

**Technická univerzita v Liberci
Hospodářská fakulta**

Studijní program: Systémové inženýrství a informatika (6209T)

Studijní obor: Manažerská informatika (M 6209)

Aplikace metod umělé inteligence v ekonomice

Application of artificial intelligence methods in economics

Číslo práce: DP-MI-KIN-17

Miloslav Zyk

Vedoucí práce: Ing. Dana Nejedlá (KIN)

Konzultant: doc. Ing. Václav Záda, CSc. (KSI)

Počet stran: 90

Datum odevzdání: 6. 1. 2006

Počet příloh: 4

Prohlášení

Byl jsem seznámen s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, zejména § 60 - školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé diplomové práce pro vnitřní potřebu TUL.

Užiji-li diplomovou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Diplomovou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím diplomové práce a konzultantem.

Datum: 29.12.2005

Podpis:

Touto cestou bych chtěl poděkovat paní inženýrce Nejedlové za vedení mé diplomové práce.

Resumé

Práce se zaměřuje na použití metod umělé inteligence v ekonomii. První část - teoretická je věnována úvodu do problematiky. Důraz je zejména kladen na metody genetických algoritmů a neuronových sítí. Vedle základního popisu jsou uvedeny i možnosti aplikací těchto metod v ekonomické praxi a jejich srovnání s klasickým statistickým nástrojem, a to s regresní analýzou.

V druhé části – praktické jsou teoretické poznatky aplikovány na čtyřech modelových příkladech. První se zabývá problematikou odhadu koeficientů regresní funkce pomocí regresní analýzy a srovnání výsledku s hodnotami dosažených pomocí genetického algoritmu. Druhý příklad je zaměřen na analýzu dat pomocí vícevrstvé neuronové sítě. Ve třetím příkladě jsou prezentovány klasifikační schopnosti neuronových sítí. V závěrečném příkladě je demonstrována jednoduchá dvouvrstvá neuronová síť a její aplikační možnosti.

Abstract

The diploma paper focuses on use of the artificial intelligence methods in economy. The first part – theoretical – is devoted to introduction to the matter. Accent is given on the methods of genetic algorithms and neural networks. Basic description is presented together with possibilities of application of these principles in economic practice and their comparison with classic statistical instrument, regression analysis.

In the second part – practical – the theoretical findings are applied to four model examples. First of them deals with problematic of forecasting of regression function coefficients using regression analysis and comparison of the result with values obtained by genetic algorithm. The second example is focused on a multi-layer neural network data analysis. The third presents classification potentialities of the neural networks. The last example illustrates a simple dual neural network and its application possibilities.

Klíčová slova

umělá inteligence

neuronové sítě

genetické algoritmy

predikce

analýza

regrese

Obsah

Seznam zkratek.....	10
1. Úvod	11
2. Historie	13
3. Aplikace metod umělé inteligence v ekonomii	16
3.1 Specifičnost aplikací v ekonomii.....	16
3.2 Umělá inteligence ve vztahu k financím a investování	17
3.3 Umělá inteligence ve vztahu k nezaměstnanosti	18
4. Regresní analýza.....	20
4.1 Regresní modely	22
4.2 Volba regresní funkce.....	25
4.3 Odhady regresních parametrů - metoda nejmenších čtverců.....	27
5. Genetické algoritmy	31
5.1 Základní popis	31
5.2 Operátory genetických algoritmů	33
5.2.1 Kódování chromozomů	33
5.2.2 Křížení	34
5.2.3 Mutace	34
5.3 Parametry genetických algoritmů.....	35
5.4 Inicializace.....	36
5.5 Selektce.....	36
5.5.1 Elitismus	39
5.6 Doporučení	39
5.7 Aplikace genetických algoritmů, jejich výhody a nevýhody	40
6. Neuronové sítě.....	41
6.1 Základní pojmy	41
6.2 Způsob učení	46
6.3 Topologie neuronových sítí	51
6.4 Standardizace dat.....	53
6.5 Neuronové sítě a regresní modely	55
6.6 Oblasti využití neuronových sítí.....	57
7. Aplikace zjištěných poznatků v ekonomické praxi	59
7.1 Regresní analýza.....	60
7.2 Stanovení parametrů regresní funkce pomocí GA	64
7.3 Predikce ceny grafické karty pomocí neuronové sítě.....	66
7.4 Klasifikace podniků do kategorií „prosperita“ a „bankrot“	70
7.5 Využití jednoduché dvouvrstvé neuronové sítě.....	73
8. Závěr.....	75
9. Seznam literatury	77
9.1 Další zdroje.....	79
10. Seznam příloh.....	80

Seznam obrázků

obr. 4-1 Přímka	26
obr. 4-2 Hyperbola.....	26
obr. 4-3 Parabola	26
obr. 4-4 Logaritmická funkce	26
obr. 5-1 Základní genetický algoritmus.....	32
obr. 5-2 Metoda rulety	37
obr. 5-3 Situace před setříděním (graf podle kritérií).....	38
obr. 5-4 Situace po setřídění (graf podle pořadí).....	38
obr. 6-1 Biologický neuron.....	42
obr. 6-2 Umělý neuron	42
obr. 6-3 Umělá neuronová síť.....	43
obr. 6-4 Neuron n_j vyjmutý ze skryté vrstvy	43
obr. 6-5 Sigmoida	45
obr. 6-6 Skoková přenosová funkce	45
obr. 6-7 Lineární prahová přenosová funkce.....	46
obr. 6-8 Gaussova přenosová funkce.....	46
obr. 6-9 Schéma mutace neuronové sítě.....	50
obr. 7-1 Ukázka zdrojových dat	60
obr. 7-2 Korelační matice	61
obr. 7-3 Korelační matice bez "pipelines"	61
obr. 7-4 Regresní parametry	62
obr. 7-5 Hodnoty determinačních indexů	63
obr. 7-6 Výsledek dotazu.....	69
obr. 7-7 Stupeň natrénování.....	72
obr. 7-8 Výsledek dotazu.....	72
obr. 7-9 Výsledek "časové kalkulace"	74

Seznam tabulek

tab. 5-1 Popis chromozomu	33
tab. 5-2 Příklad křížení chromozomů	34
tab. 5-3 Příklad mutace chromozomů.....	34
tab. 6-1 Schéma tréninkové matice	47

Seznam grafů

graf 7-1 Regresní křivky	62
graf 7-2 Regrese	64
graf 7-3 Regrese pomocí GA.....	65
graf 7-4 Suma čtverců odchylek - trénink	67
graf 7-5 Situace po ukončení trénovací fáze	68
graf 7-6 Suma čtverců odchylek – test	69
graf 7-7 Suma čtverců odchylek - trénování NS	71

Seznam zkratek

AI – Artificial Intelligence (umělá inteligence)

APT – Arbitrage pricing model (arbitrážní model)

DNA – Biochemická sekvence ve tvaru dvojité šroubovice, nesoucí genetický kód

GA – Genetický algoritmus

GPU – Graphics Processing Unit (grafický procesor)

GTO – Genetic Training Option (metoda trénování NS pomocí GA)

MNČ – Metoda nejmenších čtverců

NS – Neuronová síť

UI – Umělá inteligence

1. Úvod

Charakteristickým rysem rozvoje umělé inteligence jako vědní disciplíny bylo v posledním půlstoletí její těsné spojení s vývojem výpočetní techniky. Ta umožnila rychle řešit nejen numericky náročné úlohy, ale vedla k hledání kvalitativně nových a efektivnějších metodických postupů. Novodobá věda ukazuje, že složitost jevů reálného světa s mnoha vazbami mezi jeho objekty se neustále zvětšuje a vzniká tedy snaha se s ní vyrovnat adekvátními metodami.

Vybraná definice umělé inteligence :

“Umělá inteligence se zabývá systémy, které na stimulaci reagují konzistentně s tradičními reakcemi lidí, s lidskou schopností pozorování, odhadu a záměru. Každý takový systém by se měl sám zabývat kritickým hodnocením a výběrem různých variant úsudků. Systémy vyprodukované lidskou dovedností a prací by se měli samy řídit v souladu si životem, duchem a vnímavostí, ačkoli jsou ve skutečnosti napodobeninami.”

Latanya Sweeney

Složitost reálného světa není ovšem jedinou motivací rozvoje metod umělé inteligence. Neopomenutelná je jeho neurčitost, která může mít různý charakter i původ. Dnes jsou v literatuře rozpracovány mnohé teorie a míry neurčitosti dat i znalostí, některé však teprve čekají na využití v praxi. Fenomén neurčitosti tak výrazně ovlivňuje vývoj vědeckého výzkumu, který usiluje o řešení problémů měnícího se reálného světa, kde dosavadní zdánlivě přesná řešení nemusí vždy odpovídat skutečnosti. Někteří autoři proto přisuzují neurčitosti podstatný význam a umělou inteligenci pojímají jako schopnost systému dosáhnout cíle nebo udržet požadované chování právě v neurčitých podmínkách.

Je tedy přirozené, že se stále vyvíjejí nové metodické postupy a technické systémy kvalitativně vyšší úrovně, na které je kladen požadavek „*napodobit*“ intelektuální schopnosti člověka hlavně při rozpoznávání a analýze různých jevů, logickém usuzování a rozhodování nebo při operativním řízení, a to i v neurčitých podmínkách.

Se složitostí a neurčitostí je úzce spjata důvěryhodnost matematických modelů. Třebaže je často možné popsat různé stránky reality různými modely, odhad jejich parametrů může být zatížen nepřesnostmi. Modely je kromě toho nezbytné v mnoha

případech zjednodušovat, aby byly numericky zvládnutelné a to v přijatelném čase. Tento požadavek je zvláště důležitý při operativním řízení v reálném čase. Zjednodušením modelů se však zvětšuje jejich neurčitost a zmenšuje jejich důvěryhodnost. Nalézt optimální strukturu modelu proto může být náročným úkolem, pro jehož řešení nám zatím chybějí srovnávací studie vzájemných vztahů mezi uvedenými charakteristikami a zhodnocení jejich dopadů na modelové řešení problému.

Jiným problémem při řešení rozlehlých systémů s mnoha prvky a vazbami jsou velké objemy dat, jejichž zpracování vyžaduje efektivnější metodické postupy. Rozvíjet nové přístupy je žádoucí též při hledání optimální oblasti složitých systémů s více kritérii nebo při současném uplatnění kvalitativních a kvantitativních kritérií.

Z výše uvedených faktů je tedy zřejmé, že oblast ekonomiky doslova vybízí k nasazení metod umělé inteligence, které sice v mnohých případech nemohou nahradit klasické metody, ale na druhou stranu mohou podávat skvělé výsledky v situacích, kdy použití klasických postupů je nevyhovující, popřípadě zcela nemožné.

Cílem mé práce je tedy poskytnout srovnání klasických přístupů při analýze dat, jmenovitě se jedná o regresní analýzu, s možnostmi použití metod umělé inteligence, zejména genetických algoritmů a neuronových sítí. Toto srovnání provedu na několika praktických příkladech, které jsou popsány v druhé části této práce. Ke všem uvedeným příkladům jsou na přiloženém CD k nahlédnutí zdrojové kódy ukázkových programů.

2. Historie

Zárodky umělé inteligence (dále UI) se překvapivě objevují už v dávné prehistorii, kdy lidé byli neodolatelně přitahováni myšlenkou vytvořit jakousi inteligenci mimo lidské tělo. Náznaky UI lze nalézt v řecké mytologii (např. pokusy o vytvoření umělého větru nebo umělého člověka). Před více než třemi sty lety vyslovil René Descartes (1596-1650) patrně jako první názor, že umělá inteligence je nemožná.

V roce 1943 se objevuje první práce Warrena McCullocha a Waltera Pittse z oboru neuronových sítí. Tato práce vytvořila základ pro pozdější vývoj matematických modelů složitějších sítí, který vlastně pokračuje dodnes. První kroky UI byly umožněny hlavně nástupem počítačů a jsou spojeny též s počátky rozvoje kybernetiky. Zakladatel teorie informace Claude Shannon v roce 1950 předpokládal, že počítače budou v blízké době schopny hrát šachy. Zakladatel kybernetiky Norbert Wiener vycházel ve svých pracích z podobnosti myšlení a práce stroje. Jistě jste všichni slyšeli o testu pro prověření inteligence počítače, který navrhl britský matematik Alan Turing rovněž v roce 1950, který lze popsat jako kritérium pro rozhodnutí, zda je stroj (počítač) inteligentní. Člověk komunikuje s neviditelným oponentem tak dlouho, dokud nenabude dojmu, že komunikuje s lidskou bytostí, ale ve skutečnosti komunikuje se strojem (počítačem). Jedná se o jednosměrný test inteligence a to z důvodu, že rozhodnutí o tom, co je inteligentní, je ponecháno na člověku. Ve směru od stroje tento vztah neplatí.

Za počátek UI jako vědní disciplíny se zpravidla považuje konference o výpočetní technice a jejích perspektivách, konaná v roce 1956 na Dartmouth College, New Hampshire, USA. Zde se sešli přední odborníci z různých vědních oborů (např. Marvin Minsky – zakladatel laboratoře UI na Massachusetts University, Claude Shannon, Nathaniel Rochester z IBM, Allen Newell – první prezident americké asociace UI, Herbert Simon – nositel Nobelovy ceny, aj.) a odvážně předpověděli rozvoj výpočetní techniky, která umožní v krátké době simulovat učení nebo jiný příznak inteligence. Byla zdůrazněna myšlenka, že počítače by mohli pracovat se symboly stejně dobře jako s čísly. Podle návrhu Johna McCarthyho, jednoho z organizátorů konference, byl pro nově vznikající disciplínu přijat název umělá inteligence [10].

První období vývoje až do roku 1965 se považuje za „*období úsvitu*“, ve kterém se řešily zejména problémy geometrických analogií a rozvíjely se metody zpracování nealgoritmických (symbolických) vstupů, které byly později využity při vývoji expertních systémů.

Další období (1965-1970) se hodnotí jako „*období temna*“, které přineslo málo úspěchů. Nesplnil se optimistický předpoklad předchozího období, že vyvinutí inteligentního počítače bude snadným úkolem. Začínají se však vyvíjet expertní systémy. Jmenujme především expertní systém DENDRAL k identifikaci chemických sloučenin na základě spektrografických dat.

V období „*renesance*“ (1970-1975) pokračoval vývoj expertních systémů. Nejznámějším se stal systém MYCIN pro lékařské diagnostické účely, vyvinutý ve Stanford Research Institute.

V období „*společenství*“ (1975-1980) nastal rozmach interdisciplinární spolupráce s odborníky z jiných oblastí, zejména s lingvisty a psychology, která později začala přinášet výsledky při rozvíjení různých metodických postupů UI.

Současné období se pokládá hlavně za období komercializace („*age of entrepreneurs*“), která usiluje o aplikace na problémy reálného světa v různých inženýrských oborech.

Naléhavá potřeba aplikace UI se ukázala především při rozvíjení řídicích systémů. Současně se zřetelně projevuje úsilí o podstatné zvýšení paměti a rychlosti počítačů. Techniky numerického zpracování dat a softwarové vybavení se tak staly v éře informační exploze dominantní technologií, potřebou při řešení mnoha složitých problémů.

Metodické postupy UI se zpočátku rozvíjely v samostatných a dosti heterogenních oblastech:

- Teorie rozpoznávání
- Adaptace a učení
- Řešení úloh¹
- Expertní systémy a znalostní inženýrství
- Neuronové sítě
- Genetické algoritmy
- Robotika
- Komunikace se stojem v přirozeném jazyce

Charakterickým rysem současného vývoje UI je prorůstání jejích různých partií, podobně jako lze zaznamenat prolínání systémových disciplín [10]. Tak vznikly např. fuzzy-neuronové sítě, neuro-genetické systémy, problematika expertních systémů zčásti splývá s kvalitativním modelováním apod.

Druhou významnou charakteristikou současného vývoje UI je úsilí o zvládnutí problematiky rozhodování i v podmínkách neurčitosti. UI je pojímána jako schopnost systému dosáhnout cíle nebo udržet požadované chování i v podmínkách neurčitosti [10].

Dosavadní výsledky ukazují, že poznání a výzkum nejsou v oblasti UI zdaleka uzavřené (např. vznik disciplíny soft computing, které hledá netradiční přístupy k řešení problémů neurčitosti účelným propojením různých výpočtových metodologií – fuzzy logiky, neuropočítání, genetického počítání, pravděpodobnostního usuzování).

¹ Řešení úloh je v UI hledání přípustné cesty mezi počátečním a koncovým stavem prostředí.

3. Aplikace metod umělé inteligence v ekonomii

Cílem této kapitoly je nastínit oblasti možného použití UI metod v ekonomické praxi. Nejprve se zmíním o specifikách aplikací v ekonomii, poté podrobněji rozeberu UI ve vztahu k financím a investování a též se zmíním o možných dopadech UI na nezaměstnanost.

3.1 Specifičnost aplikací v ekonomii

Metody umělé inteligence zaznamenaly v ekonomii řadu úspěšných aplikací. Ať už jde o aplikace ve finanční oblasti nebo o marketing či makroekonomické modely, na trhu se za posledních pár let objevila celá řada komerčně využívaných systémů. Jmenujme pár nejdůležitějších oblastí, kde se umělá inteligence již využívá: určení bonity klienta, podpora auditu, investiční doporučení, predikce bankrotu, analýza cenných papírů, predikce vývoje směnných kurzů, odhad míry inflace, atd.

O vzrůstu zájmu o tuto aplikační oblast svědčí mimo jiné i vznik nových časopisů věnovaných této problematice (International Journal of Intelligent Systems in Accounting, AI in Finance, atd.).

Ekonomické aplikace mají ve srovnání s jinými oblastmi některé specifické rysy. Vzhledem k tomu, že jde o peníze (aplikace nových metod přináší uživatelům zisk i náskok před konkurencí), jen velmi málo se publikuje o úspěšných systémech. Podobně i nezdarů nebývají příliš zveřejňovány, aby nesloužily konkurenci jako informace „*tudy cesta nevede*“.

Použití metod umělé inteligence napomáhá organizačním změnám a decentralizaci u uživatelů. Rozhodovací pravomoci se přesouvají od expertů v ústředí k „*ne-expertům*“ na nižších úrovních firmy (např. z centrály banky na její pobočky). Použití metod umělé inteligence přitom zajišťuje, že rozhodnutí budou provedena standardním reprodukovatelným způsobem a že budou dobře zdokumentována.

V ekonomické oblasti se již roky pracuje s výpočetní technikou. Používají se metody operačního výzkumu a statistické metody, budují se manažerské informační

systémy. Proto se při nasazení metod umělé inteligence musí uvažovat o jejich integraci do stávajícího prostředí. Může se snadno stát, že komponenta s umělou inteligencí bude tvořit jen malou část celého systému [19].

3.2 Umělá inteligence ve vztahu k financím a investování

Počítačové zpracování dat, prezentujících finanční aktiva, má analogický charakter jako transformace jakýchkoli dat ze vstupu na výstup. Expertní (znalostní) systémy ve finančnictví a investování obsahují datové báze a *pravidla* umožňující plnit záměry uživatele. Databáze finančních dat se mohou nalézat v různých typech tabulkových procesorů (Excel, Lotus, Qpro atd.) nebo ve specializovaných relačních, resp. objektově orientovaných databázích atd. Množství a relativní přesnost, obvyklé u finančních dat, jsou výhodami pro uživatele, přestože například neuronové sítě nejsou zpravidla tak citlivé na nepřesná data a neúplné množiny dat, jako tomu bývá u běžných statistických softwarů.

Pravidly mohou například být různé softwary umělé inteligence. Ve finančnictví a investování jsou to zejména neuronové sítě, evoluční (genetické) algoritmy a systémy s fuzzy logikou. Existuje však celá řada dalších technologií (postupů) k získání požadovaných výstupů. Užívají se i kombinované technologie umělé inteligence, jako je například technologie neurofuzzy [9]. V mé práci se zmíním především o dvou technologiích umělé inteligence:

- neuronových sítích
- genetických algoritmech

Neuronové sítě (dále NS) nabízejí finančníkům a investorům řešení klasifikačních i predikčních úloh. Na kapitálových trzích jde o práci s časovými řadami cen, objemy obchodních transakcí, ale i o práci s fundamentálními proměnnými o stavu firmy, úvěrů a vlastních finančních aktiv. Technologie neuronových sítí umožňuje řešit klasické úlohy statistiky, jako je regresní analýza, tj. například arbitrážní model (arbitrage pricing model - APT), který sleduje faktory ovlivňující pohyb ocenění finančního aktiva – akcie apod. Predikce ceny neuronovou sítí může doplnit, resp. nahradit statistické odhady, prováděné například technikou klouzavých průměrů, metodou nejmenších čtverců apod.

Genetické algoritmy (dále GA) jsou podmnožinou evolučních algoritmů. Inspirována biologií, podobně jako je tomu i u neuronových sítí, je technologie genetických algoritmů určena pro optimalizační úlohy. Hlavní doménou pro nasazení této technologie mohou být úlohy optimalizace portfolia finančních aktiv.

Vedle těchto dvou technologií se ještě musím alespoň okrajově zmínit o systémech s fuzzy logikou, které lze užít všude tam, kde je postup vedoucí k rozhodnutí nejasný. Mlhavost představ o postupech vedoucích k rozhodnutí odpovídá lidskému způsobu řešení úloh a jejich algoritmizaci. Ani u superrychlých počítačů není v řadě případů možné prohledávání stavového prostoru možných řešení (s ohledem na spotřebu času). Navíc je pochybné, zda by bylo možné takovýto způsob řešení úloh nazvat inteligentním. Člověk obvykle nezkoumá u složitých úloh všechna možná řešení. Objem nákupu / prodeje finančního aktiva při kolísání cen aktiva je jedno z možných témat pro řešení systémem fuzzy.

3.3 Umělá inteligence ve vztahu k nezaměstnanosti

Jedna ze sociálních oblastí, na kterou bude mít v budoucnu umělá inteligence svůj dopad, bude nezaměstnanost. Umělá inteligence a další vývoj v oblasti počítačů změnily postavení strojů z role bezduchých monotónně pracujících automatů do role, kdy mohou vykonávat velmi komplikovanou činnost, která byla až donedávna vyhrazena jen lidem. Z toho samozřejmě plyne otázka, jestli způsobí masový nástup umělé inteligence nezaměstnanost. Existují dvě hlediska na tuto problematiku. Jedno tvrdí, že umělá inteligence není nic jiného, než další technologie, která zvýší produktivitu a efektivnost práce a zároveň přinese i pracovní příležitosti vyžadující samozřejmě vyšší vzdělání

Na druhou stranu nelze zanedbávat fakt, že zároveň ubere práci v oborech, které nevyžadují tvůrčí činnost, čehož důkazem jsou již dnes plně robotizované provozy v některých továrnách. Obě hlediska jsou do jisté míry pravdivá a zároveň i mylná. Obyčejní lidé, kteří se s tímto dilematem setkávají, pohlíží na tento problém obvykle velmi zjednodušeně. Faktem je, že použití umělé inteligence přinese v jistých oborech vlnu

nezaměstnanosti a v jiných zvýšenou poptávku po pracovních silách. V podstatě se bude jednat o přechodné období, jehož průběh a dopady budou záviset také z velké části na postoji politiků. Jestliže podcení nástup aplikací s umělou inteligencí, může to být problém.

Dnešní stroje s umělou inteligencí mohou nahradit člověka v mnohem širším záběru. Mohou být v blízké budoucnosti plně univerzálním pokladníkem v bance, spolehlivým pilotem nebo jen uklízečem či horníkem. Jejich nástup tedy může způsobit větší sociální otřesy, pokud by se s nimi v plánech společnosti nepočítalo. Nemá cenu se využití umělé inteligence stavět na odpor. Stejně, jako se vyvíjí jedinec během života, stejně tak se vyvíjí i společnost a její znalosti a dovednosti. Z tohoto hlediska lze na vývoj techniky nahlížet jako na přirozený evoluční proces, jehož brždění by nebylo to nejlepší řešení.

Kromě dočasné nezaměstnanosti, která může být při šikovných opatřeních neznatelná, přinese využití umělé inteligence zcela jistě také vzrůst ekonomiky státu. Při jejím využívání by měla vzrůst produktivita práce mnohonásobně, což by se mělo v konečném důsledku projevit i na životní úrovni obyvatelstva.

Činnost člověka je od prvního zavedení strojů neustále vytlačována z oblastí manuální dřiny směrem k tvořivé činnosti. Při masovém použití umělé inteligence lze očekávat jisté společenské změny, nicméně jaké budou a jaký bude jejich průběh, záleží pouze na těch, kteří budou mít na starosti jejich masové zavedení.

4. Regresní analýza

Regresní analýza se používá při zkoumání závislosti dvou a více číselných proměnných. Je to souhrn statistických metod a postupů sloužících k odhadu **hodnot** nebo **středních hodnot** nějaké proměnné odpovídající daným hodnotám jedné či většího počtu vysvětlujících proměnných. Údaje o těchto proměnných, zjištěné u n jednotek, se považují za výběrová data.

Problémy, k jejichž řešení lze regresní analýzy využít, vznikají v praxi poměrně často. V ekonomické oblasti se regresní analýza rozšířila snad nejvíce při analýze a prognózování spotřeby a poptávky, kdy se konstruovaly různé regresní modely sloužící odhadu střední (průměrné) spotřeby či poptávky domácností s různým příjmem, s různým počtem členů, s různým počtem dětí apod.

Je-li regresní analýza zaměřena na odhady hodnot či středních hodnot jedné proměnné odpovídajících daným hodnotám jiných proměnných, je zřejmé, že věnuje pozornost především průběhu závislostí jedné proměnné na jiné či jiných proměnných. Proměnnou, jejíž hodnoty či střední hodnoty mají být odhadovány, tj. vysvětlovanou nebo závisle proměnnou, budeme značit symbolem y . Proměnnou, pomocí níž se tyto odhady provádějí, tj. vysvětlující čili nezávisle proměnnou, budeme značit symbolem x . Bude-li vysvětlujících proměnných více, označíme je třeba x_1, x_2 nebo x_1, x_2, x_3 apod.

Ekonomické veličiny závisí zpravidla na větším počtu činitelů. Z nich lze při regresní analýze využít pouze těch, které lze měřit. Ty pak tvoří okruh vysvětlujících proměnných, použitelných k odhadům hodnot či středních hodnot vysvětlované proměnné. Zpravidla se však k těmto odhadům používají pouze některé z možných vysvětlujících proměnných. Pokud se využívá pouze jedna z nich, hovoří se o **jednoduché regresi**. Zapojí-li se do odhadů větší počet vysvětlujících proměnných, hovoří se o **vícenásobné** (tj. dvojnásobné, trojnásobné atd.) **regresi**. Od jednoduché regrese se přechází k vícenásobné proto, aby se odhady hodnot či středních hodnot vysvětlované proměnné zlepšily. Zkušenosti z aplikací regresní analýzy v ekonomické oblasti ukazují, že nebývá vhodné volit vysvětlujících proměnných příliš mnoho. Vzniká tím totiž nebezpečí, že se mezi vysvětlující proměnné zahrnou vedle podstatných činitelů i činitele nepodstatné, analýza se zbytečně komplikuje a výsledky se obtížně interpretují.

Podkladem pro regresní analýzu jsou vždy nějaká data získaná pozorováním (zjišťováním). Někdy pracujeme s hodnotami číselných proměnných zjištěných v určitém období či okamžiku u n nějakých jednotek, jimiž mohou být osoby, domácnosti, prodeje apod. V tomto případě se regresní analýza týká **prostorových řad**. Jsou-li k dispozici hodnoty číselných proměnných získaných v n po sobě jdoucích obdobích, týká se regresní analýza **časových řad**. V obou případech je podkladem pro jednoduchou regresní analýzu n dvojic, pro vícenásobnou pak n trojic, čtveřic atd. hodnot. Některé metody regresní analýzy lze stejně dobře aplikovat na prostorové i na časové řady. Obecně lze však říci, že regresní analýza časových řad má určité odlišnosti ve srovnání s analýzou prostorových řad.

O údajích, jež jsou podkladem pro regresní analýzu, předpokládáme, že byly získány náhodným výběrem. Přitom rozlišujeme:

- a) náhodný výběr jednotek s **předem volenými** hodnotami vysvětlujících proměnných, kdy se u vybraných jednotek s těmito hodnotami zjišťují pouze hodnoty vysvětlované proměnné.
- b) náhodný výběr, při němž se u n vybraných jednotek zjišťují jak hodnoty vysvětlované proměnné, tak i hodnoty vysvětlujících proměnných.

V prvním případě považujeme hodnoty vysvětlujících proměnných za známé konstanty – známe je totiž již před provedením výběru. Ve druhém případě jsou tyto hodnoty výsledkem náhodného výběru, takže je považujeme za hodnoty náhodných veličin. Hodnoty vysvětlované proměnné jsou pak hodnotami náhodných veličin v obou případech.

