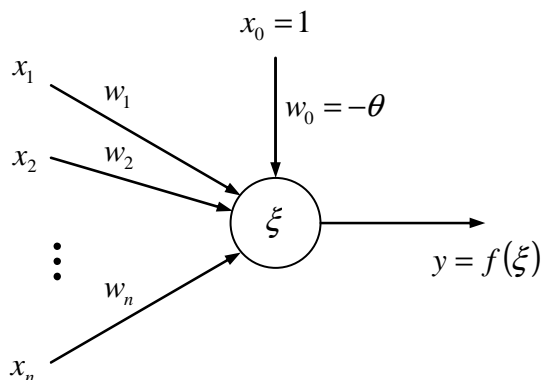
Časové řady (klasifikace) – tento typ analýzy použijeme tehdy, je-li naše cílová (závislá) proměnná kategorické povahy. Závislou proměnnou lze vysvětlovat opět pouze svým „historickým“ průběhem v čase, případně je možné zvolit další spojité i kategorické prediktory jako nezávislé vysvětlující proměnné.

Shluková analýza – tento typ analýzy nepoužívá závislou proměnnou (učení bez učitele), cílem je detekovat netriviální shluky v datech. Jde o tzv. Kohonenovu síť. Vstupem jsou pouze hodnoty (vstupních) nezávislých proměnných. Kohonenovu síť (též Self-organizing Feature Map – SOFM) chápeme jako druh shlukové analýzy (vektorovou kvantizaci).

Poznamenejme, že máme také možnost zvolit více závislých proměnných pro analýzu. Což znamená například možnost modelovat vícerozměrnou časovou řadu.

Model neuronu

Jak již bylo naznačeno, neuronové sítě se skládají z neuronů propojených vazbami. Model jednoho neuronu zachycuje obrázek níže:



Zdroj: Skripta STATISTICA NEURONOVÉ SÍTĚ (Doc. RNDr. Ing. Marcel Jiřina, Ph.D.)

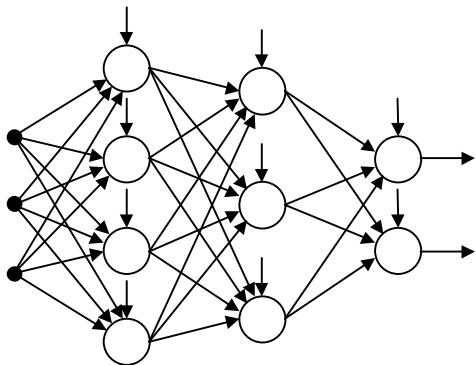
Model neuronu se skládá ze tří částí – vstupní, výstupní a funkční. Na základě vah mohou být jednotlivé vstupy potlačeny, nebo naopak zvýhodněny. Funkční část zpracuje informace ze vstupů a vygeneruje výstup. Výstupní část potom přivede výslednou informaci na vstup jiných neuronů.

Tedy výstup neuronu $f(\xi)$ je spočítán ve chvíli, když suma vstupů do neuronu x_i vynásobených jejich konkrétními vahami w_i překročí určitou hodnotu, kterou nazýváme práh (θ). Neuron lze popsat tímto způsobem: $y = f(\sum_{i=1}^n x_i \cdot w_i - \theta)$

Kde x_i je konkrétní hodnota na i -tém vstupu, w_i je potom váha tohoto vstupu, θ je prahová hodnota, n je celkový počet vstupů, f je transformační funkce a y hodnota na výstupu.

Skupina neuronů

Toto byl pouze jeden neuron, ten sám o sobě není schopen vykonávat o nic moc složitější funkci než klasická regresní analýza. Síla neuronové sítě se však projeví až při propojení neuronů mezi sebou do větších struktur. Neurony jsou uspořádány do vrstev. Nezapomeňme, že algoritmus se učí sám a všechny váhy si sám volí. Vzorem takové jednoduché neuronové sítě je například následující obrázek:



Příklad uspořádání třívrstvé NS se třemi vstupy a dvěma výstupy

Pokud si představíme, jak takovouto sítí proudí přichozí data, je nám jasné, že právě díky složitosti spojení dokáže neuronová sít najít i složitější a nelineární vztahy, na druhou stranu je ale pravda, že ze získané neuronové sítě nikdy nebudeme schopni získat interpretaci, proč to u konkrétního pozorování dopadlo, jak to dopadlo. Je to tedy metoda typu „black box“ – nejsme schopni jednoduše interpretovat výsledky či získat jednoduchý předpis závislosti mezi závislou a nezávislými proměnnými. Tato vlastnost nám vůbec nevadí, pokud nám jde pouze o předpověď veličiny v oblasti našeho zájmů, nicméně pokud chceme zjistit i důvody výsledku, pak nám tato metoda příliš nepomůže.

V modulu **STATISTICA Automatické neuronové sítě** máte na výběr několik typů sítí.

- › MLP - Vícevrstvou perceptronovou sít
- › RBF - Radial Basis Function)
- › SOFM - Kohonenovu sít
- › Lineární sít
- › Bayesovské sítě (PNN a GRNN)

Každý typ sítě má různé vlastnosti, výběr závisí na povaze úlohy a charakteru dat.