Pro výklad základů regresní analýzy je předpoklad, že se hodnoty vysvětlujících proměnných volí, velmi výhodný. Vede totiž k značnému zjednodušení matematicko - statistické teorie.

4.1 Regresní modely

Matematický model, v němž je vyjádřena představa o stochastické závislosti a výchozí předpoklady regresní analýzy, se nazývá regresní model. Všimněme si charakteristických rysů takového modelu a předpokládejme přitom, že sledujeme závislost proměnné y na jediné vysvětlující proměnné x . V tomto případě je výsledkem pozorování (zjišťování) n dvojic hodnot $x_i, y_i, i=1, 2, \dots, n$, které si můžeme představit jako n bodů v rovině (bodové diagramy).

Připusťme na okamžik, že se podařilo izolovat závislost proměnných x_i, y_i od spolupůsobení vedlejších vlivů. Potom lze předpokládat, že všechny body (x_i, y_i) budou ležet na čáře s rovnicí $y=\eta(x)$ a že tedy pro všechny pozorované dvojice hodnot lze psát

$$y_i = \eta(x_i), i = 1, 2, \dots, n. \quad (4.1)$$

Model popsáný těmito rovnicemi odpovídá představě, že změny proměnné y jsou jednoznačně určeny (determinovány) změnami proměnné x . Nazývá se **deterministický model**. Protože na proměnnou y působí kromě proměnné x i další vlivy, neleží body na čáře, ale kolísají kolem ní. Tuto skutečnost se snažíme zachytit i v matematickém modelu. Předpokládáme, že každou hodnotu vysvětlované proměnné lze rozložit na dvě složky, z nichž jedna $\eta_i \equiv \eta(x_i)$ je funkcí hodnoty x_i a druhá ε_i je výslednicí dalších (vedlejších a náhodných) vlivů. První složka je **deterministickou** a druhá **náhodnou složkou** (též rušivou složkou či náhodnou poruchou) modelu.

Nejčastěji se předpokládá, že se obě složky skládají sčítáním, tj. že

$$y_i = \eta_i + \varepsilon_i = \eta(x_i) + \varepsilon_i, i = 1, 2, \dots, n. \quad (4.2)$$

V tomto případě hovoříme o modelu **součtového** typu neboli o **aditivním** modelu. Někdy se předpokládá, že se obě složky skládají násobením, což vede k modelu **součinového** typu neboli **multiplikativnímu** modelu.

Funkce $\eta=\eta(x)$, jejíž hodnoty figurují v rovnici (4.2), se nazývá regresní funkce. Regresní funkci může být přímka $\eta=\beta_0+\beta_1x$, parabola $\eta=\beta_0+\beta_1x+\beta_2x^2$ a řada jiných funkcí. Každá regresní funkce má určitý počet parametrů:

- *přímka* dva (β_0, β_1)
- *parabola* tři ($\beta_0, \beta_1, \beta_2$)
- apod.

Tyto parametry se v regresní analýze považují za neznámé konstanty, jejíž hodnoty se dají odhadnout z pozorovaných (výběrových) dat. Počet parametrů budeme značit symbolem p .

Ve vícenásobné regresní analýze se pracuje s větším počtem vysvětlujících proměnných. Regresní funkce je tedy funkcí většího počtu proměnných a nazývá se vícenásobná regresní funkce [17].

Při bližší konkretizaci regresního modelu je třeba:

- a) specifikovat regresní funkci,
- b) zavést určité předpoklady o pravděpodobnostním rozdělení rušivých složek.

Za regresní funkce se volí různé funkce známé z matematiky. Poměrně často se používají funkce, které jsou *lineární z hlediska proměnných*. K nim patří již zmíněná přímka, dále rovina

$$\eta = \beta_0 + \beta_1 x_1 + \beta_2 x_2 \quad (4.3)$$

(x_1, x_2 jsou vysvětlující proměnné) a regresní nadrovina

$$\eta = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_m x_m \quad (4.4)$$

(s vysvětlujícími proměnnými $x_1, x_2, \dots, x_m$). Povšimněme si, že všechny tři uvedené funkce jsou lineární rovněž z hlediska parametrů. Kromě nich se užívá i řada dalších regresních, které nejsou lineární z hlediska proměnných a jsou lineární jen z hlediska parametrů.

K nim patří parabola

$$\eta = \beta_0 + \beta_1 x + \beta_2 x^2, \quad (4.5)$$

dále hyperbola

$$\eta = \beta_0 + \frac{\beta_1}{x}, \quad (4.6)$$

logaritmická funkce

$$\eta = \beta_0 + \beta_1 \log x \quad (4.7)$$

a řada dalších funkcí [16].

Regresní funkce, které jsou lineární z hlediska parametrů, se nazývají **lineární regresní funkce** a regresní modely s takovými funkcemi se nazývají **lineární regresní modely**.

V praxi se však používají takové regresní funkce, které nejsou lineární z hlediska parametrů. Jsou to například exponenciální funkce

$$\eta = \beta_0 \cdot \beta_1^x, \quad (4.8)$$

mocninná funkce

$$\eta = \beta_0 x^{\beta_1} \quad (4.9)$$

a mnohé jiné. Nazývají se **nelineární regresní funkce** a regresní modely s takovými funkcemi se nazývají **nelineární regresní modely**.

4.2 Volba regresní funkce

Rozhodneme-li se použít klasický model, je základní otázkou jeho specifikace volba regresní funkce. Tím rozumíme:

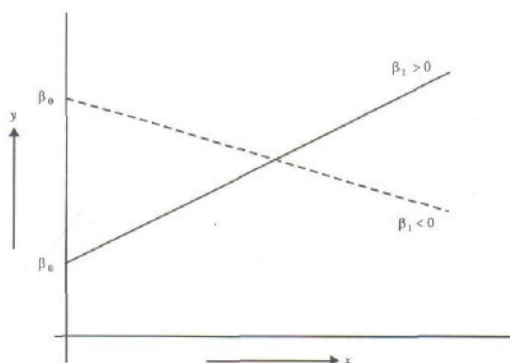
- a) rozhodnutí, které vysvětlující proměnné budou pojaty do regresní funkce,
- b) volby **typu** regresní funkce, tj. například při jedné vysvětlující proměnné rozhodnutí, zda bude regresní funkcí přímka, parabola, hyperbola či nějaká jiná lineární regresní funkce dané vysvětlující proměnné.

Pokud je to možné, měla by volba regresní funkce vyplynout z věcného rozboru vztahů ve zkoumané oblasti, tj. měla by se opírat o určitou teorii. Ta by měla pomoci především při rozhodování o tom, které vysvětlující proměnné pojmout do regresní funkce a měla by poskytnout i určitá vodítka pro volbu typu regresní funkce. Teoretická analýza může upozornit například na to, že s růstem hodnot x_i budou mít hodnoty y_i tendenci monotónně růst či klesat, že tato tendence má charakter zrychlujícího se či zpomalujícího se růstu či poklesu, že jde o závislost, kdy s růstem hodnoty y_i , který je po dosažení určitého maxima vystřídán poklesem apod. Takovéto charakteristické rysy průběhu závislosti jsou schopny odrazit různé funkce známé z matematiky, z nichž některé jsou znázorněny na obrázcích (4-1) až (4-4).

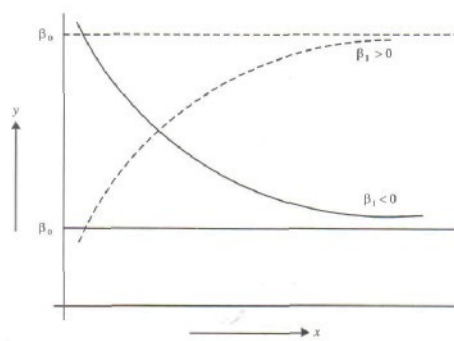
Někdy lze při volbě regresní funkce využít i zkušenost. O volbě regresní funkce vycházející z teorie či zkušenosti se někdy hovoří jako o **apriorní volbě** regresní funkce. Regresní funkce je totiž zvolena dříve, než začneme zkoumat, jak se závislost projevuje ve zjištěných údajích. Nelze-li zvolit regresní funkce apriorně, nezbyvá, než ji zvolit teprve po posouzení zjištěných údajů. Obvykle se vymezí určitá skupina možných regresních funkcí (zpravidla jednoduchých, jejíž parametry se dobře interpretují) a pak se zkoumá, které z těchto funkcí dobře přiléhají zjištěným údajům. Přílnavost regresní funkce k datům lze měřit různými charakteristikami, z nichž některé nejužívanější (reziduální součet čtverců, reziduální rozptyl, determinační index) popíši dále. Jde-li pouze o závislost proměnné y na jedné vysvětlující proměnné x , může být vodítkem pro volbu regresní funkce i bodový diagram či čára podmíněných průměrů. Vyplývá-li volba regresní funkce pouze z posouzení závislosti v datech, hovoří se o **empirickém přístupu** k volbě regresní funkce. Není-li počet pozorování velký, nemusí tento postup vést k nalezení regresní funkce

vhodné pro popis závislosti v základním souboru.

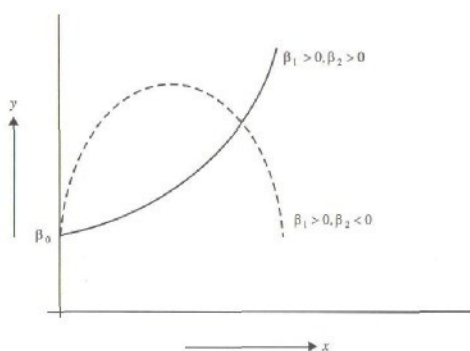
Důležitou otázkou, kterou je třeba při volbě regresní funkce posuzovat, je korelovanost regresorů figurujících v regresní funkci. Mějme vždy na mysli, že silně korelované regresory by v regresní funkci neměly být.



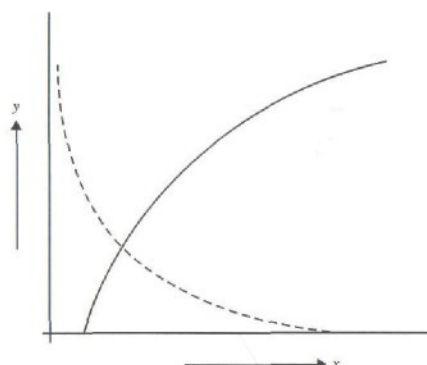
obr. 4-1 Přímka



obr. 4-2 Hyperbola



obr. 4-3 Parabola



obr. 4-4 Logaritmická funkce

4.3 Odhady regresních parametrů - metoda nejmenších čtverců

Lineární regresní funkce (4.4) má $p = m + 1$ parametrů $\beta_0, \beta_1, \beta_2, \dots, \beta_m$. Jsou to neznámé konstanty, jež se odhadují z pozorovaných (výběrových) dat. V tomto odstavci si všimneme bodových odhadů parametrů regresní funkce. Tyto odhady označíme $b_0, b_1, b_2, \dots, b_m$ a regresní funkci, v níž nahradíme parametry jejich odhady, označíme symbolem Y a nazveme **výběrovou regresní funkcí**. Vyjádříme ji ve tvaru

$$Y_i = b_0 x_{0i} + b_1 x_{1i} + b_2 x_{2i} + \dots + b_m x_{mi} \quad (4.10)$$

Její hodnoty jsou odhady deterministických složek $\eta_i, i = 1, 2, \dots, n$.

Za poměrně širokých podmínek jsou vhodnými bodovými odhady regresních parametrů odhady $b_0, b_1, b_2, \dots, b_m$ pořízené **metodou nejmenších čtverců** (MNČ). Jsou to odhady, minimalizující součet čtvercových odchylek

$$S = \sum_{i=1}^n (y_i - Y_i)^2 = \sum_{i=1}^n (y_i - b_0 x_{0i} - b_1 x_{1i} - b_2 x_{2i} - \dots - b_m x_{mi})^2. \quad (4.11)$$

Tento součet je minimální pro taková $b_0, b_1, b_2, \dots, b_m$, které se určí řešením soustavy rovnic

$$\frac{\partial S}{\partial b_0} = 0, \quad \frac{\partial S}{\partial b_1} = 0, \quad \frac{\partial S}{\partial b_2} = 0, \dots, \quad \frac{\partial S}{\partial b_m} = 0, \quad (4.12)$$

kde $\frac{\partial S}{\partial b_0} = 0$ je parciální derivace S podle b_0 a podobně ostatní.

Minimum součtu čtverců S pro daný typ regresní funkce se nazývá **reziduální součet čtverců**. Značíme jej S_R . Pokud předpokládáme, že Y_i jsou hodnoty výběrové regresní funkce, kde odhady b_0, b_1 atd. jež byly pořízeny metodou nejmenších čtverců, můžeme reziduální součet čtverců vyjádřit formálně stejně jako součet S , tj. ve tvaru

$$S_R = \sum (y_i - Y_i)^2. \quad (4.13)$$

Odchyly $e_i = y_i - Y_i$ jsou tzv. **rezidua**. Jejich velikost a tím i velikost reziduálního součtu čtverců závisí na tom, jak dobře přiléhá výběrová regresní funkce pozorovaným údajům. Přiléhá-li jim dobře, jsou rezidua a tím i reziduální součet čtverců malé a naopak.

Přiléhavost výběrové regresní funkce nám může napomoci při výběru typu regresní funkce. Máme-li vybrat mezi několika regresními funkcemi, dáváme přednost funkci s nejnižším reziduálním součtem čtverců. Ovšem nesmíme toto kritérium přeceňovat.

Dalším kritériem pro výběr regresní funkce je **reziduální rozptyl**

$$S_R^2 = \frac{S_R}{n - p}, \quad (4.14)$$

který je definovaný jako poměr reziduálního součtu čtverců k počtu pozorování n zmenšenému o počet parametrů regresní funkce p .

Často používanou charakteristikou výstižnosti regresní funkce je **determinační index**. Vychází z toho, že součet čtvercových odchylek $S_y = \sum (y_i - \bar{y})^2$ je součtem tzv. **teoretického součtu čtverců** S_T

$$S_T = \sum (Y_i - \bar{y})^2 \quad (4.15)$$

a **reziduálního součtu čtverců** S_R . Platí tedy

$$S_y = S_T + S_R. \quad (4.16)$$

Teoretický součet čtverců S_T se považuje za tu část součtu čtverců S_y , která je vysvětlena zvolenou výběrovou regresní funkcí, zatímco reziduální součet čtverců S_R se považuje za tu část, která zvolenou výběrovou regresní funkcí vysvětlena není. **Determinační index** je pak dán následujícím předpisem

$$I^2 = \frac{S_T}{S_y} = 1 - \frac{S_R}{S_y} \quad (4.17)$$

a nabývá hodnot z interval $<0,1>$. Výběrovou regresní funkce lze považovat za tím výstižnější, čím je determinační index bližší jedné.

Determinační index se používá i jako míra těsnosti závislosti proměnné y na jedné či větším počtu vysvětlujících proměnných. Je-li hodnota blízká jedné, hovoříme o silné závislosti proměnné y na proměnné x či na větším počtu vysvětlujících proměnných a opačně.

Je-li pak výběr dostatečně velký, můžeme usuzovat na silnou (těsnou) nebo slabou závislost i v souboru, z nějž výběr pochází.

V praxi se k posuzování výstižnosti regresní funkce i k posuzování těsnosti závislosti y na jedné či větším počtu vysvětlujících proměnných používá i tzv. **korelační index I** , který je odmocninou determinačního indexu.

Stejně jako reziduální součet čtverců, je i determinační index mírou, která nebere v úvahu počet parametrů regresní funkce. U regresních funkcí s více parametry vychází tedy obvykle vyšší než u regresních funkcí s méně parametry. Z toho důvodu byla pro rozhodování mezi funkcemi s různým počtem parametrů navržena jistá „oprava“ determinačního indexu, kterou značíme I_{opr}^2 . Tato oprava spočívá v tom, že se ve výrazu na pravé straně (4.17) neodečítá od jedné poměr S_R / S_y , ale poměr $(n - 1)S_R / (n - p)S_y$. Opravený determinační index se tedy počítá podle vzorce

$$I_{opr}^2 = 1 - \frac{(n-1)S_R}{(n-p)S_y} \quad (4.18)$$

Ve srovnání s determinacním indexem I^2 je sice opravený determinacní index přísnější na regresní funkce s větším počtem parametrů, ale tato přísnost se prakticky projevuje pouze při menších výběrech. Při větších výběrech bývá poměr $(n - 1) / (n - p)$ velmi blízký jedné, takže se oba determinacní indexy prakticky neliší [17].

5. Genetické algoritmy

První, kdo formuloval zákony dědičnosti a položil tak základy nauky o dědičnosti – genetiky, byl kněz Johan Gregor Mendel (1822 – 1884). Jeho teorie přenosu alel z rodičů do následných generací potomstva odpovídá jednoduchým kombinatorickým zákonům. Se vznikem počítačů se současně objevily myšlenky na sestavení systémů simulujících přirozenou evoluci, při které budou uplatněny zákony genetiky.

Genetické algoritmy (dále jen GA) jsou prohledávací metodou, založenou na principech přírodního výběru a genetiky. Poprvé byly prezentovány J. Hollandem, který definoval *operátor křížení* (crossover) a *operátor inverze*. Operátor křížení je považován za hlavní rozlišovací znak GA, které tento nekombinační operátor považují za primární.

5.1 Základní popis

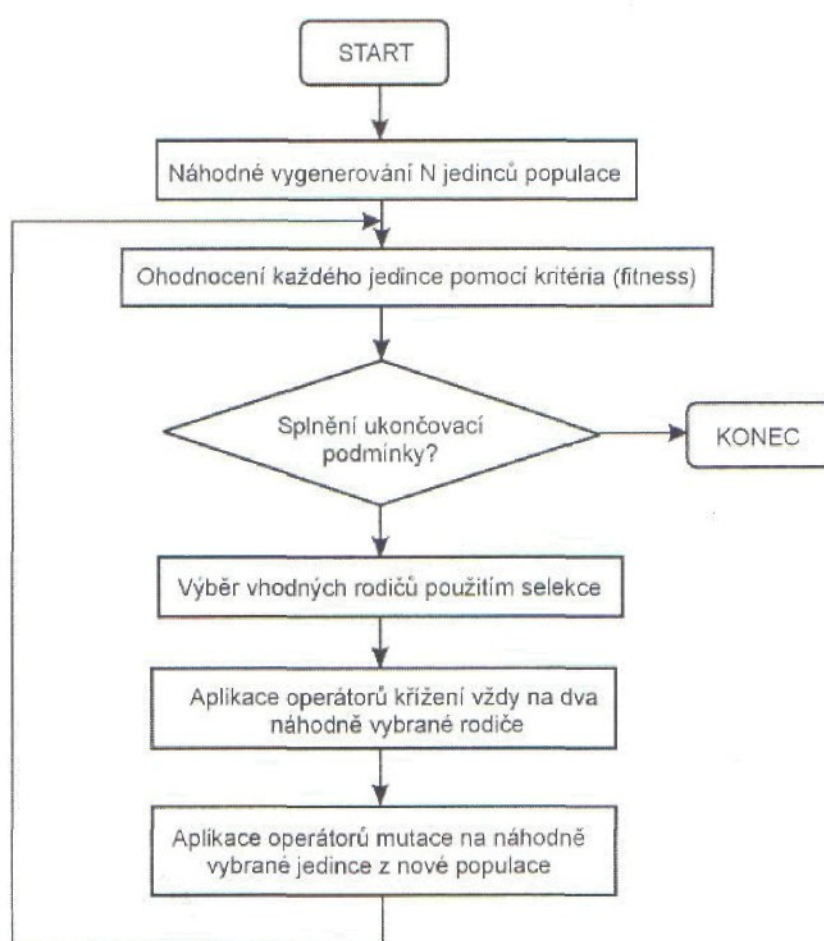
GA vycházejí ze základní Darwinovy teorie evoluce. Proto je řešení prováděné pomocí GA považováno za evoluční. Kandidát řešení je nazýván jedinec. Ve většině aplikací obsahuje jedinec jeden chromozóm. Jedinci jsou seskupovány do množin nazývaných *populace*. Po sobě následující populace se nazývají *generace*.

Řešení začínáme s počáteční populací jedinců. Členové současné populace následně ovlivňují pomocí mutace a křížení vznik další generace. Výběr těchto jedinců je dán kritériem, které přiřazuje každému jedinci míru kvality. Na základě kritéria jsou pak vybíráni jedinci (rodiče), aby se podílely na vzniku další generace. Tomuto postupu říkáme selekce.

Po selekci nejdříve proběhne tzv. křížení. Geny obou rodičů vytvářejí určitou cestou zcela nový chromozóm. Ten pak může následně mutovat. Mutací rozumíme, že části DNA se částečně pozmění. Tyto změny jsou způsobeny především chybami při kopírování genů od rodičů. Kvalita neboli způsobilost nového organismu je oceňována jeho úspěšností v dalším životě. Proces selekce, mutace a křížení, až po vytvoření nové populace, nazýváme souhrnně *reprodukcí*.

Motivací procesu reprodukce je větší kvalita další generace. Výběr rodičů, kteří jsou určeny pro vytvoření nové generace (populace potomků), je tedy prováděn podle jejich kvality (tzv. *fitness function*) – čím jsou vhodnější, tím mají větší naději na reprodukci. Tento výběr je založen na náhodném principu.

Uvedený postup se opakuje až do nalezení jedince s nejlepšími vlastnostmi (tj. optimálního řešení, pro které kritérium dosáhne extrémní hodnoty). Schéma algoritmu je uvedeno na obrázku (5-1).



obr. 5-1 Základní genetický algoritmus

Zdroj: [04]

Základní východiska pro užití GA jsou velmi obecná. Genetické algoritmy lze proto implementovat různým způsobem. První otázkou je jak vytvářet chromozomy a jaký způsob zakódování pro jejich popis vybrat. S tímto problémem jsou spojeny dvě základní operace, tj. křížení a mutace.

Druhou otázkou je výběr rodičů pro křížení, který může být proveden mnoha způsoby. Hlavní ideou je použít pouze lepší rodiče, protože se očekává, že budou produkovat lepší potomky. Vytvoření nové populace pouze z nových potomků může způsobit ztrátu nejlepších chromozomů z minulé populace (ztrátu genetického materiálu). Proto se často užívá tzv. *elitismus*. To znamená, že přinejmenším jedno nejlepší řešení (jedinec) je zkopírováno beze změn do nové populace.

5.2 Operátory genetických algoritmů

Jak bylo uvedeno v předchozím odstavci, křížení a mutace jsou dvě nejdůležitější operace v teorii GA. Rychlost a konvergence řešení jsou ovlivňovány hlavně těmito dvěma operacemi. Nejdříve si však objasníme význam chromozomů.

5.2.1 Kódování chromozomů

Každý chromozom musí nějakým způsobem obsahovat informace o řešení (jedinci), které reprezentuje. Ty mohou být popsány například vektorem, jehož prvky obsahují geny nebo také alely. Avšak nejužívanějším způsobem *zakódování* těchto informací je binární řetězec.

Chromozom 1	1101100100110110
Chromozom 2	1101111000011110

tab. 5-1 Popis chromozomu

Zdroj: vlastní

Každý chromozom má přiřazen jeden binární řetězec. Každý bit v tomto řetězci reprezentuje nějakou charakteristiku řešení, popř. celý řetězec reprezentuje jedno číslo, což bývá často užíváno v základní aplikacích.

Pro kódování GA existuje mnoho jiných způsobů. Výběr metody závisí především na řešeném problému. Někdy například pracujeme s celými čísly, jindy s reálnými čísly, nebo potřebuje zakódovat permutace atd.

5.2.2 Křížení

V případě, že jsme se již rozhodli o způsobu kódování chromozomů, můžeme pokročit k vysvětlení operace křížení. Při křížení jsou vybrány geny z rodičovských chromozomů a jsou vytvořeni noví potomci. Nejjednodušší cestou pro provedení tohoto výběru je vybrat náhodně bod křížení v chromozomu a vše před tímto bodem kopírovat z chromozomů prvního rodiče a vše za bodem křížení zkopírovat z chromozomů druhého rodiče. Příklad křížení chromozomů z tabulky (5-1) uvádíme v tabulce (5-2), kde bod křížení je označen svislou čarou za šestým znakem binárního řetězce.

Rodič 1	110110 0100110110
Rodič 2	110111 1000011110
Potomek 1	110110 1000011110
Potomek 2	110111 0100110110

tab. 5-2 Příklad křížení chromozomů

Zdroj: vlastní

Existuje mnoho dalších možností křížení. Můžeme například vybrat více bodů křížení. Křížení je výrazně závislé na kódování chromozomů. Specifické metody křížení mohou u vybraných problémů výrazně zrychlit řešení.

5.2.3 Mutace

Po provedení křížení se vytváří prostor pro mutaci, která je prevencí proti nalezení lokálního minima řešeného problému. Při této operaci se náhodně mění chromozomy nových potomků. Při binárním kódování můžeme změnit náhodně vybrané bity z 1 na 0 nebo z 0 na 1. Příklad mutace je uvedený v tabulce (5-3). Mutace závisí na způsobu kódování chromozomů stejně tak jako křížení.

Původní potomek 1	1101101000011110
Původní potomek 2	1101110100110110
Mutovaný potomek 1	1101111000011110
Mutovaný potomek 2	1101010100110110

tab. 5-3 Příklad mutace chromozomů

Zdroj: vlastní

5.3 Parametry genetických algoritmů

Při použití GA je třeba zadat hodnoty dvou základních parametrů: pravděpodobnosti křížení a pravděpodobnosti mutace [11].

Pravděpodobnost křížení (poměr křížení) vyjadřuje poměr počtu potomků, kteří vznikli křížením, k celkovému počtu jedinců v dané populaci. Pokud není provedeno křížení, jedinci jsou přesnou kopií rodičů. Pokud je provedeno křížení, potomci jsou složeni z částí rodičovských chromozomů. Jestliže je pravděpodobnost křížení rovna 100%, nová populace všech jedinců vznikla křížením. Jestliže je pravděpodobnost křížení rovna 0%, celá nová populace je přesnou kopií chromozomů staré populace jedinců. To však neznamená, že nová generace jedinců je shodná s původní generací. Jedná se opět o generaci nových jedinců.

Křížení je prováděno s nadějí, že chromozomy nových jedinců převezmou lepší části chromozomů rodičovských, a tudíž někteří noví jedinci budou mít lepší vlastnosti než jedinci generace minulé.

Pravděpodobnost mutace (poměr mutace) vyjadřuje poměr počtu potomků, u kterých došlo ke změně chromozomů mutací, k celkovému počtu jedinců v dané populaci. Pokud není provedena mutace, potomci zůstávají po křížení beze změny, nebo jsou pouze zkopírováni. Jestliže je mutace provedena, chromozomy části nových jedinců jsou změněny. Pokud je mutace 100%, chromozomy všech nových jedinců jsou změněny, a pokud je 0%, není měněno nic.

Mutace je preventivní ochranou před uváznutím řešení v lokálním extrému kritériální funkce na prohledávané oblasti. Tento problém se však u GA vyskytuje jen velmi málo, protože daná oblast je prohledávána náhodným způsobem, což vede v převážné většině řešených případů k nalezení globálního extrému kritériální funkce.

Kromě uvedených parametrů existují i jiné parametry. Z nich je nejdůležitější velikost populace, která vyjadřuje, kolik jedinců (chromozomů) se nachází v dané populaci (v jedné generaci). Pokud je v populaci velmi málo jedinců, GA mají málo možností k tomu, aby se uplatnilo křížení, což v konečném důsledku vede k tomu, že je prohledávána jen velmi malá část daného prostoru. Jestliže je však na druhé straně zadáno

příliš mnoho jedinců, GA konvergují ke správnému řešení jen velmi pomalu. Provedené výzkumy v této oblasti ukazují, že po dosažení určitého limitu v počtu jedinců (ten závisí především na způsobu kódování chromozomů a typu řešeného problému) už není užitečné ho dále zvyšovat, protože by to nepřispělo ke zvýšení rychlosti řešení.

5.4 Inicializace

GA začíná *inicializací* (počátečním nastavením) populace. Obvykle bývá všech n chromozomů inicializováno náhodně (pomocí pseudonáhodného generátoru čísel). V některých případech je však vhodné zvolit jiný druh inicializace, neboť zcela náhodné počáteční nastavení může způsobit zbytečnou ztrátu strojového času. Jako typický příklad lze uvést problém obchodního cestujícího, kde je vhodné při inicializaci zabránit duplicitě v chromozomu. V případě, že tak neučiníme, je třeba do fitness funkce zakomponovat postih za tento prohřešek pomocí penalizační funkce. V praxi však duplicitu ošetřujeme při inicializaci a při uplatnění operátorů křížení a mutace.

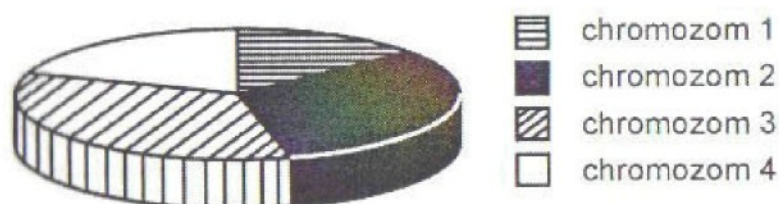
5.5 Selekce

Část jedinců (chromozomů) je vždy z populace vybrána, aby se stali rodiči pro operaci křížení [12], [13]. Problémem je jak tento výběr provést. Podle Darwinovy evoluční teorie by se měli pro křížení a tvorbu potomků používat pouze nejlepší jedinci. Existuje mnoho metod pro výběr nejlepších chromozomů, např. metoda nejlepšího výběru, uniformního výběru, rulety, pořadí, ustáleného výběru a řada dalších metod. Některé z těchto metod dále velmi stručně popíši:

- **Nejlepší výběr** – populace jsou vybráni jako rodiče pouze nejvyšší jedinci dle nejvyšší hodnoty kritéria.
- **Uniformní výběr** – rodiče jsou vybráni jako rodiče zcela náhodně z celé populace. Každý jedinec má stejnou pravděpodobnost, že bude vybrán.
- **Deterministický výběr** – deterministický selektor užívá dvoufázové selekční procedury. V první fázi se vypočítá předpokládané kritérium nových jedinců od všech rodičů. Dočasná populace je pak naplněna nejdříve jedinci s nejvyššími

očekávanými hodnotami kritéria. Ostatní pozice jsou následně obsazeny původními jedinci s nejvyšší pozicí v seznamu seřazeném podle předem stanoveného kritéria. Druhá fáze je uniformní náhodnou selekcí z dočasné populace.

- **Stochastický výběr** – selektor využívá také dvoufázové selekční procedury. V první fázi se vypočítá předpokládané kritérium nových jedinců od všech rodičů, kteří naplní dočasnou populaci jedinci s nejvyššími hodnotami. Ostatní pozice jsou následně obsazeny náhodně vybranými původními jedinci. Druhá fáze je opět uniformní náhodnou selekcí z této dočasné populace.
- **Metoda rulety** – rodiče jsou vybráni podle jejich kvality (hodnota kritéria) [12]. Čím mají jedinci kvalitnější chromozomy, tím mají větší šanci na výběr. Představme si kolo rulety, do kterého jsou umístěny všechny chromozomy z celé populace. Každý chromozom má svoje místo, jehož plocha je úměrná jeho kvalitě (hodnota kritéria). Kolo rulety se roztočí a je do něj vhozena kulička. Ta zůstane po čase stát na určitém místě a vybere svojí polohou určitého jedince (chromozom). Kvalitnější chromozomy s větší plochou na kole rulety mají větší pravděpodobnost výběru (viz obr. 5-2).



obr. 5-2 Metoda rulety

Zdroj: [04]

- **Metoda lineárního pořadí** – předchozí metoda výběru se stane problematickou, pokud se hodnoty kritérií jedinců v populaci značně liší [14]. Jestliže hodnota kritéria nejlepšího jedince dosáhne například 90% hodnoty celkového součtu hodnot kritérií ostatních jedinců, budou mít zbývající jedinci velmi malou šanci na výběr. Metoda pořadí v prvním kroku seřadí jedince podle hodnoty kritéria. Jedinec s nejnižší hodnotou kritéria má pořadí 1 a jedinec s nejvyšší hodnotou kritéria má nejvyšší pořadí N , které je rovno počtu jedinců v populaci.

Pravděpodobnost výběru jedince se tak váže k pořadí jedince, nikoliv k hodnotě jeho kritéria. Obrázky (5-3) a (5-4) ukazují, jak se situace podkladu pro výběr například metodou rulety změní, pokud hodnoty kritérií u příslušných jedinců nahradíme uvedeným pořadím.



obr. 5-3 Situace před setříděním (graf podle kritérií)

Zdroj: [04]



obr. 5-4 Situace po setřídění (graf podle pořadí)

Zdroj: [04]

Po provedené úpravě mají všechny chromozomy šanci být vybrány. Tato metoda však vede i při zpomalení konvergence ke správnému řešení, protože nejlepší chromozom se příliš neliší od ostatních s nižším pořadím.

- **Metoda exponenciálního pořadí** – od uvedení metody lineárního pořadí se liší pouze tím, že pravděpodobnost výběru jedince není po setřídění rozložena v populaci lineárně, ale je dána exponenciální závislostí. Tento selekční algoritmus patří z uvedené skupiny k nejlepším [12].
- **Metoda turnaje** – největším přínosem metody je absence požadavku na setřídění populace a jednoduchost vlastní selekce. Proto je turnajová selekce široce používána. Překladem anglického názvu tournament je soutěž nebo turnaj. Přesně v tomto duchu selekce probíhá – z N jedinců populace je vybráno t soutěžících. Do další generace postupuje nejlepší z těchto t jedinců. Celý postup se opakuje N -krát, kde N je počet jedinců v nové populaci. Metoda obvykle vybere častěji jedince s vyšším skóre než metoda rulety.

- **Metoda ustáleného výběru** – v tomto případě se nejedná o zvláštní metodu výběru rodičů. Hlavní myšlenkou je, že velká část chromozomů by měla zůstat zachována pro další generaci. GA pak pracují následujícím způsobem. V každé generaci je vybráno několik chromozomů (s vysokými hodnotami kritéria) jako rodičů pro vytvoření nových potomků. Následně je vybráno několik chromozomů (s nejnižšími hodnotami kritéria). Tyto chromozomy jsou vyjmuty a nahrazeny novými potomky. Zbytek populace je zkopírován do nové generace [14].

5.5.1 Elitismus

Idea *elitismu* již byla uvedena v předchozím textu. Pokud vytváříme novou populaci křížením a mutací, existuje velká šance, že ztratíme nejlepší chromozomy. Elitismus je název metody, která nejdříve zkopíruje nejlepší chromozomy (nebo několik nejlepších chromozomů) do nové populace. Zbytek nové populace je vytvořen klasickým postupem. Elitismus může významně zvýšit výkon GA, protože preventivně zabrání ztrátě nejlepšího dosud nalezeného řešení.

5.6 Doporučení

Doporučení jsou často výsledkem empirických studií, které byly zpracovány za použití binárního kódování [12],[14],[15].

Poměr křížení by se měl pohybovat v rozpětí 0,80 až 0,95. Některé výsledky však ukazují, že někdy jsou vhodnější hodnoty nižší, kolem 0,60.

Poměr mutace by měl vždy být velmi nízký. Nejlepší výsledky jsou dosahovány kolem 0,005 až 0,01.

Doporučená velikost populace se pohybuje v rozpětí 20 až 30 jedinců. Někdy se však používají poněkud vyšší hodnoty – asi 50 až 100 jedinců. Experimenty ukazují, že velké populace obvykle nezrychlí řešení a že velikost populace je závislá na kódování chromozomů a na velikosti kódovacích řetězců.

Pro výběr selekce je možné využívat metodu rulety. Metoda podle pořadí je však někdy vhodnější. Vždy by měla být užívána metoda elitismu, pokud ji nenahradíme jiným postupem, který zachová nejlepší jedince pro příští generaci. Někdy je vhodné též využít metodu ustáleného výběru.

Výběr metody kódování chromozomů závisí na řešeném problému. To se týká i výběru metod použitých pro operátory křížení a mutace.

5.7 Aplikace genetických algoritmů, jejich výhody a nevýhody

Genetické algoritmy mohou být užity při řešení celé řady problémů, především těch, kde se běžně využívají metody optimálního programování. Jde o vyhledání optimálního řešení problému podle zadané kritériální funkce. Zde mohou GA nahradit jiné standardní optimalizační algoritmy, protože jim hrozí menší riziko, že uváznou v pasti lokálního extrému. Přitom se uplatňuje především výhoda jejich paralelismu, kdy jedinci populace cestují prohledávaným prostorem současně na mnoha různých místech. Výhodou genetických algoritmů je i jejich snadná implementace. Pokud již máme k dispozici algoritmus pro řešení úlohy určitého typu, nebývá obvykle problém s jeho přenesením na obdobnou úlohu.

Nevýhodou genetických algoritmů bývá obvykle větší spotřeba strojového času než u klasických optimalizačních metod. Mohou být pomalejší než jiné standardní algoritmy, např. gradientní metody. Tato okolnost však již není problémem vzhledem výkonnosti současné výpočetní techniky.

Na závěr bych rád uvedl některé oblasti aplikací GA:

- *nelineární dynamické systémy* – predikce, analýzy dat,
- *umělé neuronové sítě* – topologie, proces učení,
- *neuro-fuzzy genetické systémy* – topologie, proces učení,
- *genetické programování* – vývoj programů psaných v jazyce LISP,
- *strategické plánování*,
- *řešení problémů obchodního cestujícího* atd.

6. Neuronové sítě

Umělé neuronové sítě (dále jen NS) dominují v rozpoznávání okamžité závislosti ze vzorů vstupně-výstupních vztahů. Tím se liší od expertních systémů, které dosahují výborných výsledků v posloupnosti logických operací, a fuzzy logických metod, které se vyznačují schopností reprezentovat znalosti.

První informace o NS bývá spojována se jmény McCullocha a Pittse (1943), kteří sestavili první neuronový model. V roce 1949 navrhl Donald Hebb model založený na biologické podobnosti, který je schopný učení. Na popsání principů je založena řada moderních učících se síťových mechanismů. V padesátých letech dvacátého století Frank Rosenblatt vyvinul třídu neuronů označovaných jako *perceptrony* [1]. Jeho přístup výrazně posunul vpřed teorii založenou na rozpoznávání vzorů a asociativním učení.

V roce 1982 publikoval profesor John Hopfield řadu článků o NS, ve kterých popsal zpětnou vazbu mezi neurony, nelinearitu přenosových funkcí, koncept globální energetické funkce, teorii energetických stavů a teorii minimalizace energetického toku. V témže roce se objevily první aplikace teorie zpětného šíření (backpropagation) při učení NS. V roce 1992 bylo vyvinuto první hardwarové řešení neuronových sítí – neuronové karty, které se vkládají do počítače. Hardwarové řešení urychlilo proces učení neuronových sítí více než tisíckrát.

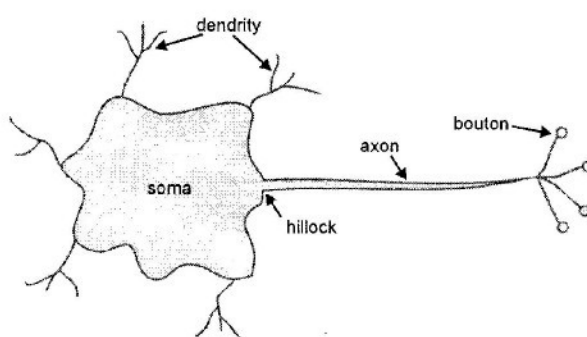
V naší odborné veřejnosti lze zaznamenat první články a publikace s popisem NS až po roce 1988.

6.1 Základní pojmy

Lidský mozek se sestává ze sítě mnoha bilionů speciálních buněk, které se nazývají neurony. Typický neuron znázorněný na obrázku (6-1). Se skládá z těla neuronu (soma), z něhož vybíhá nervové vlákno (axon). To je zakončené rozvětvením směřujícím k ostatním neuronovým buňkám v síti, s jejichž těly tvoří elektrochemickou vazbu (synapse) [2].

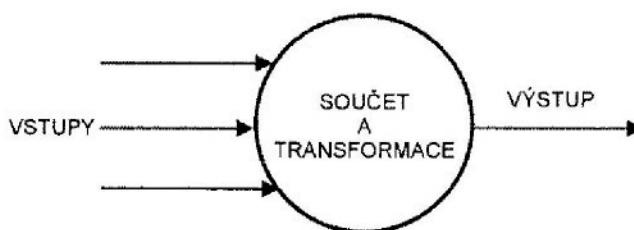
Dendrity, které spojují neuron s ostatními pomocí synapsí, přenášejí podráždění (vstupní signály) do somy. Zde jsou tyto signály sečteny a je rozhodnuto podle jejich síly a povahy, zda budou propuštěny a s jakou silou (filtrace přes membránu - hillock) do axonu. Ten pak přenáší signál do ostatních neuronů přes boutony s různou velikostí a zpožděním. Každá neuronová buňka se chová jako extrémně jednoduchý počítač [3], jehož schéma je uvedeno na obrázku (6-2).

Umělý neuron přijímá vstupní signály (vstupy) a vysílá výstupní signál (výstup). Každý neuron přijímá vstupy z mnoha jiných neuronů. Umělou NS definujeme jako orientovaný graf $G(N,H)$, ve kterém množinu vrcholů N tvoří těla neuronů n_i a množinu hran H tvoří spojnice neuronů $h_{i,j}$. V tomto grafu jsou jednotlivé neurony uspořádány do vrstev podle obrázků (6-3) tak, že vstupní vrstvu tvoří vstupní neurony, dále následuje skrytá vrstva neuronů (může být i více skrytých vrstev) a vrstva výstupních neuronů. Neurony jsou v NS průběžně očíslovány. Každý neuron v libovolné skryté vrstvě je spojen hranou s každým neuronem vrstvy předchozí i vrstvy následné.



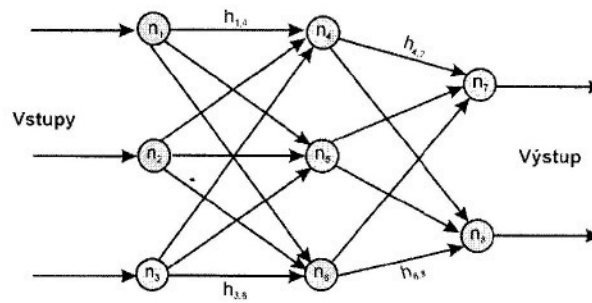
obr. 6-1 Biologický neuron

Zdroj: [04]



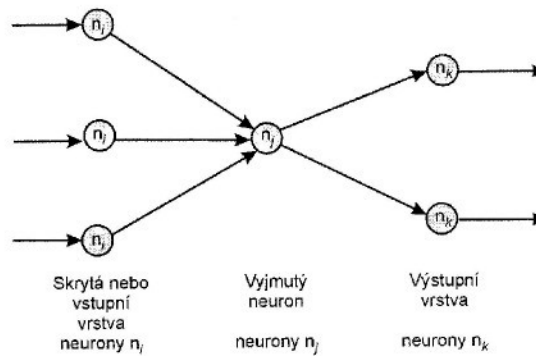
obr. 6-2 Umělý neuron

Zdroj: [04]



obr. 6-3 Umělá neuronová síť

Zdroj: [04]



obr. 6-4 Neuron n_j vyjmutý ze skryté vrstvy

Zdroj: [04]

Funkci neuronu v NS je možné popsat na příkladu neuronu n_j , který je znázorněn na obrázku (6-4). Zde jsou současně zobrazeny tři neurony n_i z předchozí vrstvy, které mu signál předávají, a symbolicky dva neurony n_k z následující výstupní vrstvy, které od něj signál přijímají. Označme u počáteční a v koncové číslo neuronu n_i předávajícího signál neuronu n_j podle průběžného číslování neuronů v NS. Dále označme O_i signál vystupující z i -tého neuronu a $O_{i,j}$ signál procházející hranou $h_{i,j}$. Platí, že dolní indexy znázorňují směr postupu signálu $O_{i,j}$ hranami $h_{i,j}$ z neuronu n_i do neuronu n_j , jejichž počáteční číslo značíme r a koncové s . Necht' $W_{i,j}$ a $W_{j,k}$ jsou synaptické váhy, kterými je vynásobena velikost signálu $O_{i,j}$ a $O_{j,k}$, a to vždy na konci příslušné spojnice. Pro signál O_j (stav neuronu) vystupující z j -tého neuronu pak platí

$$O_j = TF_j \left(\sum_{i=u}^v W_{i,j} O_{i,j} - \Theta_j \right) \quad (6.1)$$

Kde $TF_j(\cdot)$ značí přenosovou funkci j -tého neuronu a Θ_j představuje tzv. práh j -tého neuronu.

Vztah

$$A_j = \left(\sum_{i=u}^v W_{i,j} O_{i,j} - \Theta \right) \quad (6.2)$$

budeme dále nazývat aktivační funkcí (potenciálem neuronu). Pokud je aktivační funkce kladná, dochází ke vzplanutí neuronu (neuron je excitován) a propustí signál dále. Pokud jsou váhy kladné, přispívají ke vzplanutí, pokud jsou záporné, působí proti vzplanutí. Je zřejmé, že ke vzplanutí může dojít, až když

$$\sum W_{i,j} O_{i,j} \quad (6.3)$$

překročí mezní hodnotu – práh Θ_j . Z formálních důvodů je práh někdy označován jako nultý vstup do j-tého neuronu s hodnotou signálu -1 a s váhou Θ_j .

Umělá NS nemůže pracovat spojitě. Umožňuje pouze pro dané hodnoty vstupních signálů vstupujících do vstupních neuronů poskytnout odpovídající hodnoty signálů výstupních. Aby byla schopna plnit tuto funkci, je třeba znát minimálně hodnoty vah $W_{i,j}$, parametry přenosových funkcí a hodnoty prahů, při kterých dochází ke vzplanutí neuronů. Při složitějších typech neuronových sítí může být neznámých veličin více. Smyslem učení (trénování) NS je najít hodnoty těchto veličin.

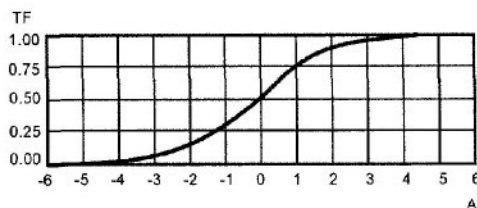
Podle toho, zda přenosová funkce je lineární nebo nelineární, můžeme vytvořit modely lineárních nebo nelineárních neuronových sítí. V nejjednodušším případě lze u lineárního modelu rovnici (6.1) přepsat do tvaru, ve kterém je přenosová funkce rovna aktivační funkci

$$O_j = A_j = \sum_{i=u}^v W_{i,j} O_{i,j} - \Theta_j \quad (6.4)$$

Pokud jsou vrcholy a hrany grafu $G(N,H)$ uspořádány, jak vidíme na obr. (6-3), a signál tedy postupuje pouze jedním směrem – zleva doprava, mluvíme o **jednosměrných** NS (feed – forward networks). Podle autorů nebo uspořádání byla nazvána řada typických modelů sítí: Hopfieldova, Perceptron, Kohonenova, BAM, ART atd.

Jako přenosové funkce $TF(A)$ se v umělých neuronových sítích používají následující typy funkcí:

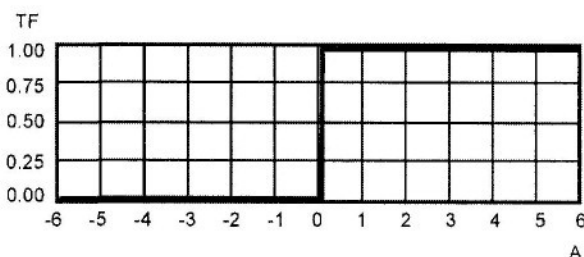
- **Logistická sigmoidální funkce** (sigmoida), jejíž tvar je na obrázku (6-5). Je nelineární, spojitá a spojitě diferencovatelná a má tudíž mnoho žádoucích vlastností, vhodných pro využití v neuronových sítích.



obr. 6-5 Sigmoida

Zdroj: [04]

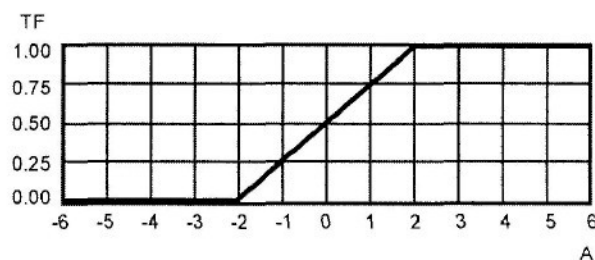
- **Skoková přenosová funkce**. Funkce je limitována dvěmi možnými hodnotami. Středem je vždy hodnota vstupu, při kterém se hodnota výstupu skokem změní (obr. 6-6). Vzhledem k nespojitosti je nelineární funkcí. Je vhodná zejména do aplikací, ve kterých neurony v neuronové síti fungují jako přepínače. Tyto typy funkcí byly často využívány v počátečních aplikacích neuronových sítí.



obr. 6-6 Skoková přenosová funkce

Zdroj: [04]

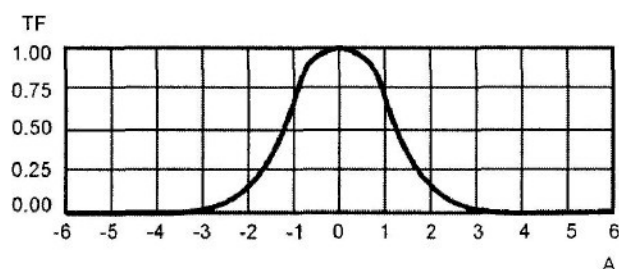
- **Lineární prahová funkce**. Neuronové sítě, které ji používají jsou schopny aproximovat značně složitější vstupně – výstupní vztahy než lineární neuronové sítě (viz obr 6-7).



obr. 6-7 Lineární prahová přenosová funkce

Zdroj: [04]

- **Gaussova přenosová funkce.** Gaussovy přenosové funkce jsou známé jako zvonové funkce. Jejich užití je dosti omezené. Tato funkce je spojitá a spojitě diferencovatelná. Proto může být využita stejně jako sigmoida. Poskytuje však velmi nezvyklé a nekonzistentní výsledky. Oblasti použití jsou stále předmětem výzkumu (obr. 6-8).



obr. 6-8 Gaussova přenosová funkce

Zdroj: [04]

6.2 Způsob učení

Nechť je dána tréninková matice **A**, ve které jsou do určitých sloupců zapsány hodnoty vstupních signálů a do zbývajících hodnoty výstupních signálů. Vždy jeden řádek odpovídá jednomu vztahu mezi vstupem a výstupem – tvoří jeden vzor. Schéma této matice je znázorněno v tabulce (6-1). Vstupní veličiny X_1 , X_2 , X_3 a X_4 jsou zapsány do vstupních sloupců označených symbolem $\downarrow$, výstupní veličiny Y_1 a Y_2 jsou pak uvedeny ve sloupcích výstupních, označených symbolem $\uparrow$. Číslo uvedené ve sloupci i udává pořadí příslušného vzoru.

	↓	↓	↓	↓	↑	↑
I	X₁	X₂	X₃	X₄	Y₁	Y₂
1	0,4	12,7	6,3	5,5	1,2	0,0
2	0,8	15,6	3,8	21,7	3,3	0,2
3	0,6	11,1	4,3	29,6	4,4	0,5
.	.	.	.	.	.	.
.	.	.	.	.	.	.
n	0,3	8,5	3,7	18,9	5,1	0,4

tab. 6-1 Schéma tréninkové matice

Zdroj: [04]

V průběhu učení (trénování) vstupují postupně (po řádcích) do NS vstupní signály (tréninkové vzory). V závislosti na velikosti vah a prahů pak vystupují z výstupních neuronů hodnoty výstupních signálů. Postup se opakuje podle počtu řádků tréninkové matice. Zadané výstupní signály a signály vypočtené z NS vykazují odchylku. Součet čtverců všech odchylek vypočteny pro každý vzor zvlášť je chybou vzoru E_v . Vztah, podle kterého je vypočítána chyba příslušného vzoru je dán předpisem

$$E_v = \frac{1}{2} \sum_j (y_j - d_{vj}), \quad (6.5)$$

v němž index j probíhá přes neurony výstupní vrstvy a d_{vj} je j -tý element požadovaného výstupu v -tého trénovacího vzoru.

Součet chyb vypočtených ze všech vzorů je celková chyba sítě E . V procesu trénování se hledají takové hodnoty vah, aby pro všechny řádky tréninkové množiny tyto odchylky nepřekročily povolenou toleranci. Předpis pro celkovou chybu sítě je definován jako

$$E = \sum_v E_v, \quad (6.6)$$

kde index v probíhá přes všechny trénovací vzory a E_v je chyba odpovídající v -tému trénovacímu vzoru.

Trénování NS současně pro všechny vzory představuje optimalizační problém, při kterém se minimalizuje celková chyba E , vypočtená ze všech vzorů. V úloze je vektorem neznámých vektor $\mathbf{w}$, jehož prvky tvoří neznámé váhy, popř. i parametry přenosových funkcí a hodnoty prahů. Úkolem je nalézt takovou hodnotu vektoru $\mathbf{w}$, při které všechny uvedené odchylky nebo předepsané procento odchylek nepřekročí předepsanou toleranci. Kritériem může být rovněž součet čtverců odchylek, směrodatná odchylka, průměrná odchylka nebo koeficient determinace apod. Protože dokonce i malé NS mají stovky a tisíce spojníc, mluvíme o „*minimalizaci skalárního pole na vektorovém prostoru se stovkami dimenzí*“. Pro řešení popsaného problému může být použita řada *optimalizačních metod* [4].

Nejužívanější metody, vhodné pro nalezení minima kritériální funkce, jsou *gradientní metody*. Metoda gradientního poklesu zahrnuje změnu proměnných s malým krokem změn směrem dolů ve směru lokálního gradientu. Uvedený postup má jeden podstatný nedostatek v tom, že lze snadno najít lokální minimum řešení místo globálního maxima, které je cílem řešení. Opustit oblast lokálního minima a pokračovat v dalším řešení však může být obtížné.

V oblasti jednosměrných NS se velmi často používá *metoda zpětného šíření* (backpropagation), kterou je možné přiřadit mezi gradientní metody. Při jejím užití se síť postupně učí pomocí korekce vah v závislosti na odchylce mezi zadanou a vypočtenou hodnotou výstupních signálů ve výstupních neuronech sítě. Korekce signálů (vah) pak probíhá v síti zpětně od výstupní vrstvy neuronů po vrstvu vstupní – podle toho získala svůj název.

Proces trénování sestává z postupně opakovaných výpočtů výstupů z NS při daných vahách pro každý vzor, kdy signály postupují od vstupní do výstupní vrstvy neuronů (forwards) a je vypočtena chyba vzoru E_v . Následně pak probíhá zpětným směrem (backwards) postupný výpočet oprav jednotlivých vah.

Hodnoty vah sítě upravíme podle vztahu

$$w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij}, \quad (6.7)$$

kde změny vah získáme ze vztahu

$$\Delta w_{ij} = -\eta \frac{\partial E}{\partial w_{ij}}, \quad (6.8)$$

kde η je **parametr učení** (learning parametr), $0 \leq \eta \leq 1$. Parciální derivace chyby podle vah sítě uvedená v tomto vztahu představuje minimalizaci chyby pomocí vah gradientní metodou. Váhy mohou být aktualizovány až po vyhodnocení všech trénovacích vzorů, tzv. **akumulované učení** (accumulated learning, off-line backpropagation true gradient method), kde nezáleží na pořadí, v jakém jsou vzory předkládány na vstupy sítě a nebo mohou být okamžitě aktualizovány po každém předloženém vzoru. Je zřejmé, že tato varianta algoritmu nemůže implementovat celkový (opravdový) gradientní pokles chyby E , ovšem praxe ukázala, že metoda je funkční v naprosté většině testovaných příkladů. V každém případě je celý proces opakován tak dlouho, dokud chyba není menší než stanovená mez [7].