Hrozba přeučení sítě

Obecně platí, že pokud sít obsahuje malý počet neuronů, její schopnost vystihnou a popsat závislosti v trénovacích datech je slabší. Pokud bude sít naopak obsahovat příliš velký počet neuronů, tato sít pravděpodobně nebude mít problém navést a reprezentovat závislosti v trénovacích datech, ale její schopnost generalizace, tedy vystihnou správný výsledek na nových datech, může být horší. Takovému jevu se říká přeučení sítě (*overfitting*). K přeučení také může docházet ve chvíli, kdy model obsahuje velký počet vstupních parametrů a relativně málo pozorování. Cílem tedy není maximalizace výkonu sítě na trénovacích datech, ale rozumný kompromis mezi trénovacím výkonem a schopností zevšeobecnovat znalosti i na nových datech. S tím, o čem zde píšeme, úzce souvisí následující princip rozdělení datového souboru, který neuronové sítě implicitně používají.

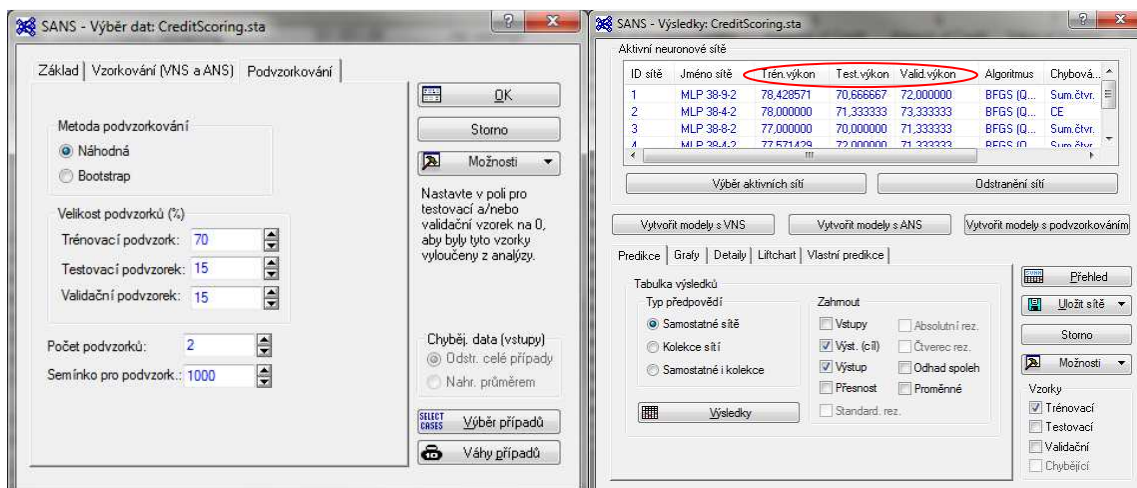
Jedná se o rozdělení dat na trénovací, testovací a případně validační množinu.

Trénovací množina – náhodně vybraná část dat, která slouží pro učení sítě

Testovací množina – další část dat sloužící k zastavení trénování, aby nedošlo k přeučení sítě

Validační množina – zbytek dat, na kterém ověříme konečnou kvalitu modelu. Jde o data, která dosud model k dispozici neměl

Typicky se toto rozdělení dělá v poměru 50-25-25, případně 70-15-15. Ve výsledcích je pak reportován výkon na každé z těchto množin, přičemž většinou vybíráme model, který nemá příliš velké výkyvy mezi výkony na jednotlivých množinách.



Využití neuronových sítí

Neuronové sítě mají (někdy až pozoruhodnou) schopnost extrahovat pravidla a trendy z komplikovaných průběhů v datech. Další vlastností je, při správné aplikaci, schopnost velmi přesně předpovědět údaje, které nebyly součástí trénovacích dat, tedy schopnost zobecňovat. Tuto schopnost mají samozřejmě i jiné typy algoritmů, které spadají do Data Miningu. Neuronové sítě se nejčastěji používají např. k

- › Odhadu dynamické stability u energetických systémů
- › Rozpoznání poruch strojů a výpočetní techniky
- › Lékařství – detekce a odhad velikosti tumoru, chemická diagnostika
- › Dopravní signalizace
- › Rozpoznání kvality výrobku bez nutnosti laboratorních experimentů
- › Předpověď finančních časových řad včetně velmi dynamických řad (burza, směnné kurzy)
- › Samoobslužné mechanismy (subsystémy řízení budov)
- › Optické rozpoznávání textů, písma a podpisů
- › NN byla nasazena také v systémech pro převod mluvené řeči do písemné podoby
- › Reakce spotřebitelů na nové zboží
- › Detekce úvěrového rizika
- › Odhad kvality ropných produktů

Konkrétní využití neuronových sítí v praxi naleznete například pod následujícími odkazy:

[UNIPETROL RPA](#)



[Česká rafinérská](#)



Více o aplikaci neuronových sítí se můžete dozvědět například na našem odborném [kurzu](#). V nějakém z příštích čísel bychom Vás rádi provedli konkrétním příkladem v modulu *STATISTICA Automatické neuronové sítě*.