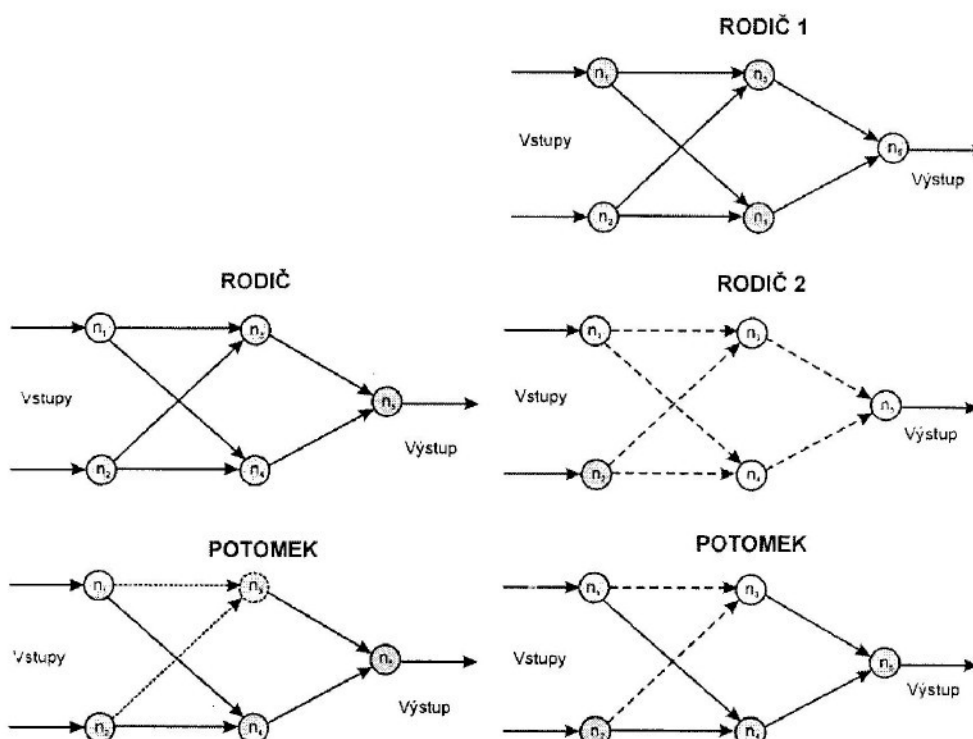
Pokud zavedeme do řešení hodnoty prahů Θ jako nulté vstupy do neuronů s hodnotou signálu -1 a s váhou Θ , pak se hodnota signálu u každého neuronu v průběhu trénování nemění a zůstává rovna -1. Opravy změn prahů se pro každý vzor počítají stejným způsobem jako u změn vah.

Při trénování NS lze užít i *genetických algoritmů* (GTO – Genetic Training Option). Algoritmus, který řídí proces učení, vychází ze dvou známých přístupů – mutace a křížení, které jsou založeny na biologické genetické evoluční teorii. Aplikujeme-li genetické algoritmy na NS, je možné nahlížet na spojnice (hrany) mezi neurony a jejich váhy jako na řetězec DNA v živých organismech.

Mutace vyžaduje pouze jednoho rodiče. Během ní je náhodné procento neuronů změněno modifikací vah v hranách, které je spojují. Frekvence změn a velikost jejich změny může být zadána uživatelem. Obrázek (6-9) znázorňuje NS rodiče a neuronovou síť

potomka. V tomto příkladu mutoval skrytý neuron n_j z rodiče na potomka. Váhy u příslušných hran, které směřují do tohoto neuronu, se změnily.

Křížení vyžaduje dva rodiče. V neuronových sítích je možné je implementovat tak, že ze dvou neuronových sítí – rodičů vytváříme třetí – potomka. Na obrázku (6-10) je znázorněno křížení dvou NS. Potomek obdržel skrytý neuron n_j od druhého rodiče a první skrytý neuron od prvního rodiče (rozumí se váhy u příslušných hran).



obr. 6-9 Schéma mutace neuronové sítě

obr. 6-10 Schéma křížení neuronových sítí

Zdroj: [04]

Po provedené mutaci nebo křížení je testován potomek – nová NS. Pokud vykazuje potomek lepší vlastnosti než jeden nebo oba rodiče, nová NS nahradí rodiče a nebo jednoho z rodičů. Přenášejí se tak informace na budoucí generace NS. GTO umožňuje kombinovat mutace a křížení. Protože mutace a křížení jsou aplikovány na váhy hran v NS, je třeba při spuštění GTO vycházet z natrénované nebo částečně natrénované NS, tj. mít dostatečný zásobník rodičů pro NS [4].

6.3 Topologie neuronových sítí

Jedním ze zásadních problémů při konstrukci NS je určení její *topologie*. Především je nutné zvážit, k jakému účelu bude NS sloužit, jaké budeme zadávat veličiny vstupní a co bude výstupem. Musíme promyslet, které veličiny mohou mít vliv na výstup NS. V této fázi je třeba zapomenout na různé procedury, pravidla a nebo vzorce. Je třeba uvažovat o veličinách a jejich možných vztazích jako takových. Různorodost vstupních dat s vazbou na veličinu výstupní zvyšuje šanci nalézt významnou korelaci mezi veličinami. Dále je třeba si uvědomit, zda budeme mít k dispozici dostatek vzorů obsahujících veličiny vybrané jako vstupy a výstupy NS. Je zřejmé, že při určení počtu neuronů ve vstupní a výstupní vrstvě se po provedení těchto úvah nevyskytnou žádné větší problémy. Počet vzorů je dán počtem vstupních a výstupních veličin. Dimenze vstupní a výstupní vrstvy je tedy dána charakterem řešené úlohy. Problémy však nastávají při určení počtu skrytých vrstev neuronů, počtu neuronů v těchto vrstvách a nutném počtu tréninkových vzorů. Problém volby počtu skrytých vrstev a skrytých neuronů je záležitostí každého řešitele. Doporučení pro tuto volbu lze nalézt v řadě publikací [3], [5], [6] a v dalších pracích.

Doporučení pro volbu počtu skrytých vrstev v NS je problematické. K použití nejjednodušších třívrstvých sítí se přiklání řada autorů, nicméně téměř stejný počet autorů aplikuje více skrytých vrstev. Přitom není prokázáno, že více skrytých vrstev vede jednoznačně k lepším výsledkům. Praxe ukazuje, že je vhodné započít trénování s jedinou vrstvou skrytých neuronů. Literatura uvádí [5], že libovolná funkce může být aproximována již třívrstvou NS s $2m + 1$ neurony ve skryté vrstvě, kde m je počet vstupních neuronů. Pokud v NS navrhujeme více skrytých vrstev, dostáváme se do méně prozkoumané oblasti a musíme postupovat obezřetně. Především podstatně prodloužíme dobu trénování. Pokud navržená síť nejde natrénovat, je vhodné zmenšit počet vrstev a znovu začít s trénováním.

Volba počtu skrytých neuronů je rovněž problémem a pro zodpovězení otázky, týkající se počtu skrytých neuronů, neexistuje žádný univerzální vzorec. Velký počet skrytých neuronů může vést k „zapamatování“ si vzorů. Jestliže však máme malý počet neuronů, NS se nedá dostatečně natrénovat. Lze uvést několik doporučení k určení postačujícího počtu skrytých neuronů. Jedním je natrénovat několik NS s různým počtem skrytých neuronů a vybrat z nich tu, která dává při testování nejlepší výsledky. Jiným

doporučením je určit počet neuronů ve skryté vrstvě jako průměrnou hodnotu z počtu vstupních a výstupních neuronů. Dalším doporučením je započít trénování s menším počtem skrytých neuronů, a pokud se nedaří síť na požadovanou toleranci natrénovat, postupně přidávat další skryté neurony, dokud se schopnost učení nezlepší. Je však možné postupovat i opačně. Zpočátku trénovat NS s větším počtem skrytých neuronů a následně vyjmout spojnice a nebo celé neurony, které poskytují velmi malé výstupní signály. Pak je třeba zopakovat trénování NS se zmenšeným počtem skrytých neuronů. Uvedená technika se nazývá „*prostřih*“.

Pokud má NS problémy s dosažením správných výsledků, může být jednou z příčin malý počet tréninkových vzorů. Tento problém je statistického rázu. Když poměr počtu tréninkových vzorů k počtu neuronů klesá, narůstá pravděpodobnost vzniku „*libovolného*“ vztahu mezi vstupy a výstupy. Čím méně tréninkových vzorů v porovnání s počtem neuronů máme k dispozici, tím je větší pravděpodobnost, že síť sice natrénujeme, ale ta nebude naučena správně.

S ohledem na nutný počet tréninkových vzorů doporučuje jedno z pravidel volit počet parametrů modelu (vah neuronové sítě) menší než počet všech vzorů v trénovacím souboru [6]. V opačném případě hrozí přetrénování (overfitting) a ztráta generalizační schopnosti NS, tj. síť má příliš mnoho spojů mezi neurony a není schopna dosáhnout zobecnění vztahu mezi vstupy a výstupy. Příliš mnoho skrytých neuronů může tedy způsobit, že si NS „*zapamatuje*“ vzory, místo aby se podle nich naučila. Neuronové sítě bude možné velmi dobře trénovat, s jejími testy to však bude podstatně horší. Ve spojitosti s tímto jevem se užívá termín, že NS je „*přetrénovaná*“. Pro dosažení správných výsledků je vhodné trénovat „*hubenější*“ NS s menším počtem skrytých neuronů, ale pro trénink použít dostatečný počet vzorů vstupních dat, rovnoměrně pokrývajících řešenou oblast. Běžná praxe ukazuje, že počet tréninkových vzorů by měl být přinejmenším roven součinu počtu skrytých neuronů a reciproké hodnoty požadované tolerance. Pokud máme například 10 skrytých neuronů a požadovanou toleranci rovnu 0,1, počet tréninkových vzorů by měl být přinejmenším $100 = 10 (1/0,1)$.

Přidáním náhodného „*šumu*“ k tréninkovým datům můžeme napomoci procesu učení NS. Šum umožní částečně změnit data, kdykoliv jsou předložena k trénování. Přidání šumu je obzvláště užitečné, pokud máme omezený počet vzorů dat. Nejvhodnější velikost

šumu činí proces trénování obtížnějším, ale ne nemožným. Přidání šumu ke vstupním datům se osvědčuje, když tréninková data obsahují mnoho nulových hodnot. Neuronová síť se mnohem lépe učí, pokud vstupní vzory obsahují místo nul malé hodnoty čísel. Někdy se osvědčuje po natrénování NS přidat šum ke vstupním datům a pokračovat v trénování. Takto dotrénované síť v praxi často vykazují velmi dobré výsledky. Neuronová síť totiž mohla uváznout v pasti lokálního minima kritériální funkce a „rozkmítání“ vstupních dat jí může pomoci s takovéto pasti vyváznout. Kombinace zašumění vstupních dat a prostřihování skrytých neuronů napomáhá k zobecňování závislostí mezi daty ve vstupních vzorech.

Konstrukce NS rovněž vyžaduje určení typu přenosových funkcí a jejich rozsahu, určení koeficientu rychlosti učení a vyhlazovacího faktoru. Rychlost učení určuje velikost oprav neznámých veličin v procesu učení (váhy, prahy, strmosti atd.). Vyhlazovací faktor určuje míru vlivu, s níž jsou uvažovány minulé korekce opravovaných veličin, a míru vlivu, s níž jsou uvažovány korekce nově vypočtené (konjugované gradienty). Dosud byla popsána základní stavba vrstvené NS. V této síti tvoří neurony vrstvy. Neurony dvou sousedních vrstev jsou propojeny systémem hran každý s každým. Existuje však řada dalších možných architektur vrstvených NS, které uvedený základní systém rozšiřují a doplňují dalšími spoji mezi neurony. Příkladem jsou rekurentní síť, ty se snaží zdokonalit proces učení zavedením spojů mezi neurony jednotlivých neuronových vrstev. Rekurentní neurony vznikly jako pokus obohatit NS o časový rozměr. Síť s rekurentními neurony se hodí pro zpracování dat, které tvoří časové řady.

6.4 Standardizace dat

Standardizací vstupních dat rozumíme transformaci, která je převede na požadované intervaly. Tyto intervaly mohou být různé a mohou se odlišovat podle toho, zda se jedná o vstupy nebo výstupy NS. Pak mluvíme o standardizaci vstupních nebo výstupních dat. V zásadě může být tato transformace jak nelineární, tak lineární. Transformaci je nutno provést ještě před započítím trénování sítě a v řadě případů je podmínkou úspěšného trénování.

Například při použití sigmoidy jako nelineární přenosové funkce, jejíž výstupy leží v intervalu $(0,1)$, je transformace vstupních dat nutností a plyne z definice této funkce. Požadovaná výstupní data je třeba přetransformovat rovněž na interval $(0,1)$. Po natrénování NS a provedeném výpočtu pro nová vstupní data jsou pak výstupy sítě inverzní transformací převedeny na interval skutečných hodnot výstupních dat. Vstupní data by teoreticky nemusela být pro sigmoidu standardizována, protože definiční obor sigmoidální přenosové funkce je $(-\infty, +\infty)$. Ve skutečnosti je však standardizace vstupu vhodná. Ukázalo se, že učení NS je mnohem rychlejší, když jsou vstupní data symetrická. Proto se transformují nejčastěji do symetrického intervalu kolem nuly. Dále se doporučuje, aby vstupní data byla přetransformována do menších hodnot. Důvodem je malá citlivost sigmoidální funkce při větších absolutních hodnotách vstupů.

Pokud vzory dat určené pro trénování zapíšeme do matice (odst. 6.2), jsou v zásadě možné dva způsoby standardizace. Jednak je to standardizace po sloupcích matice a jednak standardizace po řádcích matice. Častěji se používá standardizace po sloupcích, kdy standardizujeme každý sloupec zvlášť. Každý vstup do NS je tvořen obecně jinou veličinou, která má svůj vlastní fyzikální či jiný význam. Je proto přirozené, že také rozdělení každého vstupu je jiné a má jiný rozsah hodnot. V běžných případech se každá veličina standardizuje do téhož intervalu. Za předpokladu normálního rozdělení vstupní veličiny je možné každý vstupní údaj příslušného sloupce transformovat na normované normální rozdělení.

Standardizace výstupních veličin je závislá na typu přenosové funkce. U standardizace výstupních veličin je nutné, aby maximální a minimální možné hodnoty dané veličiny byly dobře zvoleny. Často se vyplatí dosadit za tyto extrémní hodnoty čísla, které tréninková množina třeba ani neobsahuje, ale jsou přitom pro danou veličinu reálná. Jinou možností je standardizovat výstupní veličinu preventivně do intervalu $(0,1; 0,9)$. Tato standardizace urychlí trénink a ponechává jistou rezervu pro testování, kdy se mohou objevit větší nebo menší hodnoty údajů než při tréninku. V oblasti extrapolace výstupních dat je třeba postupovat obezřetně. Praxí je ověřeno, že NS se vyznačují velmi dobrou aproximací funkcí uvnitř řešené oblasti. Extrapolace funkcí mimo tréninkovou oblast je jejich slabší stránkou a lze ji provádět jen v omezené míře.

Obecně platí, že při minimalizaci kritériální funkce, dané součtem čtverců odchylek, jsou nejlepší výsledky dosaženy, jestliže vstupní a výstupní veličiny v matici vzorů dat mají normální rozdělení pravděpodobnosti [8]. Často je však rozdělení těchto veličin nesymetrické. Pak je třeba příslušnou transformaci provést ve dvou krocích. V prvním se asymetrie odstraní nebo zmenší logaritmováním všech prvků daného souboru. Získaný soubor dat má rozdělení normální nebo téměř normální. V druhém kroku se provede popsaná transformace prvků tohoto souboru na normované normální rozdělení.

6.5 Neuronové sítě a regresní modely

Mezi NS a regresními modely lze nalézt analogii. Zejména vícerozměrný *lineární regresní model* připomíná svým tvarem vztah pro výstupní neuron jednosměrné lineární NS s jedinou vrstvou vstupních neuronů, jediným výstupním neuronem a s nulovým prahem u všech neuronů.

Je zřejmé, že lze sestavit NS, které mohou nahradit i vícerozměrné nelineární regresní modely. Jde například o polynomičké regresní modely, mocninné regresní modely i další velmi složité typy regresních modelů. Neuronové sítě a regresní modely mají mnoho společného. Zejména to však platí pro velmi jednoduché jednosměrné vícevrstvé NS. Značným problémem u vícerozměrných nelineárních úloh je volba vhodného tvaru regresní rovnice. Tyto vztahy se nedají znázornit jednoduchým grafem a řešitel často sklouzne do oblasti intuitivní a uvedený problém řeší metodou pokusů a omylů. Opřít se tu můžeme jenom o své zkušenosti z předchozích prací, literaturu a zkušenosti svých kolegů.

Samostatným problémem je kalibrace nelineárního regresního modelu (výpočet regresních koeficientů). Řešitel musí být dobře obeznámen s nelineární optimalizací a musí mít k dispozici adekvátní software umožňující tuto operaci provést. Problémem je i volba kritéria, podle kterého bude kalibrace provedena. Ta má pro způsob proložení regresní křivky množinou bodů rozhodující význam. Pokud pro tento účel použijeme komerční software, dostaneme se do pastí nabídky tvaru regresních funkcí, která nemusí být postačující.

Výhody NS oproti regresním modelům je možné shrnout do následujících bodů:

- NS nevyžadují volbu regresní závislosti. Vyžadují pouze volbu počtu skrytých vrstev, počtu neuronu v těchto vrstvách a volbu tvaru přenosových funkcí. Při určitých zkušenostech s jejich použitím však lze tyto volby provést rutinně. Navíc lze vícevrstvé nelineární NS použít k aproximaci téměř libovolné nelineární funkce. Stávají se tak univerzálními aproximátory.
- Učení (trénink) NS je analogickým termínem pro kalibraci regresního modelu. Neuronové sítě jsou však v procesu učení mnohem obecnější, samy si vyberou, kterému vstupu a parametru mají určit větší váhu a optimalizují větší počet parametrů.
- NS mohou obsahovat více výstupů jejichž hodnotami odpovídají na zadané hodnoty vstupu, což je pro běžné regresní modely nedostižné.

Závěrem můžu uvést, že pro jednoduché úlohy se lépe hodí použít klasické statistické regresní modely. Technika kalibrace těchto modelů je dostatečně propracovaná. Navíc umožňují určit meze spolehlivosti, konfidenční interval a řadu další důležitých údajů, jejichž určení se u komerčních softwarů, modelujících NS vyskytuje jen zřídka. Pro složitější úlohy z oblasti vícerozměrné nelineární regrese je však vhodnější použít neuronové sítě, zejména pokud požadujeme souběžně více výstupů. Hodně však záleží na názorech a zkušenostech řešitele a na dostupnosti použitého softwaru. Řešitel se sám musí rozhodnout jakou technologii pro řešení zvolí.

6.6 Oblasti využití neuronových sítí

Odpovědi na otázku, jaké problémy lze pomocí NS řešit, je možné shrnout do několika bodů [3]:

- NS jsou vynikajícím prostředkem pro pozorování závislostí mezi vstupními a výstupními údaji. Pokud potřebujeme nějakou závislost rozpoznat a oklasifikovat, NS daný problém vyřeší vždy rychleji a přesněji než člověk. Neuronová síť dokáže opticky rozpoznat cokoli a určit, co to je, když část dat je neúplných nebo neplatných. Tato vlastnost se s velkým úspěchem využívá v medicíně při určování diagnóz (rozpoznávání rakovinových buněk při analýze provedených snímků), při identifikaci letadel z odražených radarových signálů, ale i trhlin v betonu z odražených zvukových vln apod. Uvedené vlastnosti NS lze rovněž využít ve vodním hospodářství při konstrukci povodňových varovných systémů, kde pomáhají odhalit hrozící nebezpečí a zároveň ho kvantifikovat stupněm povodňové aktivity.
- NS nevynikají přesností. Máme-li například sečíst 5,02 a 7,01 získáme pravděpodobně hrubou odpověď, že 12,0. Pokud tedy potřebujeme výpočty s vysokou přesností, neuronové sítě nejsou vhodným prostředkem.
- Pokud je však rychlost výpočtu důležitější než přesnost, NS jsou tím pravým prostředkem pro řešení. Výpočty na natrénovaných sítích probíhají řádově v milisekundách až mikrosekundách. Přitom analýza stavů řízených systémů klasickými optimalizačními nástroji může překročit svým trváním přístupné pozdější, kdy ještě můžeme systém řídit operativně.
- Pokud nelze popsat vztah mezi soubory dat exaktně (ať už je absence exaktního vztahu dána jeho značnou složitostí, nebo prostě tím, že není znám), ale je známe, které veličiny jsou vstupní a mají vliv na veličiny výstupní, mohou NS nalézt požadovaný vztah.
- Výhodou NS rovněž je, že při případné změně procesu, který řídíme nebo analyzujeme, stačí doplnit nové tréninkové vzory do trénovací matice a přetrénovat NS. To je podstatně jednodušší, než sestavení nového výpočetního vzorce,

resp. provedení změn pravidel použitých pro řízení systému a s tím spojeného následného přepisu řídicích programů.

7. Aplikace zjištěných poznatků v ekonomické praxi

Cílem této kapitoly je aplikace zjištěných teoretických poznatků z oblasti genetických algoritmů a neuronových sítí na problémy ekonomické praxe a následné porovnání takto získaných výsledků s výstupy klasických ekonomických metod. Ukázkové programy jsem vytvořil pomocí vývojového prostředí Borland Delphi 7 a kompletní zdrojový kód ke každému příkladu je uveden z úsporných důvodů pouze jako příloha na CD nosiči, které je součástí diplomové práce. Analyzovaná ekonomická data jsou uvedeny v příloze 1 a v příloze 2.

V následujícím textu uvedu tyto čtyři praktické ukázky:

- 1) zjištění parametrů regresní funkce pomocí genetických algoritmů a porovnání výsledků s výstupy regresní analýzy
- 2) predikce ceny grafické karty pomocí vícevrstvé neuronové sítě a porovnání výsledků s výstupy regresní analýzy
- 3) klasifikace podniků do kategorií „*prosperita*“ a „*bankrot*“ na základě analýzy vstupních dat pomocí vícevrstvé neuronové sítě
- 4) využití jednoduché dvouvrstvé neuronové sítě k odhadu časové náročnosti zakázky

V prvních dvou úlohách porovnávám výsledky dosažené metodami umělé inteligence – genetickými algoritmy a neuronovými sítěmi, s výsledky dosažených pomocí klasické statistické metody – regresní analýzy. Zbylé dva příklady slouží jako demonstrace možností metod umělé inteligence při řešení ekonomických problémů.

7.1 Regresní analýza

Metodu regresní analýzy, která byla popsána v kapitole 4.0, jsem aplikoval na data, která byla získána průzkumem trhu s grafickými kartami [I-1]. Cílem bylo odhadnout závislost ceny grafické karty na vybraných vlastnostech (*počet pipelines, rychlost graf. čipu, šířka pamět. sběrnice, atd.*). Ukázka vstupních dat je uvedena na obrázku (7-1).

Název	Počet pipelines	Rychlost graf. čipu [MHz]	Šířka pamětové sběrnice [bit]	Velikost graf. paměti [MB]	Rychlost graf. paměti [MHz]	Cena s DPH [Kč]
Asus A7000/T 64MB	1	150	64	64	300	837
Inno3D GeForce2 MX400 64MB	2	200	64	64	166	885
MicroStar MX4000-T64 64MB	2	250	32	64	400	906
Asus A9200SE/T 64MB	4	200	64	64	333	1060
Sapphire Atlantis ATI Radeon 9200 SE 128MB	4	250	64	128	333	1127
MicroStar MX4000-T128 128MB	2	250	64	128	332	1130
Asus V9400 Magic 128MB	2	250	64	128	332	1147
GigaByte MAYA GV-N52128D 128MB	4	250	64	128	400	1175
Sapphire Atlantis ATI Radeon 9250 64MB	4	240	64	64	400	1186
Club3D ATI 9250 128MB(64bit)	4	240	64	128	400	1220
Sapphire Atlantis ATI Radeon 9250 128MB(64bit)	4	240	64	128	400	1235
Asus A9200SET 128MB	4	200	64	128	333	1241
Asus A9250/TD 128MB	4	240	64	128	400	1290
Asus V9520 Magic 128MB	4	250	64	128	332	1344
Asus V9520-X/TD 128MB	4	250	64	128	400	1344
Sapphire Atlantis ATI Radeon 9250 128MB	4	240	128	128	400	1359

obr. 7-1 Ukázka zdrojových dat

Zdroj: vlastní

Jak je patrné, tak každý řádek vstupních dat obsahuje informace o jedné grafické kartě (o jednom **vzoru**). Celková velikost vstupních dat je 159 vzorů, které byly voleny s ohledem na to, aby rovnoměrně pokrývaly sledovanou oblast. Význam jednotlivých vlastností vstupních vzorů je zřejmě jasný, snad jen první vlastnost „*počet pipelines*“ by mohla působit problémy, a tak ji pro jistotu ve zkratce vysvětlím. Tato vlastnost zjednodušeně udává, kolik „*úkolů*“ je daný procesor (např. GPU) schopen paralelně obstarávat v daný okamžik. Má-li GPU například 4 texturovací pipelines, tak to znamená, že je schopen v jednu chvíli zpracovávat informace o 4 různých texturách.

Závislost mezi vysvětlujícími proměnnými (počet pipelines, rychlost grafického čipu, šířka pamětové sběrnice, velikost grafických pamětí a rychlost grafických pamětí) a vysvětlovanou proměnnou (cena grafické karty) jsem se pokusil vystihnout pomocí regresního modelu, jehož rovnici tvoří parabola $y = \beta_0 + \beta_1 x + \beta_2 x^2$.

Analýzu zkoumaných dat jsem začal elementární metodou popisu závislosti proměnných, tj. **korelační maticí** (matice korelačních koeficientů), kterou lze sestavit pro libovolný počet proměnných a je tudíž pro naše účely velmi vhodná. K sestavení této

korelační matice jsem použil tabulkový procesor Excel (**MS EXCEL: Nástroje – Analýza Dat – Korelace**). Ukázka je uvedena na obrázku (7-2).

V regresním modelu se nesmí vyskytovat silná korelace mezi vysvětlujícími (nezávislými) proměnnými, protože jinak nám použití metody nejmenších čtverců nedává relevantní výsledky – odhady směřodatných chyb regresních parametrů jsou příliš velké, takže intervaly spolehlivosti pro regresní parametry jsou moc široké. V extrémním případě, kdy je mezi vysvětlujícími proměnnými funkční závislost (korelační koeficient je roven 1 nebo -1) nelze parametry modelu pomocí metody nejmenších čtverců odhadnout.

	Cena	Pipelines	Rychlost graf. čipu	Šířka pamět. sběrnice	Velikost graf. paměti	Rychlost graf. paměti
Cena	1					
Pipelines	0,943430183	1				
Rychlost graf. čipu	0,631274858	0,52752198	1			
Šířka pamět. sběrnice	0,861735998	0,86589148	0,4749636	1		
Velikost graf. paměti	0,591654197	0,5581557	0,3450282	0,525260748	1	
Rychlost graf. paměti	0,885468762	0,85122858	0,7890816	0,730332579	0,449036313	1

obr. 7-2 Korelační matice

Zdroj: vlastní

Z uvedené ukázky je patrné, že mezi proměnnými *Pipelines* a *Šířka pamět'ové sběrnice* a mezi *Pipelines* a *Rychlost grafických pamětí* existuje vysoká korelace (čísla označená červeně). Jedná se o případ **multikolinearity**, která je podle některých autorů nezdavá, pokud je korelační koeficient libovolné dvojice proměnných větší než 0,8. Multikolinearita v praxi znamená, že jedna z dvojice vysvětlujících proměnných, které jsou vzájemně silně závislé, je v modelu navíc a měla by být vyřazena. Z tohoto důvodu jsem se rozhodl z modelu vyřadit vysvětlující proměnnou *Pipelines* a poté jsem sestavil novou korelační matici. Ukázka je na obrázku (7-3).

	Cena	Rychlost graf. čipu	Šířka pamět. sběrnice	Velikost graf. paměti	Rychlost graf. paměti
Cena	1				
Rychlost graf. čipu	0,631275	1			
Šířka pamět. sběrnice	0,861736	0,474963648	1		
Velikost graf. paměti	0,591654	0,345028203	0,525260748	1	
Rychlost graf. paměti	0,885469	0,789081557	0,730332579	0,449036313	1

obr. 7-3 Korelační matice bez "pipelines"

Zdroj: vlastní

Po vyřazení proměnné *Pipelines* se již v modelu multikolinearita nevyskytuje. Cena je stále vysoce korelována s proměnnými šířka paměťové sběrnice a rychlost grafických pamětí, ale tyto dvě proměnné již silně korelovány nejsou.

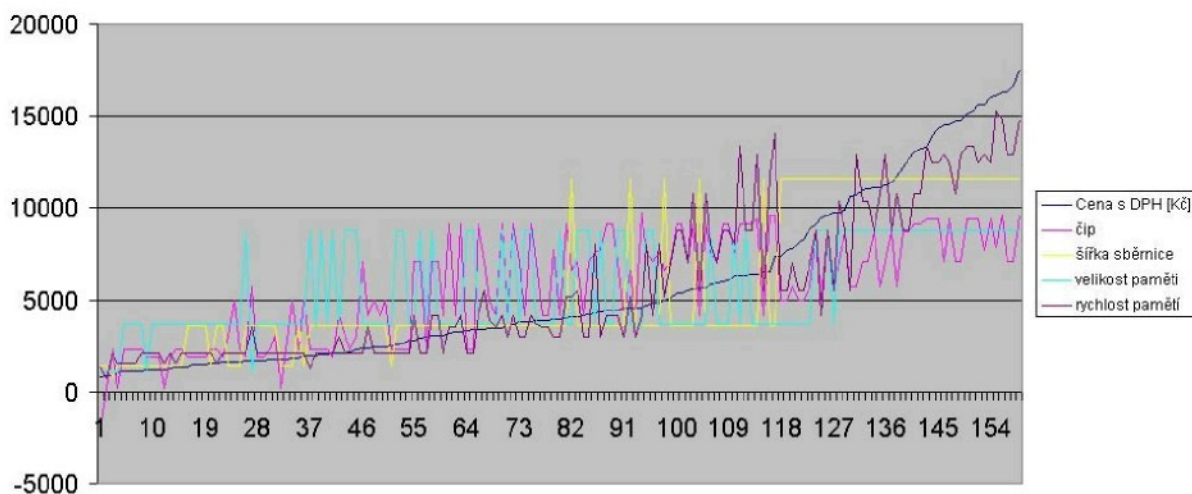
Na takto upravená vstupní data lze aplikovat daný regresní model. Pro přehlednost uvádím jen výsledné hodnoty regresních parametrů, které jsou zobrazeny na obrázku (7-4). Regresní parametry jsou počítány pro každou vysvětlující proměnnou zvlášť. Průběh výpočtu regresního modelu je zobrazen v příloze 3 a též je přiložen v elektronické podobě (soubor *regrese – grafické karty.xls*) na CD nosiči.

regresní parametr	čip	šířka sběrnice	velikost paměti	rychlost pamětí
a	-0,048369285	0,145182782	0,002711973	0,010284215
b	63,57252147	6,369404554	39,44548541	0,043605894
c	-10575,39954	424,1415256	-1447,799374	422,1239626

obr. 7-4 Regresní parametry

Zdroj: vlastní

Pomocí těchto parametrů jsem pro každou vlastnost zvlášť vypočítal odhadovanou velikost ceny grafické karty v závislosti na příslušnou vlastnost. Křivky jednotlivých regresních křivek jsou zobrazeny na následujícím grafu (7-1).



graf 7-1 Regresní křivky

Zdroj: vlastní

Z grafu je patrné, že parabolický regresní model poměrně dobře vystihuje závislost *ceny grafické karty na rychlosti grafických pamětí*. Toto tvrzení je podepřeno i hodnotou determinačního indexu, který pro tuto vlastnost vyšel **0,801332578**, což naznačuje poměrně

silnou závislost těchto zkoumaných dat. Determinační indexy ostatních zkoumaných dat již nevykazují tak silné závislosti. Jen v případě *šířky paměťové sběrnice* se dá hovořit o poměrně slušném vystižení závislosti, což dokládá i hodnota determinačního indexu, který pro tuto vlastnost vyšel **0,757329378**.

Opačný pól tvoří vlastnosti *velikost grafických pamětí* a *rychlost grafického čipu*, jejichž hodnoty determinačních indexů jsou rovny **0,350056443** resp. **0,407781113** a vypovídají o tom, že vystižení závislosti těchto vlastností na ceně graf. karty pomocí parabolické funkce, je jen velmi malé. Takto modelovaná závislost je velmi slabá. Hodnoty všech determinačních indexů jsou uvedeny na následujícím obrázku.

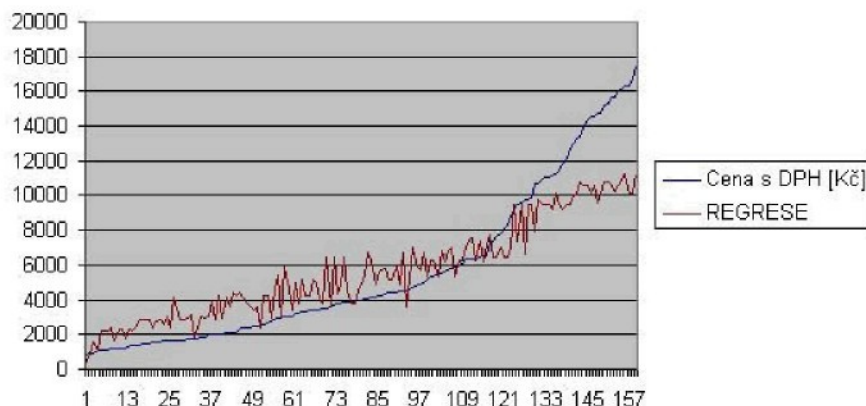
	Rychlost paměti	Velikost paměti	Šířka sběrnice	Rychlost graf. čipu	Průměrný determinační index
Determinační index	0,801332578	0,350056443	0,757329378	0,407781113	0,579124878

obr. 7-5 Hodnoty determinačních indexů

Zdroj: vlastní

Na výše uvedeném obrázku je uvedený *průměrný determinační index*, který jsem získal jako průměr dílčích determinačních indexů. Jeho hodnota **0,579124878** naznačuje, že volba parabolické regresní funkce nebyla pro závislosti mezi těmito daty příliš šťastnou. V praxi bych zajisté přistoupil k volbě jiného regresního modelu, ale já jsem tuto parabolickou regresní funkci zvolil záměrně, abych mohl porovnat výsledky dosažené regresní analýzou s výsledky, jež byly získány pomocí metod umělé inteligence (jmenovitě GA).

Výslednou hodnotu odhadnuté ceny pro každý vzor dostáváme jako průměr jednotlivých regresí, které byly zvlášť vypočítány pro každou vlastnost. Pokud tento postup aplikujeme na všechny vzory a výsledky graficky zobrazíme regresní křivkou, můžeme porovnat průběh takto získané regresní křivky s křivkou zachycující průběh reálných hodnot (hodnot získaných ze zdrojových dat), viz graf (7-2).



graf 7-2 Regrese

Zdroj: vlastní

I tento graf potvrzuje mou domněnku o nepříliš vhodné volbě regresního modelu. Znalost studované problematiky též hovoří proti volbě parabolického regresního modelu, který ze své podstaty má zásadní omezení pro tuto problematiku. Zejména je to fakt, že parabola dosáhne svého maxima (resp. minima) a pak dále klesá (resp. roste), což může mít například paradoxně za následek, že s růstem velikosti paměti by cena grafické karty klesala. Obdobné závěry lze provést i pro ostatní vlastnosti.

7.2 Stanovení parametrů regresní funkce pomocí GA

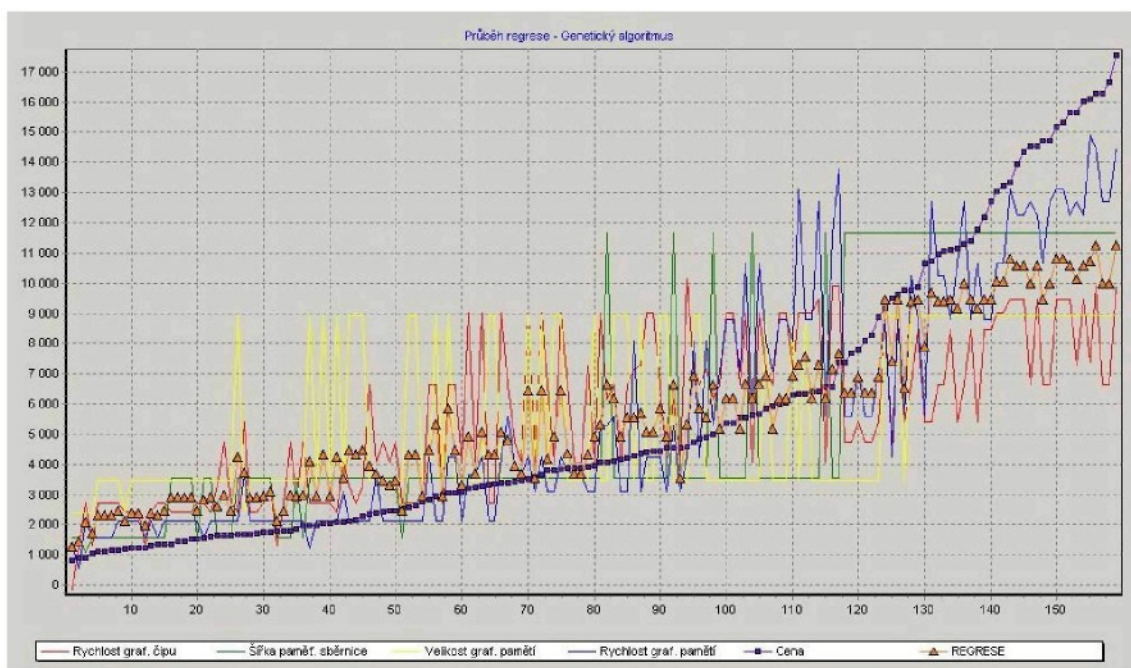
V této úloze jsem se pokusil pomocí metody genetických algoritmů odhadnout regresní koeficienty parabolické regresní funkce uvedené v kapitole 4.0. Mým cílem bylo zhodnotit takto dosažené výsledky a porovnat je s hodnotami dosaženými pomocí regresní analýzy.

Po prostudování literatury a četných experimentech jsem se rozhodl danou problematiku řešit pomocí populace, která čítá 30 jedinců. Takto velká populace se jeví jako optimální vzhledem k povaze zkoumaných dat a jak mi praktické testy ukázaly, tak větší populace nezrychlila ani nezlepšila dosažené výsledky.

Hodnoty ostatních parametrů GA byly nastaveny následovně:

- poměr křížení – 0.80
- poměr mutace – 0.06
- maximální počet tréninkových cyklů – 600000

GA hledá postupně řešení pro každou vlastnost zvlášť. Jak už bylo uvedeno, maximální počet tréninkových cyklů je 600000, ale evoluce parametrů pro danou vlastnost může skončit i dříve, a to za předpokladu, že se vlastnosti nejlepšího jedince nezmění během 1000 po sobě následujících generací. Ke křížení jedinců bylo užito jednobodového křížení, kde bod křížení je vždy volen náhodně. V případě mutace je náhodně změněn jeden parametr jedince a to s ohledem na charakter právě zkoumané vlastnosti (velikost mutace je odvozena od průměrné hodnoty zkoumané vlastnosti – vždy 1 % z průměrné hodnoty). Mutovaný parametr je vždy vybírán náhodně. Výběr nejlepších jedinců je prováděn turnajovou metodou a v rámci zachování nejlepšího řešení je též aplikována metoda elitismu. Průběh evoluce regresních parametrů je zachycen na následujícím grafu.



graf 7-3 Regrese pomocí GA

Zdroj: vlastní

Na první pohled je patrné, že průběh výsledné regresní křivky je velmi podobný křivce získané pomocí regresní analýzy. A i průběhy jednotlivých dílčích regresí je v podstatě stejný jako na obrázku (7-2). Můžeme tedy říci, že použití GA v tomto případě nepřineslo žádné velké zlepšení, ale na druhou stranu je nutné poznamenat, že výsledky jsou ovlivněny volbou parabolické regresní rovnice, která jak už bylo řečeno, není pro tento druh závislostí příliš vhodná.

7.3 Predikce ceny grafické karty pomocí neuronové sítě

Cílem této úlohy je aplikace vícevrstvé neuronové sítě na shodná vstupní data jako v předchozím příkladě a prezentace možností neuronových sítí při analýze vstupních dat, u kterých není předem známa závislost jednotlivých vstupních vlastností na vlastnosti výstupní. V takovýchto úlohách by podle teoretických poznatků měly neuronové sítě vykazovat velmi dobré výsledky a mým úkolem bylo tyto teorie prověřit v praxi. Analyzovaná data jsou shodná jako v předchozí úloze a byla vybrána tak, aby pokrývala rovnoměrně sledovanou oblast.

Při řešení této úlohy jsem se potýkal s problémem stanovení počtu skrytých vrstev neuronů a následně s problémem stanovení vhodného počtu neuronů těchto skrytých vrstev. Postupoval jsem pomocí metody „*pokusů a omylů*“. Začal jsem u nejjednodušších dvouvrstevných sítí a postupným zvyšováním, jak počtu skrytých vrstev, tak počtu neuronů těchto vrstev, jsem došel k závěru, že v tomto případě bude vhodné zvolit neuronovou síť o následující struktuře:

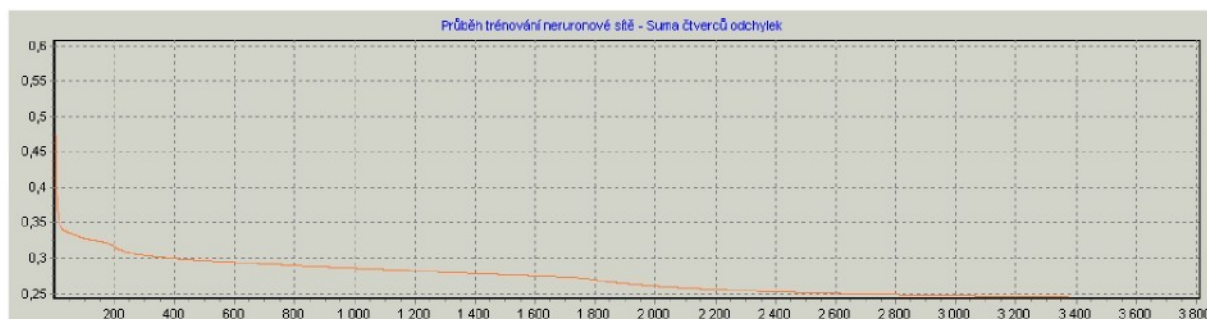
- typ přenosové funkce: **logistická sigmoidální funkce**
- počet neuronů vstupní vrstvy: **5**
- počet skrytých vrstev: **2**
 - počet neuronů – vrstva 1: **6**
 - počet neuronů – vrstva 2: **2**
- počet neuronů výstupní vrstvy: **1**

Ostatní parametry NS byly nastaveny na následující hodnoty:

- velikost tréninkových dat: **159** vzorů
- velikost testovacích dat: **50** vzorů
- koeficient učení: **0.1**
- požadovaná přesnost: **0.1**

Před zahájením trénování NS jsem musel standardizovat vstupní hodnoty na interval (0,1) a to z důvodu, že zvolená přenosová funkce (sigmoida) dává výstupní hodnoty též v tomto intervalu (0,1) a je lépe citlivá na menší vstupní hodnoty. Standardizaci vstupních hodnot jsem provedl pomocí lineární transformace a výstupní hodnoty NS jsem inverzním postupem transformoval zpět na interval skutečných hodnot výstupních dat.

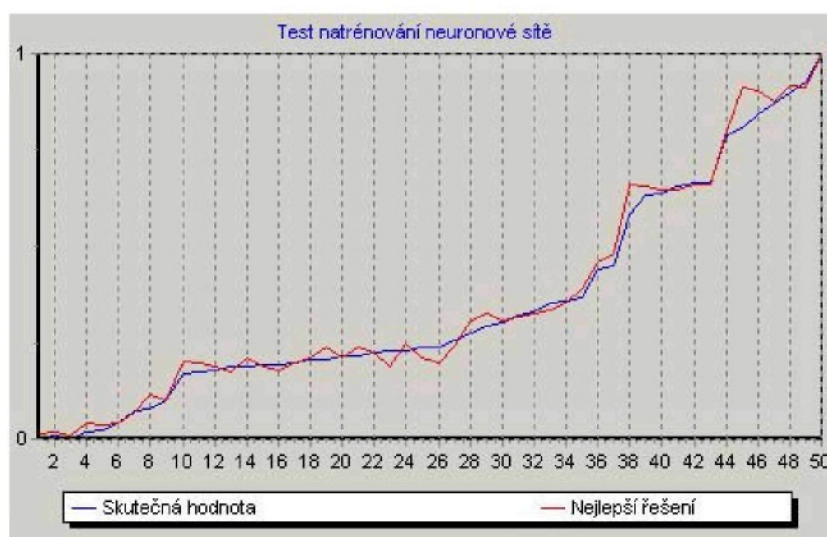
Neuronová síť s těmito parametry vykazovala během fáze učení velmi dobré výsledky a i následně provedený test na cca. 50 testovacích vzorech ukázal, že síť je dostatečně natrénovaná (viz graf 7-4 a 7-5). Oproti tomu síť s menším počtem skrytých vrstev, resp. s menším počtem skrytých neuronů, nedokázaly jednoznačně určit vztahy mezi vstupními veličinami a síť nebylo možné na požadovanou úroveň přesnosti natrénovat. Použití většího počtu skrytých vrstev (resp. neuronů v těchto vrstvách) zase způsobilo, že síť během tréninkové fáze sice vykazovala velmi dobré výsledky, ale ve fázi testovací úplně propadla. Tento jev jsem si vysvětlil tím, že došlo k onomu „zapamatování“ si tréninkových vzorů a síť tudíž nebyla schopna poskytnout relevantní výstupní hodnoty pro nové vstupní (testovací) vzory



graf 7-4 Suma čtverců odchylek - trénink

Zdroj: vlastní

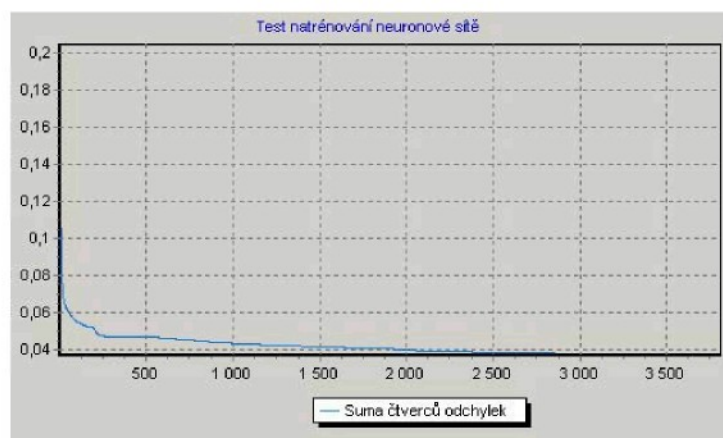
Graf (7-4) zobrazuje vývoj sumy čtverců odchylek během tréninkové fáze. Na první pohled je patrné, že křivka má logaritmický charakter, tzn. že učení (korekce vah mezi neurony) zpočátku přinášelo větší pokles sumy čtverců odchylek a postupem času se vývoj (pokles) odchylek zpomaluje až do bodu, kdy je suma čtverců odchylek téměř konstantní. V tomto okamžiku je vhodné tréninkovou fázi ukončit, neboť další trénink již nepřináší zlepšení. Graf (7-5) zobrazuje situaci po končení trénovací fáze, kde modrá křivka představuje reálné hodnoty cen grafických karet a červená křivka představuje výstupní hodnoty neuronové sítě.



graf 7-5 Situace po ukončení trénovací fáze

Zdroj: vlastní

Během tréninku se vždy po nalezení lepšího řešení, než bylo to stávající, provedl test natrénování NS na skupině testovacích vzorů. Na grafu (7-6) je zachycen průběh sumy čtverců odchylek tréninkových dat.



graf 7-6 Suma čtverců odchylek – test

Zdroj: vlastní

Průběh křivky z grafu (7-6) poměrně dobře koresponduje s průběhem křivky z grafu (7-4). Po vyhodnocení těchto dvou grafů jsem usoudil, že je natrénování neuronové sítě dostatečné a provedl jsem zkouškový odhad ceny modelové grafické karty (**MicroStar NX6600GT-VTD**). Skutečná cena této grafické karty byla 6006 Kč. Na obrázku (7-6) je zobrazen výsledek testu.

Dotaz		Cena
Pipelines:	8	6003.944974
Rychlost čipu (MHz):	500	
Šířka sběrnice (bit):	128	
Velikost paměti (MB):	128	
Rychlost paměti (MHz):	900	
		Dotaz

obr. 7-6 Výsledek dotazu

Zdroj: vlastní

I tento test mi potvrdil, že NS se výborně hodí na úlohy tohoto typu. Jsou-li vstupní (trénovací) data vhodně zvolena a je-li optimálně zvolena i topologie NS, dávají neuronové sítě v podobných úlohách skvělé výsledky. Pro jejich použití v praxi hovoří i to, že v případě jakékoliv změny ve sledované oblasti stačí tréninkovou množinu doplnit o nové vzory, poté přetrénovat a NS je opět připravena.

7.4 Klasifikace podniků do kategorií „prosperita“ a „bankrot“

V literatuře se v souvislosti s využitím neuronových sítí často hovoří o klasifikaci. Schopnosti NS jsou v této oblasti doslova vyzdvihovány do nebe a tak jsem se rozhodl i tuto schopnost NS prověřit.

Jako vstupní data jsem použil informace o podnicích (viz. příloha 2), kde vstupními proměnnými byly 4 následující proměnné:

1. Úvěr / aktiva
2. Tržby / aktiva
3. Zisk / celková aktiva
4. Dividendy

a výstupní proměnnou bylo zařazení podniku do jedné z kategorií „prosperita“ nebo „bankrot“.

Pro realizaci této úlohy jsem zvolil NS s následujícími parametry:

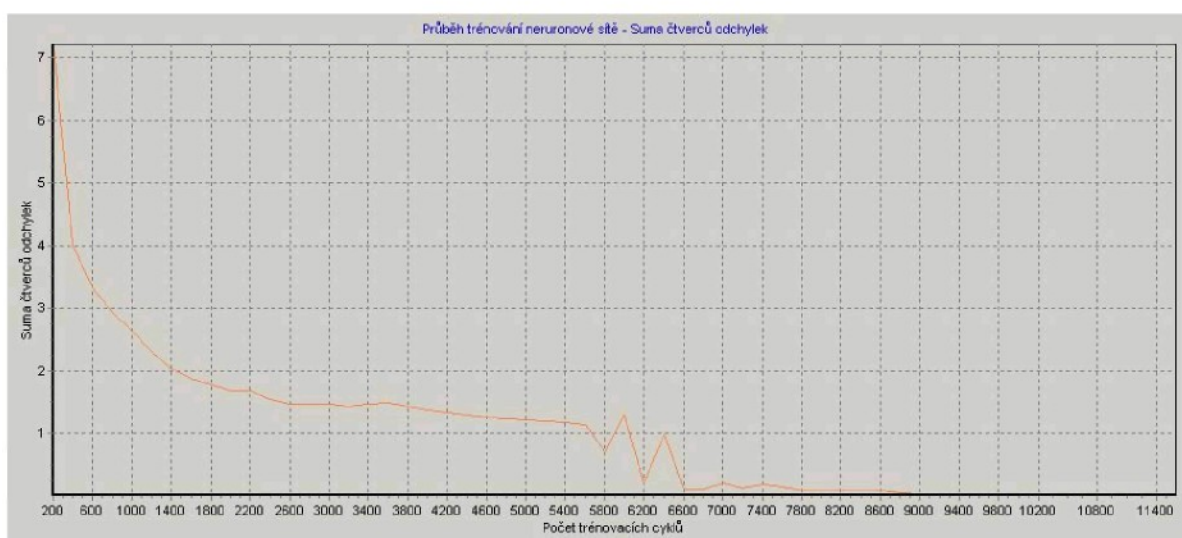
- typ přenosové funkce: **logistická sigmoidální funkce**
- počet neuronů vstupní vrstvy: **4**
- počet skrytých vrstev: **2**
 - počet neuronů – vrstva 1: **13**
 - počet neuronů – vrstva 2: **10**
- počet neuronů výstupní vrstvy: **1**

Ostatní parametry NS byly nastaveny na následující hodnoty:

- velikost tréninkových dat: **70** vzorů
- velikost testovacích dat: **30** vzorů
- koeficient učení: **0.1**
- požadovaná přesnost: **0.07**

Před zahájením trénování NS jsem opět provedl lineární transformaci vstupních proměnných na interval (0,1) a výstupem NS byla hodnota buď 0, což znamená zařazení podniku do kategorie „bankrot“ a nebo hodnota 1, která značí podnik z kategorie „prosperita“.

Poté jsem zahájil trénink NS a následující graf (7-7) zobrazuje křivku, jejíž průběh zachycuje vývoj sumy čtverců odchylek. Opět má tato křivka logaritmický tvar jako v předchozím příkladě.



graf 7-7 Suma čtverců odchylek - trénování NS

Zdroj: vlastní

Během tréninku NS je v pravidelných intervalech prováděn test, který má ověřit stupeň natrénování NS (viz obr. 7-7). Na vstupní neurony NS jsou přikládány vzory z testovací množiny a analyzuje se jejich správné zařazení do příslušných kategorií. Trénink NS má v tomto případě dvě ukončovací podmínky:

1. natrénování NS s danou přesností pro 100% trénovacích vzorů
2. 100% úspěšnost v kategorizaci testovacích vzorů

Výsledky klasifikace				
	Trénink	Trénink	Test	Test
	Prosperita	Bankrot	Prosperita	Bankrot
Počet vzorů	35	35	10	20
Správně	35	35	10	20
Chybně	0	0	0	0

obr. 7-7 Stupeň natrénování

Zdroj: vlastní

Po ukončení tréninkové fáze jsem provedl praktický test, kdy jsem na vstupy sítě vložil hodnoty, které nebyly v trénovací ani testovací množině vzorů (údaje patří prosperujícímu podniku). Výsledek je zobrazen na obrázku (7-8).

Zadej dotaz	
Úvěr / aktiva:	0.170687625
Tržby / aktiva:	2.727585053
Zisk / celková aktiva:	0.179784142
Dividendy:	900
Výsledek:	prosperita
	Dotaz

obr. 7-8 Výsledek dotazu

Zdroj: vlastní

Vstupní hodnoty patřily „prosperujícímu“ podniku. A jak výše uvedená ukázka naznačuje, tak i v tomto typu úlohy neuronová síť obstála s velmi dobrými výsledky. A znovu se potvrdila univerzálnost použití NS, neboť v tomto příkladě byla použita jen lehce modifikovaná NS z předchozího příkladu a to i přesto, že tyto dvě úlohy si nejsou povahou dat a vztahů mezi vstupními proměnnými vůbec podobné. Snadná modifikace a univerzálnost použití NS je jejich obrovskou výhodou.

Klasifikační možnosti NS mají širokou škálu praktického využití. Mohou být například použity při ohodnocení bonity zákazníka či dodavatele nebo při hodnocení rizik spojených například s uzavíráním obchodních smluv.

7.5 Využití jednoduché dvouvrstvé neuronové sítě

V posledním příkladě jsem se zaměřil na použití jednoduchých dvouvrstvých sítí a na jejich možnosti využití v praxi. Jako modelovou situaci jsem si zvolil výrobní podnik, který se zabývá výrobou a návrhem kytarových zesilovačů a kytarových komb². Mezi hlavní činnosti podniku patří následující činnosti:

- výroba kytarových zesilovačů
- výroba kytarových komb
- výroba reproboxů
- vývoj zesilovačů
- vývoj reproboxů

Dále firma zaměstnává pracovníky následujících profesí:

- truhlář
- čalouník
- zámečník
- elektrotechnik
- montážní technik
- vývojář
- kontrolní pracovník

Tato firma má značné problémy s plánováním svých zakázek, kdy velmi často dochází k situacím, kdy jsou některé profese v rámci jedné zakázky buď přetíženy nebo naopak nedostatečně využity. Proto se v tomto příkladě pokusím pomocí NS navrhnout jednoduchý systém, který na základě vstupních podílů jednotlivých činností v rámci jedné zakázky poskytne časový rozpis jednotlivých profesí.

V tomto příkladě použiji NS s 5 neurony ve vstupní vrstvě a se 7 ve výstupní vrstvě. K natrénování této sítě jsem použil pouze 5 trénovacích vzorů, kde vstupní hodnoty tvořily podíly jednotlivých činností v rámci jedné zakázky (součet podílů jednotlivých činností musí dávat 1) a výstupní hodnoty tvořily časové náročnosti daných profesí.

² Jedná se o spojení kytarového zesilovače a reproduktoru do jednoho kompaktního celku

Po natrénování byla NS schopna dávat relevantní výsledky i pro situace, na které nebyla natrénována (viz obr. 7-9).

obr. 7-9 Výsledek "časové kalkulačky"

Zdroj: vlastní

Podnik se sedmi zaměstnanci patrně nebude potřebovat podobný plánovací systém, ale jistě si dokážete představit situaci, kdy podnik zaměstnává několik desítek a možná i stovek zaměstnanců a tam už by se nasazení takového systému mohlo jevit jako opodstatněné.

Tento druh NS má i přes svoji značnou jednoduchost značné možnosti praktického využití. Například by se mohlo jednat o vyhodnocování výkonnosti jednotlivých členů pracovního týmu v situaci, kdy jsme schopni měřit pouze výkon celého týmu. Tam nám může NS velmi pomoci, stačí jen při trénování použít vhodná data a NS už sama pozná, kdo je „tahounem týmu“ a kdo se pouze „veze“. Další možností je třeba nasazení v oblasti skladového hospodářství, kde by se pomocí NS dala velmi dobře řídit například zásoba materiálů a podniky by předešly situacím, kdy jim materiál buď chybí a nebo zbytečně leží na skladě. A podobných úloh najdeme v ekonomické praxi spousty.

8. Závěr

Problematika využití metod umělé inteligence v ekonomii je stále ještě předmětem zkoumání, ale i přes tento fakt se čím dál častěji setkáváme s nejrůznějšími aplikacemi těchto metod na problémy ekonomické reality a do budoucna se tento trend ještě umocní.

Cílem mé práce bylo prověřit možnosti vybraných uměle-inteligentních metod. Podařilo se mi zpracovat dvě základní teorie:

1. genetické algoritmy
2. neuronové sítě

Teoretické poznatky a předpoklady z kapitol 4.0 až 6.0 jsem prověřil ve čtyřech praktických příkladech.

V prvním příkladě jsem se zaměřil na problém stanovení ceny grafické karty v závislosti na hodnotách vybraných vstupních parametrů. Porovnal jsem výsledky získané pomocí klasické statistické metody analýzy dat – regresní analýzy s výsledky dosažených pomocí genetického algoritmu. GA jsem v úloze použil k odhadu koeficientů regresní funkce. V domněnce, že GA by mohl dosáhnout lepších výsledků než regresní analýza, jsem záměrně zvolil parabolickou regresní funkci, která jen velmi málo vystihuje závislosti těchto vstupních dat. Ovšem po prostudování výsledných grafů musím konstatovat, že použití GA v tomto případě přináší zhruba srovnatelné výsledky s regresní analýzou. I přes mírně lepší výsledky GA nemohu říci, že by jejich použití bylo významným přínosem.

V druhém případě jsem použil vícevrstvou neuronovou síť k analýze stejných vstupních dat jako v předcházejícím příkladě. Po natrénování neuronové sítě jsem provedl test na 50 testovacích vzorech. Ve srovnání s výsledky dosažených pomocí regresní analýzy a GA neuronová síť dosahovala výrazně lepších výstupních hodnot než obě zmiňované metody. Proto se domnívám, že za předpokladu dostupnosti dostatečného množství relevantních tréninkových dat, je použití NS v podobných případech velkým přínosem.

Třetí příklad se věnoval prověření klasifikačních schopností neuronových sítí. Sít' byla natrénována na virtuálním souboru vybraných informací o podnicích (cca. 70 tréninkových vzorů), na jejichž základě by se sít' měla naučit správně kategorizovat podniky do dvou skupin – prosperita a bankrot. Poté jsem provedl test (cca. 30 testovacích vzorů) a porovnal dosažené výsledky. I v této úloze se prokázala obrovská síla NS, které jsou po správném natrénování v podobných úlohách doslova nepřekonatelné.

V poslední úloze jsem zkoumal možnosti využití nejjednodušších dvouvrstvých neuronových sítí. Tuto NS jsem využil při řešení jednoduché modelové situace, kde má NS podniku pomoci při optimálním časovém využití výrobních zdrojů. NS byla natrénována jen na velmi malém počtu vstupních vzorů (cca. 5 vzorů), ale i přesto dokázala poskytovat relevantní výstupy. Využití dvouvrstvých sítí má široké možnosti, pro názornost uveďme třeba optimální plánování skladovacích zásob nebo tvorba pracovních týmů.

Jak naznačují provedené studie využití NS přináší velmi dobré výsledky, ale na druhou stranu bychom neměli propadat přehnanému optimismu. Výsledky neuronových sítí jsou velmi těsně spjaté s kvalitou vstupních – tréninkových dat. Jedině dobře zvolená vstupní data nám zaručí kvalitní výsledky. Proto předesílám, že by výběr vstupních dat měli provádět jak kvalifikovaní lidé, kteří dokážou rozhodnout, která data jsou pro následnou analýzu důležitá, tak i lidé, jichž specializace se sledovanou oblastí přímo nespojuje. Jen tak se můžeme vyvarovat jednostrannému pohledu na danou problematiku.

Metody umělé inteligence mají v ekonomické oblasti velké možnosti využití, ať už se jedná o zmiňované NS a GA nebo o fuzzy množina atd. Cílem podnikatelských aktivit je pracovat lépe a tedy více vydělat. To mnohdy znamená lépe rozhodovat a právě v této oblasti by nám metody umělé inteligence mohly výrazně pomoci. Na závěr bych rád uvedl následující citát [18]:

“Databáze a tabulkové kalkulátory zpřístupnily informace, nutné pro rozhodování. Metody umělé inteligence jdou ještě mnohem dále tím, že (částečně nebo úplně) přebírají rozhodování samotné.”

9. Seznam literatury

- [01] Rosenblatt, F. The perceptron: a probabilistic model for information storage and organization in the brain. In: Psychological Review 65, 1958.
- [02] Novák, M. Neuronové sítě a neuropočítače. Praha: Senzo a.s., 1992.
- [03] Lawrence, J. Introduction to Neural Network. Scientific Software, 6th Edition. Nevada City, 1994.
- [04] Nacházel, K., Starý, M. a Zezulák, J. Využití metod umělé inteligence ve vodním hospodářství. Praha: Academia, 2004. ISBN 80-200-0229-4
- [05] Kosko, B. Fuzzy systems as Universal Approximators. San Diego, 1992.
- [06] Kůrková, V. An Incremental Architecture Algorithm for Feedforward Neural Nets. IEEE Workshop: Computer-Intensive Methods in Control and Signal Processing, 1998.
- [07] Mařík, V., Štěpánková, O. a Lažanský, J. Umělá inteligence 4. Praha: Academia, 2003. ISBN 80-200-1044-0
- [08] Fošumpaur, P. Použití umělých neuronových sítí ve vodním hospodářství. [doktorská disertace], Praha: ČVUT, 1998.
- [09] Fanta, J. Psychologie, algoritmy a umělá inteligence na kapitálových trzích. Praha: Grada, 2001.
- [10] Mařík, V., Štěpánková, O. a Lažanský, J. Umělá inteligence 1. Praha: Academia, 1993. ISBN 80-200-0496-3
- [11] White, M., Flockton, S. Modelling the Behaviour of the Genetic Algorithm. GALESIA, 1995.
- [12] Blickle, T., Thiele, L. Comparison of Selection Schemes Used in Genetic Algorithms. TIK/Report, 1995.
- [13] Goldberg, D. Genetic Algorithms in Search, Optimization and Machine Learning. Addison Wesley, 1989.
- [14] Goldberg, D., Deb, K. A Comparative Analysis of Selection Schemes Used in Genetic Algorithms. Morgan Kaufmann Publishers, 1991.

- [15] Vavak, F., Fogarty, T. C. a Jukes, K. Algorithm for Load Balancing in the Progress Industry. Mendel, 1995.
- [16] Seger, J., Hindls, R. Statistické metody v ekonomii. Praha: Nakladatelství H&H, 1993. ISBN 80-85787-26-1
- [17] Hindls, R., Kaňoková, J. a Novák, I. Metody statistické analýzy pro ekonomy. Praha: Management Press, 1997. ISBN 80-85943-44-1
- [18] O'Leary, D. E., Kingston, J. Artificial Intelligence in Business II: Development, Integration and Organizational Issues. The Knowledge Engineering Review, 1994.
- [19] Mařík, V., Štěpánková, O. a Lažanský, J. Umělá inteligence 2. Praha: Academia, 1997. ISBN 80-200-0504-8
- [20] Zelinka, I. Umělá inteligence. Hrozba nebo naděje? Praha: BEN – technická literatura, 2003. ISBN 80-7300-068-7
- [21] Mařík, V., Štěpánková, O. a Lažanský, J. Umělá inteligence 3. Praha: Academia, 2001. ISBN 80-200-0472-6
- [22] Pokorný, M. Umělá inteligence v modelování a řízení. Praha: BEN–technická literatura, 1996. ISBN 80-901984-4-9
- [23] Zelinka, I. Umělá inteligence v problémech globální optimalizace. Praha: BEN–technická literatura, 2002. ISBN 80-7300-069-5

9.1 Další zdroje

[I-01] Czech Computer [online].[cit.17. 6.2005].

Dostupné z: <http://www.czechcomputer.cz>

[I-02] Umělá inteligence [online].[cit. 13.11.2005].

Dostupné z: <http://programujte.com/view.php?cislocclanku=2005052304>

[I-03] Otázky umělé inteligence a kognitivních věd [online].[cit. 20.12.2005].

Dostupné z: <http://hyperkrychle.cz/ai2004.pdf>

[I-04] Umělá inteligence [online].[cit. 4.12.2005].

Dostupné z: <http://www.kyberpunk.org/modules.php?name=News&file=article&sid=24>

10. Seznam příloh

Příloha č. 1: Vstupní data – grafické karty	5
Příloha č. 2: Vstupní data – „ <i>prosperita a bankrot</i> “	3
Příloha č. 3: Regresní analýza – grafické karty	1
Příloha č. 4: Zdrojové kódy ukázkových programů	1

Příloha č. 1: Vstupní data – grafické karty

Název	Cena s DPH [Kč]	Rychlost graf. čipu [MHz]	Šířka pamět. sběrnice [bit]	Velikost graf. paměti [MB]	Rychlost graf. paměti [MHz]	Počet pipelines
<u>Asus A7000/T 64MB</u>	837	150	64	64	300	1
<u>Inno3D GeForce2 MX400 64MB</u>	885	200	64	64	166	2
<u>MicroStar MX4000-T64 64MB</u>	906	250	32	64	400	2
<u>Asus A9200SE/T 64MB</u>	1060	200	64	64	333	4
<u>Sapphire Atlantis ATI Radeon 9200 SE 128MB</u>	1127	250	64	128	333	4
<u>MicroStar MX4000-T128 128MB</u>	1130	250	64	128	332	2
<u>Asus V9400 Magic 128MB</u>	1147	250	64	128	332	2
<u>GigaByte MAYA GV-N52128D 128MB</u>	1175	250	64	128	400	4
<u>Sapphire Atlantis ATI Radeon 9250 64MB</u>	1186	240	64	64	400	4
<u>Club3D ATI 9250 128MB(64bit)</u>	1220	240	64	128	400	4
<u>Sapphire Atlantis ATI Radeon 9250 128MB(64bit)</u>	1235	240	64	128	400	4
<u>Asus A9200SET 128MB</u>	1241	200	64	128	333	4
<u>Asus A9250/TD 128MB</u>	1290	240	64	128	400	4
<u>Asus V9520 Magic 128MB</u>	1344	250	64	128	332	4
<u>Asus V9520-X/TD 128MB</u>	1344	250	64	128	400	4
<u>Sapphire Atlantis ATI Radeon 9250 128MB</u>	1359	240	128	128	400	4
<u>GigaByte MAYA GV-R925128D 128MB</u>	1457	240	128	128	400	4
<u>Club3D ATI 9250 128MB(128bit)</u>	1458	240	128	128	400	4
<u>Sapphire Atlantis ATI Radeon 9250 128MB VIVO</u>	1517	240	128	128	400	4
<u>GigaByte MAYA GV-R955128T 128MB OEM</u>	1526	250	64	128	400	4
<u>Inno3D GeForce FX 5200 128MB</u>	1553	250	128	128	333	4
<u>Asus A9250GE/TD 128MB</u>	1605	240	128	128	400	4
<u>Gainward FX PowerPack Pro/665 TV/DVI 128MB</u>	1636	270	64	128	400	4
<u>MicroStar RX300SE-TD128E 128MB, PCI-E</u>	1642	325	64	128	400	4
<u>MicroStar FX5200-TD128 128MB</u>	1659	250	64	128	400	4
<u>Sapphire Atlantis ATI Radeon 9250 256MB</u>	1664	240	128	256	400	4
<u>MicroStar NX6200TC-TD64E 64MB, PCI-E</u>	1668	350	128	64	550	4
<u>GigaByte MAYA GV-R925128VH 128MB VIVO OEM</u>	1670	240	128	128	400	4
<u>Club3D ATI 9250 128MB(128bit) VIVO</u>	1702	240	128	128	400	4
<u>Asus V9520/TD 128MB</u>	1735	250	128	128	400	4
<u>Inno3D GeForce FX-5500 128MB</u>	1753	270	128	128	400	4
<u>MicroStar FX5500-T128 128MB</u>	1783	200	64	128	400	4
<u>Asus A9550/TD 128MB</u>	1795	250	64	128	400	4

<u>Sapphire Atlantis ATI Radeon 9600SE 128MB</u>	1808	325	64	128	400	4
<u>Sapphire Atlantis ATI Radeon 9550 128MB</u>	1846	250	128	128	400	4
<u>MicroStar RX300LE-TD128E</u>	1984	325	64	128	400	4
<u>Club3D FX5200 256MB</u>	1990	250	128	256	286	4
<u>Asus A9550GE/TD 128MB</u>	1994	250	128	125	400	4
<u>Club3D ATI 9550 256MB(128bit)</u>	2058	250	128	256	400	4
<u>Sapphire Atlantis ATI Radeon 9550 128MB VIVO</u>	2067	250	128	128	400	4
<u>Asus A9250GE/TD 256MB</u>	2094	240	128	256	400	4
<u>GigaByte MAYA GV-N55128DP 128MB</u>	2104	300	128	128	500	4
<u>Inno3D GeForce FX-5500 256MB</u>	2119	270	128	256	400	4
<u>Sapphire Atlantis ATI Radeon 9550 256MB</u>	2157	250	128	256	400	4
<u>Inno3D GeForce FX-5500 256MB, VIVO</u>	2286	270	128	256	400	4
<u>Club3D ATI 9600 128MB</u>	2353	400	128	128	400	4
<u>MicroStar NX6200-TD128E 128MB, PCI-E</u>	2393	300	128	128	550	4
<u>Sapphire Atlantis ATI Radeon 9600 128MB</u>	2418	325	128	128	400	4
<u>MicroStar NX6200-TD128 128MB</u>	2444	300	128	128	400	4
<u>Asus A9600GE/TD 128MB</u>	2453	325	128	128	400	4
<u>Asus V9570LE/TD 128MB</u>	2528	250	64	128	400	4
<u>Asus A9550GE/TD 256MB</u>	2582	250	128	256	400	4
<u>Abit Radeon 9550-256CDT 256MB</u>	2636	250	128	256	400	4
<u>Asus V9570GE/TD 128MB</u>	2756	250	128	128	400	4
<u>Sapphire Atlantis ATI Radeon X600Pro 128MB, PCI-E</u>	2818	400	128	128	600	4
<u>Club3D ATI 9600 256MB</u>	2948	400	128	256	400	4
<u>Asus V9520 VideoSuite 128MB</u>	2983	250	128	128	400	4
<u>Sapphire Atlantis ATI Radeon X600Pro 256MB, PCI-E</u>	3040	400	128	256	600	4
<u>Club3D ATI 9600PRO 128MB</u>	3074	400	128	128	600	4
<u>Inno3D GeForce 6200 128MB</u>	3084	300	128	128	400	4
<u>MicroStar RX600XT-TD128E II 128MB, PCI-E</u>	3214	500	128	128	550	4
<u>Asus N6200GE/TD 128MB</u>	3236	300	128	128	550	4
<u>Sapphire Atlantis ATI Radeon 9600XT 128MB</u>	3286	500	128	128	600	4
<u>MicroStar FX5700LE-TD256 256MB</u>	3311	250	128	256	400	4
<u>Asus V9570GE/TD 256MB</u>	3371	250	128	256	400	4
<u>Sapphire Atlantis ATI Radeon 9600XT 128MB VIVO</u>	3402	500	128	128	600	4
<u>Sapphire Atlantis ATI Radeon X700 128MB, PCI-E</u>	3408	400	128	128	700	8
<u>Sapphire Atlantis ATI Radeon 9800SE 128MB</u>	3439	325	128	128	580	4
<u>Inno3D GeForce 6600 128MB, AGP</u>	3482	300	128	128	550	8
<u>Sapphire Atlantis ATI Radeon 9600XT 256MB</u>	3502	500	128	256	600	4

<u>Gainward FX PowerPack Ultra/1760 XP TV</u>	3589	300	128	128	500	8
<u>Sapphire Atlantis ATI Radeon 9600XT 256MB VIVO</u>	3669	500	128	256	600	4
<u>Sapphire Atlantis ATI Radeon X700 128MB</u>	3785	400	128	128	500	8
<u>Inno3D GeForce 6600 256MB, AGP</u>	3814	300	128	256	500	8
<u>Club3D ATI 9600XT 256MB</u>	3835	500	128	256	600	4
<u>Gainward FX PowerPack Ultra/1760 XP Golden Sample</u>	3860	400	128	128	560	8
<u>MicroStar NX6600-TD128E 128MB, PCI-E</u>	3885	300	128	128	550	8
<u>Asus N6600/TD 128MB</u>	3889	300	128	128	550	8
<u>Asus V9570 TD 128MB</u>	3922	425	128	128	500	4
<u>Gainward FX PowerPack Ultra/1780 TV/DVI 256MB</u>	3965	300	128	256	500	8
<u>Club3D ATI 9600XT 128MB</u>	4046	500	128	128	680	4
<u>Sapphire Atlantis ATI Radeon 9800Pro 128MB</u>	4081	380	256	128	680	8
<u>Sapphire Atlantis ATI Radeon X700 256MB, PCI-E</u>	4106	400	128	256	700	8
<u>MicroStar NX6600-TD256E 256MB, PCI-E</u>	4186	300	128	256	500	8
<u>Sapphire Atlantis ATI Radeon X700 256MB</u>	4202	400	128	256	500	8
<u>MicroStar RX700 PRO-TD128E 128MB, PCI-E</u>	4276	420	128	128	860	8
<u>Asus V9570/TD 256MB</u>	4373	425	128	256	500	4
<u>Asus A9600XT 128MB</u>	4410	500	128	128	600	4
<u>GigaByte MAYA GV-R96X128D 128MB</u>	4424	500	128	128	600	4
<u>Sapphire Atlantis ATI Radeon X700 Fireblade 256MB</u>	4428	400	128	256	600	8
<u>Asus N6600/TD 256MB</u>	4542	300	128	256	500	8
<u>Club3D All-In-Wonder 9800SE 128MB</u>	4547	380	256	128	680	4
<u>MicroStar NX6600-VTD256EH 256MB, VIVO, PCI-E</u>	4555	300	128	128	500	8
<u>GigaByte MAYA GV-R96X128DU 128MB</u>	4585	550	128	125	600	4
<u>Sapphire Atlantis ATI Radeon X700PRO 256MB, PCI-E</u>	4720	420	128	256	860	8
<u>MicroStar NX6600-VTD256 256MB</u>	4855	400	128	256	600	8
<u>Sapphire Atlantis ATI Radeon X700PRO 128MB</u>	4869	420	128	128	860	8
<u>Club3D ATI 9800Pro 256bit 128MB</u>	4987	380	256	128	680	8
<u>MicroStar NX6600-VTD128E Diamond 128MB, VIVO</u>	5130	400	128	128	800	8
<u>Inno3D GeForce 6600GT 128MB, AGP</u>	5378	500	128	128	900	8
<u>Gainward FX PowerPack Ultra/1960 TV-DVI 128MB</u>	5380	500	128	128	900	8
<u>Inno3D GeForce 6600PE 128MB, AGP</u>	5544	400	128	128	800	8
<u>GigaByte MAYA GV-N66T128D 128MB</u>	5555	500	128	128	1000	8
<u>Asus V9999LE/TD 128MB</u>	5627	300	256	128	700	8
<u>MicroStar NX6600GT-TD128E 128MB, PCI-E</u>	5669	500	128	128	1000	8
<u>Sapphire Atlantis ATI Radeon X700PRO 256MB</u>	5837	420	128	256	860	8
<u>MicroStar NX6600-VTD128 Diamond</u>	5915	400	128	128	800	8

<u>MicroStar NX6600GT-VTD128 128MB</u>	6006	500	128	128	900	8
<u>Asus N6600GT/TD 128MB</u>	6044	500	128	128	900	8
<u>MicroStar RX700 PRO-TD256E 256MB, PCI-E</u>	6315	420	128	256	860	8
<u>GigaByte MAYA GV-N66T128VP 128MB</u>	6334	500	128	128	1120	8
<u>Gainward FX PowerPack Ultra/1960 TV-XP 256MB</u>	6349	500	128	256	900	8
<u>Gainward FX PowerPack Ultra/1960 XP 128MB, AGP</u>	6401	500	128	128	900	8
<u>Asus N6600GT TOP 128MB</u>	6416	520	128	128	1100	8
<u>MicroStar NX6800LE-TD128 128MB</u>	6545	300	256	128	700	8
<u>Gainward FX PowerPack Ultra/1960 XP</u>	6558	540	128	128	1050	8
<u>Gainward FX PowerPack Ultra/1960 XP GLH,</u>	7396	540	128	128	1150	8
<u>Gainward FX PowerPack Ultra/2100 TV/DVI 128MB</u>	7396	325	256	128	700	12
<u>MicroStar NX6800-TD128 128MB</u>	7684	325	256	128	700	12
<u>Inno3D GeForce 6800PE 128MB</u>	7805	350	256	128	800	12
<u>Leadtek Winfast A400 TDH 128MB</u>	8098	325	256	128	700	12
<u>Asus V9999/TD 128MB</u>	8291	325	256	128	700	12
<u>Gainward FX PowerPack Ultra/2100 TV/DVI</u>	8888	350	256	128	800	12
<u>Sapphire Atlantis ATI Radeon X800Pro 256MB</u>	9162	475	256	256	900	12
<u>MicroStar NX6800-TD256E 256MB, PCI-E</u>	9506	325	256	256	600	12
<u>Sapphire Atlantis ATI Radeon X800 Pro 256 MB, PCI-E</u>	9603	475	256	256	900	12
<u>Asus V9999GT/TD 128MB</u>	9764	350	256	128	700	16
<u>Sapphire Atlantis ATI Radeon X800 XL 256 MB, PCI-E</u>	9773	400	256	256	980	16
<u>Excalibur X800Pro IceQ II 256MB</u>	9872	475	256	256	900	12
<u>Asus V9999GT/TD 256MB</u>	10670	350	256	256	700	16
<u>Inno3D GeForce 6800GT 256MB</u>	10734	350	256	256	1100	16
<u>Sapphire Atlantis ATI Radeon X800 XL Ultimate</u>	10974	400	256	256	980	16
<u>Asus AX800XL/2DTV 256MB</u>	11071	400	256	256	980	16
<u>MicroStar RX800 PRO VTD 256MB VIVO</u>	11105	475	256	256	900	12
<u>MicroStar NX6800GT-TD256 256MB</u>	11127	350	256	256	1000	16
<u>Gainward FX PowerPack Ultra/2400 TV/DVI</u>	11284	400	256	256	1100	16
<u>Club3D ATIX800 Pro 256MB VIVO</u>	11401	475	256	256	900	12
<u>Gainward FX PowerPack Ultra/2400 TV/DVI</u>	11768	350	256	256	1000	16
<u>Club3D ATIX800 Pro 256MB</u>	12198	475	256	256	900	12
<u>Asus AX800 PRO/TD 256MB</u>	12697	475	256	256	900	12
<u>Sapphire Atlantis ATI Radeon X800XT</u>	13031	500	256	256	1000	16
<u>Club3D ATIX800 XT 256MB VIVO</u>	13245	500	256	256	1000	16
<u>Sapphire Atlantis ATI Radeon X800XT Platinum</u>	13340	520	256	256	1120	16
<u>Sapphire Atlantis ATI Radeon X850 XT 256 MB</u>	13923	520	256	256	1080	16

<u>Asus AX850PRO/HTVD 256MB, VIVO</u>	14340	520	256	256	1080	12
<u>Gainward FX PowerPack Ultra/2600 TV-DVI 256MB</u>	14530	400	256	256	1100	16
<u>Sapphire Atlantis ATI Radeon X850 XT 256 MB, VIVO</u>	14542	520	256	256	1080	16
<u>Inno3D GeForce 6800 Ultra 256MB</u>	14720	400	256	256	1000	16
<u>MicroStar NX6800Ultra-T2D256 256MB</u>	14732	400	256	256	1100	16
<u>Asus AX800XT 256MB</u>	15184	520	256	256	1120	16
<u>Asus AX800XT/TV D 256MB, VIVO</u>	15303	520	256	256	1120	16
<u>Asus AX850XT/HTVD 256MB, VIVO</u>	15637	520	256	256	1080	16
<u>Asus V9999 Ultra/2DT 256MB</u>	15637	425	256	256	1100	16
<u>Sapphire Atlantis ATI Radeon X850 XT 256 MB, VIVO</u>	16041	520	256	256	1080	16
<u>Gainward FX PowerPack Ultra/2600 Golden Sample</u>	16101	430	256	256	1200	16
<u>Asus AX850XTP/HTVD 256MB, VIVO</u>	16291	540	256	256	1180	16
<u>Asus V9999 Ultra Deluxe 256MB</u>	16291	400	256	256	1100	16
<u>MicroStar NX6800Ultra-T2D256E 256MB, PCI-E</u>	16648	400	256	256	1100	16
<u>Sapphire Atlantis ATI Radeon X850 XT Platinum</u>	17552	540	256	256	1180	16

Příloha č. 2: Vstupní data – „prosperita a bankrot“

Úvěr / aktiva	Tržby / aktiva	Zisk / celková aktiva	Dividendy	Výsledek
0,366961576	2,636251738	0,187858279	30	prosperita
1,03219575	1,647638725	-0,14553719	0	bankrot
0,10844603	4,513776786	-0,363222677	0	bankrot
2,344676097	4,509772201	-0,810668028	0	bankrot
1,66467302	5,501549589	-0,458303137	0	bankrot
0,090923099	0,610970194	0,215597493	50	prosperita
0,571309819	3,057737496	0,11623887	150	prosperita
0,67189837	1,740419488	-0,588250634	0	bankrot
7,308322506	18,08469534	-2,22987759	0	bankrot
1,113049173	14,71518718	1,722569013	10	prosperita
0,986573782	2,010830123	-0,018798123	0	bankrot
0,539784901	0,490497583	0,030450071	22	prosperita
0,766504689	3,76712355	0,035022831	100	prosperita
0,756469081	0,020221011	0,100119606	15	bankrot
1,501323469	6,277719146	-0,666669431	0	bankrot
0,381747738	2,043345106	0,409607412	650	prosperita
0,345445174	16,90733191	0,89017976	200	prosperita
0,38392635	0,139651445	0,138052298	25	prosperita
2,049994922	4,976599021	0,341228819	15	prosperita
0,405263069	2,923479387	-0,298725219	0	bankrot
0,035842957	5,700284669	-0,066131697	0	bankrot
4,144827006	69,15039145	-5,243076016	0	prosperita
0,094473883	4,44092018	-0,045045306	0	bankrot
1,204173726	4,501762268	0,351911519	45	prosperita
0,143183934	1,706545933	0,007124309	2	bankrot
1,15847853	2,32430458	0,38317913	56	prosperita
0,436266214	2,863016294	0,123422334	12	bankrot
8,470391052	34,60724957	-2,574808568	0	prosperita
0,240928369	1,449076751	-0,163342087	0	bankrot
0,475701999	1,191242038	-0,140092062	0	bankrot
0,638318285	1,21646912	-0,178645303	0	bankrot
1,684403155	2,60967942	0,351392362	25	prosperita
0,516381959	2,13724331	0,253106859	105	prosperita

4,527017783	23,94349104	0,940712731	50	prosperita
3,178201894	2,542055956	-2,010382465	0	bankrot
0,079502976	2,415095392	-0,042209835	0	bankrot
1,849628205	4,519175012	0,775442391	30	prosperita
2,526260046	0,182231637	-0,46790958	0	bankrot
5,877160187	28,36612057	-1,38452528	0	prosperita
0,093484301	1,009425755	-0,146049882	0	bankrot
0,081497178	0,227839792	0,037609499	60	prosperita
1,768040093	7,739787153	-0,054801321	0	bankrot
0,557353028	1,760998473	0,1849015	450	prosperita
0,53279971	7,844893941	-0,411004781	0	bankrot
0,758378007	2,21945886	0,093161297	80	prosperita
0,488138326	9,17913303	-0,72641958	0	bankrot
0,437661824	1,811398456	0,033628936	20	prosperita
1,397280142	8,086153629	0,181317326	0	bankrot
0,575983236	0,24473618	0,139806611	230	prosperita
0,351121279	2,713632413	-0,060241514	0	bankrot
1,246614097	1,863257645	0,334066349	60	prosperita
1,457907525	5,43974437	-0,654266779	0	bankrot
0,728194382	3,103237345	-0,090651254	0	bankrot
0,338826315	0,769127537	0,216162335	850	prosperita
0,088954953	3,529722492	-0,060411684	0	bankrot
0,4251087	0,428445107	0,327665702	1000	prosperita
1,694468206	3,122668948	0,35489022	35	prosperita
7,32469246	26,609375	1,4921875	20	prosperita
0,508392945	0,544943824	0,110488633	190	prosperita
0,166598839	2,665121431	-0,119040455	0	bankrot
0,068327336	1,977779181	-0,212760137	0	bankrot
0,296681496	2,574546043	0,053372304	80	prosperita
0,034813739	2,370861856	0,110520102	0	bankrot
1,195470341	4,752390591	0,021575148	250	prosperita
0,449137441	0,425365105	-0,100545289	0	bankrot
0,68399343	2,594692478	-0,16870913	0	bankrot
0,05771584	4,799224299	0,337218574	580	prosperita
0,541501894	1,031055121	0,013821542	80	prosperita
2,466188904	9,858848289	-1,0358685	0	bankrot
1,126435267	1,993080155	0,092890082	30	prosperita

0,835269207	2,685699855	0,274915241	1600	prosperita
0,399356676	9,304149325	0,414425216	650	prosperita
2,891825675	11,21616391	-0,543972623	0	bankrot
0,627945412	0,051516219	-0,039012722	0	bankrot
0,236065365	1,252680197	-0,089257679	0	bankrot
0,616596554	1,929718478	0,008554137	10	bankrot
0,187476164	2,754947161	-0,228050723	0	bankrot
0,839315919	4,96824482	0,415584067	450	prosperita
0,108873701	0,352699821	0,039024952	60	prosperita
2,169744824	11,36493593	0,984944482	30	prosperita
0,163853592	4,473429063	-0,117409796	0	bankrot
0,862228109	7,439065727	-0,276941534	0	bankrot
0,574786667	1,092018354	-0,102303441	0	bankrot
0,743883133	0,362266286	-0,205930237	0	bankrot
0,170687625	2,727585053	0,179784142	900	prosperita
0,442972595	2,84751591	-0,077646123	0	bankrot
0,589736098	2,852019416	0,35833959	560	prosperita
0,061911017	0,70137567	-0,011731627	0	bankrot
0,405686236	2,978708078	-0,119569076	0	bankrot
0,756163001	1,500985969	0,004989286	0	bankrot
0,329088011	2,015877416	-0,187678759	0	bankrot
1,32804118	7,326807053	0,750327386	870	prosperita
9,996402673	37,8990927	0,153341757	40	prosperita
0,32668564	0,892783383	-0,276618054	0	bankrot
0,212268915	3,075820039	0,717314313	300	prosperita
1,808289247	3,295854521	-0,544882624	0	bankrot
0,349505034	1,303907434	-0,084031817	0	bankrot
0,170695102	0,902623118	-0,24568823	0	bankrot
0,464606162	0,394489555	-0,102820551	0	bankrot
0,193148352	4,877290032	-0,037265467	0	bankrot

Příloha č. 3: Regresní analýza – grafické karty

Výpočet regresní analýzy se nachází na přiloženém CD v souboru *regrese.xls*.

Příloha č. 4: Zdrojové kódy ukázkových programů

Zdrojové kódy ke všem ukázkovým programům se nachází na přiloženém CD v adresáři „*ukázkové programy*“.

	Rychlost paměti	Velikost paměti	Šířka sběrnice	Rychlost graf. čipu
Determinační index	0,801332578	0,350056443	0,757329378	0,407781113

Průměrný determinační index
0,579124878

Název	x_i	y_i
Asus A7000/T 64MB	300	837
Inno3D GeForce2 MX400 64MB	166	885
MicroStar MX4000-T64 64MB	400	906
Asus A9200SE/T 64MB	333	1060
Sapphire Atlantis ATI Radeon 9200 SE 128MB	333	1127
MicroStar MX4000-T128 128MB	332	1130
Asus V9400 Magic 128MB	332	1147
GigaByte MAYA GV-N52128D 128MB	400	1175
Sapphire Atlantis ATI Radeon 9250 64MB	400	1186
Club3D ATI 9250 128MB(64bit)	400	1220
Sapphire Atlantis ATI Radeon 9250 128MB(64bit)	400	1235
Asus A9200SET 128MB	333	1241
Asus A9250/TD 128MB	400	1290
Asus V9520 Magic 128MB	332	1344
Asus V9520-X/TD 128MB	400	1344
Sapphire Atlantis ATI Radeon 9250 128MB	400	1359
GigaByte MAYA GV-R925128D 128MB	400	1457
Club3D ATI 9250 128MB(128bit)	400	1458
Sapphire Atlantis ATI Radeon 9250 128MB VIVO	400	1517
GigaByte MAYA GV-R955128T 128MB OEM	400	1526
Inno3D GeForce FX 5200 128MB	333	1553
Asus A9250GE/TD 128MB	400	1605
Gainward FX PowerPack Pro/665 TV/DVI 128MB	400	1636
MicroStar RX300SE-TD128E 128MB, PCI-E	400	1642
MicroStar FX5200-TD128 128MB	400	1659
Sapphire Atlantis ATI Radeon 9250 256MB	400	1664
MicroStar NX6200TC-TD64E 64MB, PCI-E	550	1668
GigaByte MAYA GV-R925128VH 128MB VIVO OEM	400	1670
Club3D ATI 9250 128MB(128bit) VIVO	400	1702
Asus V9520/TD 128MB	400	1735
Inno3D GeForce FX-5500 128MB	400	1753
MicroStar FX5500-T128 128MB	400	1783
Asus A9550/TD 128MB	400	1795
Sapphire Atlantis ATI Radeon 9600SE 128MB	400	1808
Sapphire Atlantis ATI Radeon 9550 128MB	400	1846
MicroStar RX300LE-TD128E	400	1984
Club3D FX5200 256MB	286	1990
Asus A9550GE/TD 128MB	400	1994
Club3D ATI 9550 256MB(128bit)	400	2058
Sapphire Atlantis ATI Radeon 9550 128MB VIVO	400	2067
Asus A9250GE/TD 256MB	400	2094
GigaByte MAYA GV-N55128DP 128MB	500	2104
Inno3D GeForce FX-5500 256MB	400	2119
Sapphire Atlantis ATI Radeon 9550 256MB	400	2157
Inno3D GeForce FX-5500 256MB, VIVO	400	2286
Club3D ATI 9600 128MB	400	2353
MicroStar NX6200-TD128E 128MB, PCI-E	550	2393
Sapphire Atlantis ATI Radeon 9600 128MB	400	2418
MicroStar NX6200-TD128 128MB	400	2444
Asus A9600GE/TD 128MB	400	2453
Asus V9570LE/TD 128MB	400	2528
Asus A9550GE/TD 256MB	400	2582
Abit Radeon 9550-256CDT 256MB	400	2636

Asus V9570GE/TD 128MB	400	2756
Sapphire Atlantis ATI Radeon X600Pro 128MB, PCI-E	600	2818
Club3D ATI 9600 256MB	400	2948
Asus V9520 VideoSuite 128MB	400	2983
Sapphire Atlantis ATI Radeon X600Pro 256MB, PCI-E	600	3040
Club3D ATI 9600PRO 128MB	600	3074
Inno3D GeForce 6200 128MB	400	3084
MicroStar RX600XT-TD128E II 128MB, PCI-E	550	3214
Asus N6200GE/TD 128MB	550	3236
Sapphire Atlantis ATI Radeon 9600XT 128MB	600	3286
MicroStar FX5700LE-TD256 256MB	400	3311
Asus V9570GE/TD 256MB	400	3371
Sapphire Atlantis ATI Radeon 9600XT 128MB VIVO	600	3402
Sapphire Atlantis ATI Radeon X700 128MB, PCI-E	700	3408
Sapphire Atlantis ATI Radeon 9800SE 128MB	580	3439
Inno3D GeForce 6600 128MB, AGP	550	3482
Sapphire Atlantis ATI Radeon 9600XT 256MB	600	3502
Gainward FX PowerPack Ultra/1760 XP TV-DVI 128MB	500	3589
Sapphire Atlantis ATI Radeon 9600XT 256MB VIVO	600	3669
Sapphire Atlantis ATI Radeon X700 128MB	500	3785
Inno3D GeForce 6600 256MB, AGP	500	3814
Club3D ATI 9600XT 256MB	600	3835
Gainward FX PowerPack Ultra/1760 XP Golden Sample 128MB	560	3860
MicroStar NX6600-TD128E 128MB, PCI-E	550	3885
Asus N6600/TD 128MB	550	3889
Asus V9570 TD 128MB	500	3922
Gainward FX PowerPack Ultra/1780 TV/DVI 256MB	500	3965
Club3D ATI 9600XT 128MB	680	4046
Sapphire Atlantis ATI Radeon 9800Pro 128MB	680	4081
Sapphire Atlantis ATI Radeon X700 256MB, PCI-E	700	4106
MicroStar NX6600-TD256E 256MB, PCI-E	500	4186
Sapphire Atlantis ATI Radeon X700 256MB	500	4202
MicroStar RX700 PRO-TD128E 128MB, PCI-E	860	4276
Asus V9570/TD 256MB	500	4373
Asus A9600XT 128MB	600	4410
GigaByte MAYA GV-R96X128D 128MB	600	4424
Sapphire Atlantis ATI Radeon X700 Fireblade 256MB	600	4428
Asus N6600/TD 256MB	500	4542
Club3D All-In-Wonder 9800SE 128MB	680	4547
MicroStar NX6600-VTD256EH 256MB, VIVO, PCI-E	500	4555
GigaByte MAYA GV-R96X128DU 128MB	600	4585
Sapphire Atlantis ATI Radeon X700PRO 256MB, PCI-E	860	4720
MicroStar NX6600-VTD256 256MB	600	4855
Sapphire Atlantis ATI Radeon X700PRO 128MB	860	4869
Club3D ATI 9800Pro 256bit 128MB	680	4987
MicroStar NX6600-VTD128E Diamond 128MB, VIVO, PCI-E	800	5130
Inno3D GeForce 6600GT 128MB, AGP	900	5378
Gainward FX PowerPack Ultra/1960 TV-DVI 128MB	900	5380
Inno3D GeForce 6600PE 128MB, AGP	800	5544
GigaByte MAYA GV-N66T128D 128MB	1000	5555
Asus V9999LE/TD 128MB	700	5627
MicroStar NX6600GT-TD128E 128MB, PCI-E	1000	5669
Sapphire Atlantis ATI Radeon X700PRO 256MB	860	5837
MicroStar NX6600-VTD128 Diamond	800	5915
MicroStar NX6600GT-VTD128 128MB	900	6006
Asus N6600GT/TD 128MB	900	6044

MicroStar RX700 PRO-TD256E 256MB, PCI-E	860	6315
GigaByte MAYA GV-N66T128VP 128MB	1120	6334
Gainward FX PowerPack Ultra/1960 TV-DVI-DVI XP 256MB	900	6349
Gainward FX PowerPack Ultra/1960 XP 128MB, AGP	900	6401
Asus N6600GT TOP 128MB	1100	6416
MicroStar NX6800LE-TD128 128MB	700	6545
Gainward FX PowerPack Ultra/1960 XP Golden Sample 128MB, A	1050	6558
Gainward FX PowerPack Ultra/1960 XP 128MB Golden Sample G	1150	7396
Gainward FX PowerPack Ultra/2100 TV/DVI 128MB	700	7396
MicroStar NX6800-TD128 128MB	700	7684
Inno3D GeForce 6800PE 128MB	800	7805
Leadtek Winfast A400 TDH 128MB	700	8098
Asus V9999/TD 128MB	700	8291
Gainward FX PowerPack Ultra/2100 TV/DVI Golden Sample 128M	800	8888
Sapphire Atlantis ATI Radeon X800Pro 256MB	900	9162
MicroStar NX6800-TD256E 256MB, PCI-E	600	9506
Sapphire Atlantis ATI Radeon X800 Pro 256 MB, PCI-E	900	9603
Asus V9999GT/TD 128MB	700	9764
Sapphire Atlantis ATI Radeon X800 XL 256 MB, PCI-E	980	9773
Excibur X800Pro IceQ II 256MB	900	9872
Asus V9999GT/TD 256MB	700	10670
Inno3D GeForce 6800GT 256MB	1100	10734
Sapphire Atlantis ATI Radeon X800 XL Ultimate 256MB	980	10974
Asus AX800XL/2DTV 256MB	980	11071
MicroStar RX800 PRO VTD 256MB VIVO	900	11105
MicroStar NX6800GT-TD256 256MB	1000	11127
Gainward FX PowerPack Ultra/2400 TV/DVI Golden Sample 256M	1100	11284
Club3D ATI X800 Pro 256MB VIVO	900	11401
Gainward FX PowerPack Ultra/2400 TV/DVI 256MB	1000	11768
Club3D ATI X800 Pro 256MB	900	12198
Asus AX800 PRO/TD 256MB	900	12697
Sapphire Atlantis ATI Radeon X800XT 256 MB VIVO	1000	13031
Club3D ATI X800 XT 256MB VIVO	1000	13245
Sapphire Atlantis ATI Radeon X800XT Platinum 256MB	1120	13340
Sapphire Atlantis ATI Radeon X850 XT 256 MB	1080	13923
Asus AX850PRO/HTVD 256MB, VIVO	1080	14340
Gainward FX PowerPack Ultra/2600 TV-DVI 256MB	1100	14530
Sapphire Atlantis ATI Radeon X850 XT 256 MB, VIVO, PCI-E	1080	14542
Inno3D GeForce 6800 Ultra 256MB	1000	14720
MicroStar NX6800Ultra-T2D256 256MB	1100	14732
Asus AX800XT 256MB	1120	15184
Asus AX800XT/TVD 256MB, VIVO	1120	15303
Asus AX850XT/HTVD 256MB, VIVO	1080	15637
Asus V9999 Ultra/2DT 256MB	1100	15637
Sapphire Atlantis ATI Radeon X850 XT 256 MB, VIVO	1080	16041
Gainward FX PowerPack Ultra/2600 Golden Sample 256MB	1200	16101
Asus AX850XTP/HTVD 256MB, VIVO	1180	16291
Asus V9999 Ultra Deluxe 256MB	1100	16291
MicroStar NX6800Ultra-T2D256E 256MB, PCI-E	1100	16648
Sapphire Atlantis ATI Radeon X850 XT Platinum 256 MB, VIVO	1180	17552
Součty	104470	895740
Průměr		5633,585

Sy

Z tabulky je zřejma platnost vztahu $Sy = St + Sr$, protože

3227339182,60

Vzhledk k tomu, ze uvazujeme $n=159$ pozorovani a protoze regresni rovnice ma $p=3$ parametry, j

Y_i	$y_i - \bar{y}$	$Y_i - \bar{Y}$	$y_i - Y_i$	$(y_i - \bar{y})^2$	$(Y_i - \bar{Y})^2$
1360,785125	-4796,584906	-4272,799781	-523,7851245	23007226,76	18256817,97
712,7543829	-4748,584906	-4920,830523	172,2456171	22549058,61	24214573,03
2085,040798	-4727,584906	-3548,544108	-1179,040798	22350059,04	12592165,28
1577,051096	-4573,584906	-4056,533809	-517,0510963	20917678,89	16455466,55
1577,051096	-4506,584906	-4056,533809	-450,0510963	20309307,51	16455466,55
1570,168487	-4503,584906	-4063,416419	-440,1684871	20282277	16511352,99
1570,168487	-4486,584906	-4063,416419	-423,1684871	20129444,12	16511352,99
2085,040798	-4458,584906	-3548,544108	-910,0407979	19878979,36	12592165,28
2085,040798	-4447,584906	-3548,544108	-899,0407979	19781011,49	12592165,28
2085,040798	-4413,584906	-3548,544108	-865,0407979	19479731,72	12592165,28
2085,040798	-4398,584906	-3548,544108	-850,0407979	19347549,17	12592165,28
1577,051096	-4392,584906	-4056,533809	-336,0510963	19294802,15	16455466,55
2085,040798	-4343,584906	-3548,544108	-795,0407979	18866729,83	12592165,28
1570,168487	-4289,584906	-4063,416419	-226,1684871	18400538,66	16511352,99
2085,040798	-4289,584906	-3548,544108	-741,0407979	18400538,66	12592165,28
2085,040798	-4274,584906	-3548,544108	-726,0407979	18272076,12	12592165,28
2085,040798	-4176,584906	-3548,544108	-628,0407979	17443861,47	12592165,28
2085,040798	-4175,584906	-3548,544108	-627,0407979	17435509,3	12592165,28
2085,040798	-4116,584906	-3548,544108	-568,0407979	16946271,29	12592165,28
2085,040798	-4107,584906	-3548,544108	-559,0407979	16872253,76	12592165,28
1577,051096	-4080,584906	-4056,533809	-24,0510963	16651173,17	16455466,55
2085,040798	-4028,584906	-3548,544108	-480,0407979	16229496,34	12592165,28
2085,040798	-3997,584906	-3548,544108	-449,0407979	15980685,08	12592165,28
2085,040798	-3991,584906	-3548,544108	-443,0407979	15932750,06	12592165,28
2085,040798	-3974,584906	-3548,544108	-426,0407979	15797325,17	12592165,28
2085,040798	-3969,584906	-3548,544108	-421,0407979	15757604,32	12592165,28
3557,082389	-3965,584906	-2076,502517	-1889,082389	15725863,64	4311862,703
2085,040798	-3963,584906	-3548,544108	-415,0407979	15710005,3	12592165,28
2085,040798	-3931,584906	-3548,544108	-383,0407979	15457359,87	12592165,28
2085,040798	-3898,584906	-3548,544108	-350,0407979	15198964,27	12592165,28
2085,040798	-3880,584906	-3548,544108	-332,0407979	15058939,21	12592165,28
2085,040798	-3850,584906	-3548,544108	-302,0407979	14827004,12	12592165,28
2085,040798	-3838,584906	-3548,544108	-290,0407979	14734734,08	12592165,28
2085,040798	-3825,584906	-3548,544108	-277,0407979	14635099,87	12592165,28
2085,040798	-3787,584906	-3548,544108	-239,0407979	14345799,42	12592165,28
2085,040798	-3649,584906	-3548,544108	-101,0407979	13319469,98	12592165,28
1275,802938	-3643,584906	-4357,781967	714,1970618	13275710,96	18990263,68
2085,040798	-3639,584906	-3548,544108	-91,0407979	13246578,29	12592165,28
2085,040798	-3575,584906	-3548,544108	-27,0407979	12784807,42	12592165,28
2085,040798	-3566,584906	-3548,544108	-18,0407979	12720527,89	12592165,28
2085,040798	-3539,584906	-3548,544108	8,959202105	12528661,3	12592165,28
3014,980781	-3529,584906	-2618,604125	-910,980781	12457969,61	6857087,562
2085,040798	-3514,584906	-3548,544108	33,9592021	12352307,06	12592165,28
2085,040798	-3476,584906	-3548,544108	71,9592021	12086642,61	12592165,28
2085,040798	-3347,584906	-3548,544108	200,9592021	11206324,7	12592165,28
2085,040798	-3280,584906	-3548,544108	267,9592021	10762237,32	12592165,28
3557,082389	-3240,584906	-2076,502517	-1164,082389	10501390,53	4311862,703
2085,040798	-3215,584906	-3548,544108	332,9592021	10339986,29	12592165,28
2085,040798	-3189,584906	-3548,544108	358,9592021	10173451,87	12592165,28
2085,040798	-3180,584906	-3548,544108	367,9592021	10116120,34	12592165,28
2085,040798	-3105,584906	-3548,544108	442,9592021	9644657,606	12592165,28
2085,040798	-3051,584906	-3548,544108	496,9592021	9312170,436	12592165,28
2085,040798	-2997,584906	-3548,544108	550,9592021	8985515,267	12592165,28

2085,040798	-2877,584906	-3548,544108	670,9592021	8280494,889	12592165,28
4150,605074	-2815,584906	-1482,979832	-1332,605074	7927518,361	2199229,182
2085,040798	-2685,584906	-3548,544108	862,9592021	7212366,286	12592165,28
2085,040798	-2650,584906	-3548,544108	897,9592021	7025600,342	12592165,28
4150,605074	-2593,584906	-1482,979832	-1110,605074	6726682,663	2199229,182
4150,605074	-2559,584906	-1482,979832	-1076,605074	6551474,889	2199229,182
2085,040798	-2549,584906	-3548,544108	998,9592021	6500383,191	12592165,28
3557,082389	-2419,584906	-2076,502517	-343,0823886	5854391,116	4311862,703
3557,082389	-2397,584906	-2076,502517	-321,0823886	5748413,38	4311862,703
4150,605074	-2347,584906	-1482,979832	-864,6050737	5511154,889	2199229,182
2085,040798	-2322,584906	-3548,544108	1225,959202	5394400,644	12592165,28
2085,040798	-2262,584906	-3548,544108	1285,959202	5119290,455	12592165,28
4150,605074	-2231,584906	-1482,979832	-748,6050737	4979971,191	2199229,182
5491,913676	-2225,584906	-141,6712295	-2083,913676	4953228,172	20070,73726
3907,02547	-2194,584906	-1726,559435	-468,0254704	4816202,908	2981007,483
3557,082389	-2151,584906	-2076,502517	-75,08238864	4629317,606	4311862,703
4150,605074	-2131,584906	-1482,979832	-648,6050737	4543654,21	2199229,182
3014,980781	-2044,584906	-2618,604125	574,019219	4180327,436	6857087,562
4150,605074	-1964,584906	-1482,979832	-481,6050737	3859593,852	2199229,182
3014,980781	-1848,584906	-2618,604125	770,019219	3417266,153	6857087,562
3014,980781	-1819,584906	-2618,604125	799,019219	3310889,229	6857087,562
4150,605074	-1798,584906	-1482,979832	-315,6050737	3234907,663	2199229,182
3671,673239	-1773,584906	-1961,911666	188,3267605	3145603,418	3849097,386
3557,082389	-1748,584906	-2076,502517	327,9176114	3057549,172	4311862,703
3557,082389	-1744,584906	-2076,502517	331,9176114	3043576,493	4311862,703
3014,980781	-1711,584906	-2618,604125	907,019219	2929522,889	6857087,562
3014,980781	-1668,584906	-2618,604125	950,019219	2784175,587	6857087,562
5207,197211	-1587,584906	-426,3876947	-1161,197211	2520425,833	181806,4662
5207,197211	-1552,584906	-426,3876947	-1126,197211	2410519,889	181806,4662
5491,913676	-1527,584906	-141,6712295	-1385,913676	2333515,644	20070,73726
3014,980781	-1447,584906	-2618,604125	1171,019219	2095502,059	6857087,562
3014,980781	-1431,584906	-2618,604125	1187,019219	2049435,342	6857087,562
8065,830804	-1357,584906	2432,245899	-3789,830804	1843036,776	5915820,112
3014,980781	-1260,584906	-2618,604125	1358,019219	1589074,304	6857087,562
4150,605074	-1223,584906	-1482,979832	259,3949263	1497160,021	2199229,182
4150,605074	-1209,584906	-1482,979832	273,3949263	1463095,644	2199229,182
4150,605074	-1205,584906	-1482,979832	277,3949263	1453434,965	2199229,182
3014,980781	-1091,584906	-2618,604125	1527,019219	1191557,606	6857087,562
5207,197211	-1086,584906	-426,3876947	-660,1972109	1180666,757	181806,4662
3014,980781	-1078,584906	-2618,604125	1540,019219	1163345,399	6857087,562
4150,605074	-1048,584906	-1482,979832	434,3949263	1099530,304	2199229,182
8065,830804	-913,5849057	2432,245899	-3345,830804	834637,3799	5915820,112
4150,605074	-778,5849057	-1482,979832	704,3949263	606194,4553	2199229,182
8065,830804	-764,5849057	2432,245899	-3196,830804	584590,078	5915820,112
5207,197211	-646,5849057	-426,3876947	-220,1972109	418072,0402	181806,4662
7038,906588	-503,5849057	1405,321683	-1908,906588	253597,7572	1974929,032
8791,58381	-255,5849057	3157,998905	-3413,58381	65323,644	9972957,081
8791,58381	-253,5849057	3157,998905	-3411,58381	64305,30438	9972957,081
7038,906588	-89,58490566	1405,321683	-1494,906588	8025,455322	1974929,032
10749,94534	-78,58490566	5116,360436	-5194,945342	6175,587398	26177144,11
5491,913676	-6,58490566	-141,6712295	135,0863238	43,36098256	20070,73726
10749,94534	35,41509434	5116,360436	-5080,945342	1254,228907	26177144,11
8065,830804	203,4150943	2432,245899	-2228,830804	41377,70061	5915820,112
7038,906588	281,4150943	1405,321683	-1123,906588	79194,45532	1974929,032
8791,58381	372,4150943	3157,998905	-2785,58381	138693,0025	9972957,081
8791,58381	410,4150943	3157,998905	-2747,58381	168440,5497	9972957,081

8065,830804	681,4150943	2432,245899	-1750,830804	464326,5308	5915820,112
13371,48247	700,4150943	7737,897563	-7037,482468	490581,3044	59875058,69
8791,58381	715,4150943	3157,998905	-2442,58381	511818,7572	9972957,081
8791,58381	767,4150943	3157,998905	-2390,58381	588925,927	9972957,081
12913,99118	782,4150943	7280,406277	-6497,991183	612173,3799	53004315,56
5491,913676	911,4150943	-141,6712295	1053,086324	830677,4742	20070,73726
11806,25772	924,4150943	6172,672818	-5248,257724	854543,2666	38101889,72
14073,14572	1762,415094	8439,560814	-6677,14572	3106106,965	71226186,74
5491,913676	1762,415094	-141,6712295	1904,086324	3106106,965	20070,73726
5491,913676	2050,415094	-141,6712295	2192,086324	4204202,059	20070,73726
7038,906588	2171,415094	1405,321683	766,0934116	4715043,512	1974929,032
5491,913676	2464,415094	-141,6712295	2606,086324	6073341,757	20070,73726
5491,913676	2657,415094	-141,6712295	2799,086324	7061854,984	20070,73726
7038,906588	3254,415094	1405,321683	1849,093412	10591217,61	1974929,032
8791,58381	3528,415094	3157,998905	370,4161898	12449713,08	9972957,081
4150,605074	3872,415094	-1482,979832	5355,394926	14995598,66	2199229,182
8791,58381	3969,415094	3157,998905	811,4161898	15756256,19	9972957,081
5491,913676	4130,415094	-141,6712295	4272,086324	17060328,85	20070,73726
10341,81829	4139,415094	4708,233385	-568,8182907	17134757,32	22167461,61
8791,58381	4238,415094	3157,998905	1080,41619	17964162,51	9972957,081
5491,913676	5036,415094	-141,6712295	5178,086324	25365477	20070,73726
12913,99118	5100,415094	7280,406277	-2179,991183	26014234,13	53004315,56
10341,81829	5340,415094	4708,233385	632,1817093	28520033,38	22167461,61
10341,81829	5437,415094	4708,233385	729,1817093	29565482,91	22167461,61
8791,58381	5471,415094	3157,998905	2313,41619	29936383,13	9972957,081
10749,94534	5493,415094	5116,360436	377,0546582	30177609,4	26177144,11
12913,99118	5650,415094	7280,406277	-1629,991183	31927190,74	53004315,56
8791,58381	5767,415094	3157,998905	2609,41619	33263076,87	9972957,081
10749,94534	6134,415094	5116,360436	1018,054658	37631048,55	26177144,11
8791,58381	6564,415094	3157,998905	3406,41619	43091545,53	9972957,081
8791,58381	7063,415094	3157,998905	3905,41619	49891832,79	9972957,081
10749,94534	7397,415094	5116,360436	2281,054658	54721750,08	26177144,11
10749,94534	7611,415094	5116,360436	2495,054658	57933639,74	26177144,11
13371,48247	7706,415094	7737,897563	-31,48246848	59388833,61	59875058,69
12464,72727	8289,415094	6831,142364	1458,27273	68714402,61	46664506
12464,72727	8706,415094	6831,142364	1875,27273	75801663,79	46664506
12913,99118	8896,415094	7280,406277	1616,008817	79146201,53	53004315,56
12464,72727	8908,415094	6831,142364	2077,27273	79359859,49	46664506
10749,94534	9086,415094	5116,360436	3970,054658	82562939,27	26177144,11
12913,99118	9098,415094	7280,406277	1818,008817	82781157,23	53004315,56
13371,48247	9550,415094	7737,897563	1812,517532	91210428,47	59875058,69
13371,48247	9669,415094	7737,897563	1931,517532	93497588,27	59875058,69
12464,72727	10003,41509	6831,142364	3172,27273	100068313,5	46664506
12913,99118	10003,41509	7280,406277	2723,008817	100068313,5	53004315,56
12464,72727	10407,41509	6831,142364	3576,27273	108314288,9	46664506
15283,72133	10467,41509	9650,136428	817,278666	109566778,8	93125133,09
14793,32056	10657,41509	9159,735653	1497,679441	113580496,5	83900757,24
12913,99118	10657,41509	7280,406277	3377,008817	113580496,5	53004315,56
12913,99118	11014,41509	7280,406277	3734,008817	121317339,9	53004315,56
14793,32056	11918,41509	9159,735653	2758,679441	142048618,4	83900757,24
895740	0,00	0,00	0,00	3227339182,60	2586172026,39

St	Sr	Vysledek
2586172026,39	641167156,21	3227339182,60

je rezidualni rozptyl

$$S_R^2 = \frac{S_R}{n-p}$$

4110045,873

Determinacni index je

0,801332578

$$(y_i - Y_i)^2$$

274350,8567
29668,55259
1390137,203
267341,8362
202545,9893
193748,297
179071,5685
828174,2538
808274,3563
748295,582
722569,3581
112930,3393
632089,8703
51152,18456
549141,4641
527135,2402
394435,2438
393180,1622
322670,3481
312526,6137
578,4552331
230439,1676
201637,6382
196285,1486
181510,7615
177275,3535
3568632,271
172258,8639
146720,2529
122528,5602
110251,0915
91228,64359
84123,66444
76751,6037
57140,50306
10209,24284
510077,4431
8288,426881
731,2047508
325,4703887
80,26730235
829885,9833
1153,227408
5178,126768
40384,60091
71802,13399
1355087,808
110861,8303
128851,7088
135393,9744
196212,8547
246968,4486
303556,0424

450186,2509
1775836,283
744698,5845
806330,7286
1233443,63
1159078,485
997919,4875
117705,5254
103093,9003
747541,9335
1502975,965
1653691,069
560409,5564
4342696,21
219047,841
5637,365084
420688,5417
329498,0638
231943,447
592929,5977
638431,7124
99606,56257
35466,96873
107529,9598
110169,3007
822683,8637
902536,5165
1348378,963
1268320,158
1920756,718
1371286,011
1409014,626
14362817,53
1844216,199
67285,72777
74744,78571
76947,94512
2331787,695
435860,3573
2371659,195
188698,952
11194583,77
496172,2121
10219727,19
48486,8117
3643924,363
11652554,43
11638904,09
2234745,708
26987457,1
18248,31488
25816005,57
4967686,754
1263166,019
7759477,164
7549216,794

3065408,505
49526159,49
5966215,67
5714890,954
42223889,42
1108990,805
27544209,13
44584274,96
3625544,728
4805242,451
586899,1153
6791685,927
7834884,248
3419146,445
137208,1536
28680254,82
658396,233
18250721,56
323554,2478
1167299,143
26812577,98
4752361,558
399653,7136
531705,9652
5351894,467
142170,2153
2656871,257
6809052,851
1036435,287
11603671,26
15252275,62
5203210,354
6225297,747
991,1458214
2126559,355
3516647,812
2611484,496
4315061,995
15761333,99
3305156,058
3285219,802
3730759,975
10063314,27
7414777,017
12789726,64
667944,4179
2243043,708
11404188,55
13942821,84
7610312,258
641167156,21

Název	x_i	y_i
Asus A7000/T 64MB	64	837
Inno3D GeForce2 MX400 64MB	64	885
MicroStar MX4000-T64 64MB	64	906
Asus A9200SE/T 64MB	64	1060
Sapphire Atlantis ATI Radeon 9200 SE 128MB	128	1127
MicroStar MX4000-T128 128MB	128	1130
Asus V9400 Magic 128MB	128	1147
GigaByte MAYA GV-N52128D 128MB	128	1175
Sapphire Atlantis ATI Radeon 9250 64MB	64	1186
Club3D ATI 9250 128MB(64bit)	128	1220
Sapphire Atlantis ATI Radeon 9250 128MB(64bit)	128	1235
Asus A9200SET 128MB	128	1241
Asus A9250/TD 128MB	128	1290
Asus V9520 Magic 128MB	128	1344
Asus V9520-X/TD 128MB	128	1344
Sapphire Atlantis ATI Radeon 9250 128MB	128	1359
GigaByte MAYA GV-R925128D 128MB	128	1457
Club3D ATI 9250 128MB(128bit)	128	1458
Sapphire Atlantis ATI Radeon 9250 128MB VIVO	128	1517
GigaByte MAYA GV-R955128T 128MB OEM	128	1526
Inno3D GeForce FX 5200 128MB	128	1553
Asus A9250GE/TD 128MB	128	1605
Gainward FX PowerPack Pro/665 TV/DVI 128MB	128	1636
MicroStar RX300SE-TD128E 128MB, PCI-E	128	1642
MicroStar FX5200-TD128 128MB	128	1659
Sapphire Atlantis ATI Radeon 9250 256MB	256	1664
MicroStar NX6200TC-TD64E 64MB, PCI-E	64	1668
GigaByte MAYA GV-R925128VH 128MB VIVO OEM	128	1670
Club3D ATI 9250 128MB(128bit) VIVO	128	1702
Asus V9520/TD 128MB	128	1735
Inno3D GeForce FX-5500 128MB	128	1753
MicroStar FX5500-T128 128MB	128	1783
Asus A9550/TD 128MB	128	1795
Sapphire Atlantis ATI Radeon 9600SE 128MB	128	1808
Sapphire Atlantis ATI Radeon 9550 128MB	128	1846
MicroStar RX300LE-TD128E	128	1984
Club3D FX5200 256MB	256	1990
Asus A9550GE/TD 128MB	125	1994
Club3D ATI 9550 256MB(128bit)	256	2058
Sapphire Atlantis ATI Radeon 9550 128MB VIVO	128	2067
Asus A9250GE/TD 256MB	256	2094
GigaByte MAYA GV-N55128DP 128MB	128	2104
Inno3D GeForce FX-5500 256MB	256	2119
Sapphire Atlantis ATI Radeon 9550 256MB	256	2157
Inno3D GeForce FX-5500 256MB, VIVO	256	2286
Club3D ATI 9600 128MB	128	2353
MicroStar NX6200-TD128E 128MB, PCI-E	128	2393
Sapphire Atlantis ATI Radeon 9600 128MB	128	2418
MicroStar NX6200-TD128 128MB	128	2444
Asus A9600GE/TD 128MB	128	2453
Asus V9570LE/TD 128MB	128	2528
Asus A9550GE/TD 256MB	256	2582
Abit Radeon 9550-256CDT 256MB	256	2636

Asus V9570GE/TD 128MB	128	2756
Sapphire Atlantis ATI Radeon X600Pro 128MB, PCI-E	128	2818
Club3D ATI 9600 256MB	256	2948
Asus V9520 VideoSuite 128MB	128	2983
Sapphire Atlantis ATI Radeon X600Pro 256MB, PCI-E	256	3040
Club3D ATI 9600PRO 128MB	128	3074
Inno3D GeForce 6200 128MB	128	3084
MicroStar RX600XT-TD128E II 128MB, PCI-E	128	3214
Asus N6200GE/TD 128MB	128	3236
Sapphire Atlantis ATI Radeon 9600XT 128MB	128	3286
MicroStar FX5700LE-TD256 256MB	256	3311
Asus V9570GE/TD 256MB	256	3371
Sapphire Atlantis ATI Radeon 9600XT 128MB VIVO	128	3402
Sapphire Atlantis ATI Radeon X700 128MB, PCI-E	128	3408
Sapphire Atlantis ATI Radeon 9800SE 128MB	128	3439
Inno3D GeForce 6600 128MB, AGP	128	3482
Sapphire Atlantis ATI Radeon 9600XT 256MB	256	3502
Gainward FX PowerPack Ultra/1760 XP TV-DVI 128MB	128	3589
Sapphire Atlantis ATI Radeon 9600XT 256MB VIVO	256	3669
Sapphire Atlantis ATI Radeon X700 128MB	128	3785
Inno3D GeForce 6600 256MB, AGP	256	3814
Club3D ATI 9600XT 256MB	256	3835
Gainward FX PowerPack Ultra/1760 XP Golden Sample 128MB	128	3860
MicroStar NX6600-TD128E 128MB, PCI-E	128	3885
Asus N6600/TD 128MB	128	3889
Asus V9570 TD 128MB	128	3922
Gainward FX PowerPack Ultra/1780 TV/DVI 256MB	256	3965
Club3D ATI 9600XT 128MB	128	4046
Sapphire Atlantis ATI Radeon 9800Pro 128MB	128	4081
Sapphire Atlantis ATI Radeon X700 256MB, PCI-E	256	4106
MicroStar NX6600-TD256E 256MB, PCI-E	256	4186
Sapphire Atlantis ATI Radeon X700 256MB	256	4202
MicroStar RX700 PRO-TD128E 128MB, PCI-E	128	4276
Asus V9570/TD 256MB	256	4373
Asus A9600XT 128MB	128	4410
GigaByte MAYA GV-R96X128D 128MB	128	4424
Sapphire Atlantis ATI Radeon X700 Fireblade 256MB	256	4428
Asus N6600/TD 256MB	256	4542
Club3D All-In-Wonder 9800SE 128MB	128	4547
MicroStar NX6600-VTD256EH 256MB, VIVO, PCI-E	128	4555
GigaByte MAYA GV-R96X128DU 128MB	125	4585
Sapphire Atlantis ATI Radeon X700PRO 256MB, PCI-E	256	4720
MicroStar NX6600-VTD256 256MB	256	4855
Sapphire Atlantis ATI Radeon X700PRO 128MB	128	4869
Club3D ATI 9800Pro 256bit 128MB	128	4987
MicroStar NX6600-VTD128E Diamond 128MB, VIVO, PCI-E	128	5130
Inno3D GeForce 6600GT 128MB, AGP	128	5378
Gainward FX PowerPack Ultra/1960 TV-DVI 128MB	128	5380
Inno3D GeForce 6600PE 128MB, AGP	128	5544
GigaByte MAYA GV-N66T128D 128MB	128	5555
Asus V9999LE/TD 128MB	128	5627
MicroStar NX6600GT-TD128E 128MB, PCI-E	128	5669
Sapphire Atlantis ATI Radeon X700PRO 256MB	256	5837
MicroStar NX6600-VTD128 Diamond	128	5915
MicroStar NX6600GT-VTD128 128MB	128	6006
Asus N6600GT/TD 128MB	128	6044

MicroStar RX700 PRO-TD256E 256MB, PCI-E	256	6315
GigaByte MAYA GV-N66T128VP 128MB	128	6334
Gainward FX PowerPack Ultra/1960 TV-DVI-DVI XP 256MB	256	6349
Gainward FX PowerPack Ultra/1960 XP 128MB, AGP	128	6401
Asus N6600GT TOP 128MB	128	6416
MicroStar NX6800LE-TD128 128MB	128	6545
Gainward FX PowerPack Ultra/1960 XP Golden Sample 128MB, A	128	6558
Gainward FX PowerPack Ultra/1960 XP 128MB Golden Sample G	128	7396
Gainward FX PowerPack Ultra/2100 TV/DVI 128MB	128	7396
MicroStar NX6800-TD128 128MB	128	7684
Inno3D GeForce 6800PE 128MB	128	7805
Leadtek Winfast A400 TDH 128MB	128	8098
Asus V9999/TD 128MB	128	8291
Gainward FX PowerPack Ultra/2100 TV/DVI Golden Sample 128M	128	8888
Sapphire Atlantis ATI Radeon X800Pro 256MB	256	9162
MicroStar NX6800-TD256E 256MB, PCI-E	256	9506
Sapphire Atlantis ATI Radeon X800 Pro 256 MB, PCI-E	256	9603
Asus V9999GT/TD 128MB	128	9764
Sapphire Atlantis ATI Radeon X800 XL 256 MB, PCI-E	256	9773
Excibur X800Pro IceQ II 256MB	256	9872
Asus V9999GT/TD 256MB	256	10670
Inno3D GeForce 6800GT 256MB	256	10734
Sapphire Atlantis ATI Radeon X800 XL Ultimate 256MB	256	10974
Asus AX800XL/2DTV 256MB	256	11071
MicroStar RX800 PRO VTD 256MB VIVO	256	11105
MicroStar NX6800GT-TD256 256MB	256	11127
Gainward FX PowerPack Ultra/2400 TV/DVI Golden Sample 256M	256	11284
Club3D ATI X800 Pro 256MB VIVO	256	11401
Gainward FX PowerPack Ultra/2400 TV/DVI 256MB	256	11768
Club3D ATI X800 Pro 256MB	256	12198
Asus AX800 PRO/TD 256MB	256	12697
Sapphire Atlantis ATI Radeon X800XT 256 MB VIVO	256	13031
Club3D ATI X800 XT 256MB VIVO	256	13245
Sapphire Atlantis ATI Radeon X800XT Platinum 256MB	256	13340
Sapphire Atlantis ATI Radeon X850 XT 256 MB	256	13923
Asus AX850PRO/HTVD 256MB, VIVO	256	14340
Gainward FX PowerPack Ultra/2600 TV-DVI 256MB	256	14530
Sapphire Atlantis ATI Radeon X850 XT 256 MB, VIVO, PCI-E	256	14542
Inno3D GeForce 6800 Ultra 256MB	256	14720
MicroStar NX6800Ultra-T2D256 256MB	256	14732
Asus AX800XT 256MB	256	15184
Asus AX800XT/TVD 256MB, VIVO	256	15303
Asus AX850XT/HTVD 256MB, VIVO	256	15637
Asus V9999 Ultra/2DT 256MB	256	15637
Sapphire Atlantis ATI Radeon X850 XT 256 MB, VIVO	256	16041
Gainward FX PowerPack Ultra/2600 Golden Sample 256MB	256	16101
Asus AX850XTP/HTVD 256MB, VIVO	256	16291
Asus V9999 Ultra Deluxe 256MB	256	16291
MicroStar NX6800Ultra-T2D256E 256MB, PCI-E	256	16648
Sapphire Atlantis ATI Radeon X850 XT Platinum 256 MB, VIVO	256	17552
Součty	28154	895740
Průměr		5633,585

Sy

Z tabulky je zřejma platnost vztahu $Sy = St + Sr$, protože

3227339182,60

Vzhledk k tomu, ze uvazujeme $n=159$ pozorovani a protoze regresni rovnice ma $p=3$ parametry, j

Y_i	$y_i - \bar{y}$	$Y_i - \bar{Y}$	$y_i - Y_i$	$(y_i - \bar{y})^2$	$(Y_i - \bar{Y})^2$
1087,819932	-4796,584906	-4545,764974	-250,8199321	23007226,76	20663979,2
1087,819932	-4748,584906	-4545,764974	-202,8199321	22549058,61	20663979,2
1087,819932	-4727,584906	-4545,764974	-181,8199321	22350059,04	20663979,2
1087,819932	-4573,584906	-4545,764974	-27,81993207	20917678,89	20663979,2
3645,655719	-4506,584906	-1987,929186	-2518,655719	20309307,51	3951862,45
3645,655719	-4503,584906	-1987,929186	-2515,655719	20282277	3951862,45
3645,655719	-4486,584906	-1987,929186	-2498,655719	20129444,12	3951862,45
3645,655719	-4458,584906	-1987,929186	-2470,655719	19878979,36	3951862,45
1087,819932	-4447,584906	-4545,764974	98,18006793	19781011,49	20663979,2
3645,655719	-4413,584906	-1987,929186	-2425,655719	19479731,72	3951862,45
3645,655719	-4398,584906	-1987,929186	-2410,655719	19347549,17	3951862,45
3645,655719	-4392,584906	-1987,929186	-2404,655719	19294802,15	3951862,45
3645,655719	-4343,584906	-1987,929186	-2355,655719	18866729,83	3951862,45
3645,655719	-4289,584906	-1987,929186	-2301,655719	18400538,66	3951862,45
3645,655719	-4289,584906	-1987,929186	-2301,655719	18400538,66	3951862,45
3645,655719	-4274,584906	-1987,929186	-2286,655719	18272076,12	3951862,45
3645,655719	-4176,584906	-1987,929186	-2188,655719	17443861,47	3951862,45
3645,655719	-4175,584906	-1987,929186	-2187,655719	17435509,3	3951862,45
3645,655719	-4116,584906	-1987,929186	-2128,655719	16946271,29	3951862,45
3645,655719	-4107,584906	-1987,929186	-2119,655719	16872253,76	3951862,45
3645,655719	-4080,584906	-1987,929186	-2092,655719	16651173,17	3951862,45
3645,655719	-4028,584906	-1987,929186	-2040,655719	16229496,34	3951862,45
3645,655719	-3997,584906	-1987,929186	-2009,655719	15980685,08	3951862,45
3645,655719	-3991,584906	-1987,929186	-2003,655719	15932750,06	3951862,45
3645,655719	-3974,584906	-1987,929186	-1986,655719	15797325,17	3951862,45
8827,976736	-3969,584906	3194,39183	-7163,976736	15757604,32	10204139,16
1087,819932	-3965,584906	-4545,764974	580,1800679	15725863,64	20663979,2
3645,655719	-3963,584906	-1987,929186	-1975,655719	15710005,3	3951862,45
3645,655719	-3931,584906	-1987,929186	-1943,655719	15457359,87	3951862,45
3645,655719	-3898,584906	-1987,929186	-1910,655719	15198964,27	3951862,45
3645,655719	-3880,584906	-1987,929186	-1892,655719	15058939,21	3951862,45
3645,655719	-3850,584906	-1987,929186	-1862,655719	14827004,12	3951862,45
3645,655719	-3838,584906	-1987,929186	-1850,655719	14734734,08	3951862,45
3645,655719	-3825,584906	-1987,929186	-1837,655719	14635099,87	3951862,45
3645,655719	-3787,584906	-1987,929186	-1799,655719	14345799,42	3951862,45
3645,655719	-3649,584906	-1987,929186	-1661,655719	13319469,98	3951862,45
8827,976736	-3643,584906	3194,39183	-6837,976736	13275710,96	10204139,16
3525,260876	-3639,584906	-2108,32403	-1531,260876	13246578,29	4445030,216
8827,976736	-3575,584906	3194,39183	-6769,976736	12784807,42	10204139,16
3645,655719	-3566,584906	-1987,929186	-1578,655719	12720527,89	3951862,45
8827,976736	-3539,584906	3194,39183	-6733,976736	12528661,3	10204139,16
3645,655719	-3529,584906	-1987,929186	-1541,655719	12457969,61	3951862,45
8827,976736	-3514,584906	3194,39183	-6708,976736	12352307,06	10204139,16
8827,976736	-3476,584906	3194,39183	-6670,976736	12086642,61	10204139,16
8827,976736	-3347,584906	3194,39183	-6541,976736	11206324,7	10204139,16
3645,655719	-3280,584906	-1987,929186	-1292,655719	10762237,32	3951862,45
3645,655719	-3240,584906	-1987,929186	-1252,655719	10501390,53	3951862,45
3645,655719	-3215,584906	-1987,929186	-1227,655719	10339986,29	3951862,45
3645,655719	-3189,584906	-1987,929186	-1201,655719	10173451,87	3951862,45
3645,655719	-3180,584906	-1987,929186	-1192,655719	10116120,34	3951862,45
3645,655719	-3105,584906	-1987,929186	-1117,655719	9644657,606	3951862,45
8827,976736	-3051,584906	3194,39183	-6245,976736	9312170,436	10204139,16
8827,976736	-2997,584906	3194,39183	-6191,976736	8985515,267	10204139,16

3645,655719	-2877,584906	-1987,929186	-889,6557192	8280494,889	3951862,45
3645,655719	-2815,584906	-1987,929186	-827,6557192	7927518,361	3951862,45
8827,976736	-2685,584906	3194,39183	-5879,976736	7212366,286	10204139,16
3645,655719	-2650,584906	-1987,929186	-662,6557192	7025600,342	3951862,45
8827,976736	-2593,584906	3194,39183	-5787,976736	6726682,663	10204139,16
3645,655719	-2559,584906	-1987,929186	-571,6557192	6551474,889	3951862,45
3645,655719	-2549,584906	-1987,929186	-561,6557192	6500383,191	3951862,45
3645,655719	-2419,584906	-1987,929186	-431,6557192	5854391,116	3951862,45
3645,655719	-2397,584906	-1987,929186	-409,6557192	5748413,38	3951862,45
3645,655719	-2347,584906	-1987,929186	-359,6557192	5511154,889	3951862,45
8827,976736	-2322,584906	3194,39183	-5516,976736	5394400,644	10204139,16
8827,976736	-2262,584906	3194,39183	-5456,976736	5119290,455	10204139,16
3645,655719	-2231,584906	-1987,929186	-243,6557192	4979971,191	3951862,45
3645,655719	-2225,584906	-1987,929186	-237,6557192	4953228,172	3951862,45
3645,655719	-2194,584906	-1987,929186	-206,6557192	4816202,908	3951862,45
3645,655719	-2151,584906	-1987,929186	-163,6557192	4629317,606	3951862,45
8827,976736	-2131,584906	3194,39183	-5325,976736	4543654,21	10204139,16
3645,655719	-2044,584906	-1987,929186	-56,65571918	4180327,436	3951862,45
8827,976736	-1964,584906	3194,39183	-5158,976736	3859593,852	10204139,16
3645,655719	-1848,584906	-1987,929186	139,3442808	3417266,153	3951862,45
8827,976736	-1819,584906	3194,39183	-5013,976736	3310889,229	10204139,16
8827,976736	-1798,584906	3194,39183	-4992,976736	3234907,663	10204139,16
3645,655719	-1773,584906	-1987,929186	214,3442808	3145603,418	3951862,45
3645,655719	-1748,584906	-1987,929186	239,3442808	3057549,172	3951862,45
3645,655719	-1744,584906	-1987,929186	243,3442808	3043576,493	3951862,45
3645,655719	-1711,584906	-1987,929186	276,3442808	2929522,889	3951862,45
8827,976736	-1668,584906	3194,39183	-4862,976736	2784175,587	10204139,16
3645,655719	-1587,584906	-1987,929186	400,3442808	2520425,833	3951862,45
3645,655719	-1552,584906	-1987,929186	435,3442808	2410519,889	3951862,45
8827,976736	-1527,584906	3194,39183	-4721,976736	2333515,644	10204139,16
8827,976736	-1447,584906	3194,39183	-4641,976736	2095502,059	10204139,16
8827,976736	-1431,584906	3194,39183	-4625,976736	2049435,342	10204139,16
3645,655719	-1357,584906	-1987,929186	630,3442808	1843036,776	3951862,45
8827,976736	-1260,584906	3194,39183	-4454,976736	1589074,304	10204139,16
3645,655719	-1223,584906	-1987,929186	764,3442808	1497160,021	3951862,45
3645,655719	-1209,584906	-1987,929186	778,3442808	1463095,644	3951862,45
8827,976736	-1205,584906	3194,39183	-4399,976736	1453434,965	10204139,16
8827,976736	-1091,584906	3194,39183	-4285,976736	1191557,606	10204139,16
3645,655719	-1086,584906	-1987,929186	901,3442808	1180666,757	3951862,45
3645,655719	-1078,584906	-1987,929186	909,3442808	1163345,399	3951862,45
3525,260876	-1048,584906	-2108,32403	1059,739124	1099530,304	4445030,216
8827,976736	-913,5849057	3194,39183	-4107,976736	834637,3799	10204139,16
8827,976736	-778,5849057	3194,39183	-3972,976736	606194,4553	10204139,16
3645,655719	-764,5849057	-1987,929186	1223,344281	584590,078	3951862,45
3645,655719	-646,5849057	-1987,929186	1341,344281	418072,0402	3951862,45
3645,655719	-503,5849057	-1987,929186	1484,344281	253597,7572	3951862,45
3645,655719	-255,5849057	-1987,929186	1732,344281	65323,644	3951862,45
3645,655719	-253,5849057	-1987,929186	1734,344281	64305,30438	3951862,45
3645,655719	-89,58490566	-1987,929186	1898,344281	8025,455322	3951862,45
3645,655719	-78,58490566	-1987,929186	1909,344281	6175,587398	3951862,45
3645,655719	-6,58490566	-1987,929186	1981,344281	43,36098256	3951862,45
3645,655719	35,41509434	-1987,929186	2023,344281	1254,228907	3951862,45
8827,976736	203,4150943	3194,39183	-2990,976736	41377,70061	10204139,16
3645,655719	281,4150943	-1987,929186	2269,344281	79194,45532	3951862,45
3645,655719	372,4150943	-1987,929186	2360,344281	138693,0025	3951862,45
3645,655719	410,4150943	-1987,929186	2398,344281	168440,5497	3951862,45

8827,976736	681,4150943	3194,39183	-2512,976736	464326,5308	10204139,16
3645,655719	700,4150943	-1987,929186	2688,344281	490581,3044	3951862,45
8827,976736	715,4150943	3194,39183	-2478,976736	511818,7572	10204139,16
3645,655719	767,4150943	-1987,929186	2755,344281	588925,927	3951862,45
3645,655719	782,4150943	-1987,929186	2770,344281	612173,3799	3951862,45
3645,655719	911,4150943	-1987,929186	2899,344281	830677,4742	3951862,45
3645,655719	924,4150943	-1987,929186	2912,344281	854543,2666	3951862,45
3645,655719	1762,415094	-1987,929186	3750,344281	3106106,965	3951862,45
3645,655719	1762,415094	-1987,929186	3750,344281	3106106,965	3951862,45
3645,655719	2050,415094	-1987,929186	4038,344281	4204202,059	3951862,45
3645,655719	2171,415094	-1987,929186	4159,344281	4715043,512	3951862,45
3645,655719	2464,415094	-1987,929186	4452,344281	6073341,757	3951862,45
3645,655719	2657,415094	-1987,929186	4645,344281	7061854,984	3951862,45
3645,655719	3254,415094	-1987,929186	5242,344281	10591217,61	3951862,45
8827,976736	3528,415094	3194,39183	334,0232643	12449713,08	10204139,16
8827,976736	3872,415094	3194,39183	678,0232643	14995598,66	10204139,16
8827,976736	3969,415094	3194,39183	775,0232643	15756256,19	10204139,16
3645,655719	4130,415094	-1987,929186	6118,344281	17060328,85	3951862,45
8827,976736	4139,415094	3194,39183	945,0232643	17134757,32	10204139,16
8827,976736	4238,415094	3194,39183	1044,023264	17964162,51	10204139,16
8827,976736	5036,415094	3194,39183	1842,023264	25365477	10204139,16
8827,976736	5100,415094	3194,39183	1906,023264	26014234,13	10204139,16
8827,976736	5340,415094	3194,39183	2146,023264	28520033,38	10204139,16
8827,976736	5437,415094	3194,39183	2243,023264	29565482,91	10204139,16
8827,976736	5471,415094	3194,39183	2277,023264	29936383,13	10204139,16
8827,976736	5493,415094	3194,39183	2299,023264	30177609,4	10204139,16
8827,976736	5650,415094	3194,39183	2456,023264	31927190,74	10204139,16
8827,976736	5767,415094	3194,39183	2573,023264	33263076,87	10204139,16
8827,976736	6134,415094	3194,39183	2940,023264	37631048,55	10204139,16
8827,976736	6564,415094	3194,39183	3370,023264	43091545,53	10204139,16
8827,976736	7063,415094	3194,39183	3869,023264	49891832,79	10204139,16
8827,976736	7397,415094	3194,39183	4203,023264	54721750,08	10204139,16
8827,976736	7611,415094	3194,39183	4417,023264	57933639,74	10204139,16
8827,976736	7706,415094	3194,39183	4512,023264	59388833,61	10204139,16
8827,976736	8289,415094	3194,39183	5095,023264	68714402,61	10204139,16
8827,976736	8706,415094	3194,39183	5512,023264	75801663,79	10204139,16
8827,976736	8896,415094	3194,39183	5702,023264	79146201,53	10204139,16
8827,976736	8908,415094	3194,39183	5714,023264	79359859,49	10204139,16
8827,976736	9086,415094	3194,39183	5892,023264	82562939,27	10204139,16
8827,976736	9098,415094	3194,39183	5904,023264	82781157,23	10204139,16
8827,976736	9550,415094	3194,39183	6356,023264	91210428,47	10204139,16
8827,976736	9669,415094	3194,39183	6475,023264	93497588,27	10204139,16
8827,976736	10003,41509	3194,39183	6809,023264	100068313,5	10204139,16
8827,976736	10003,41509	3194,39183	6809,023264	100068313,5	10204139,16
8827,976736	10407,41509	3194,39183	7213,023264	108314288,9	10204139,16
8827,976736	10467,41509	3194,39183	7273,023264	109566778,8	10204139,16
8827,976736	10657,41509	3194,39183	7463,023264	113580496,5	10204139,16
8827,976736	10657,41509	3194,39183	7463,023264	113580496,5	10204139,16
8827,976736	11014,41509	3194,39183	7820,023264	121317339,9	10204139,16
8827,976736	11918,41509	3194,39183	8724,023264	142048618,4	10204139,16
895740	0,00	0,00	0,00	3227339182,60	1129750875,29

St	Sr	Vysledek
1129750875,29	2097588307,31	3227339182,60

je rezidualni rozptyl

$$S_R^2 = \frac{S_R}{n - p}$$

13446078,89

Determinacni index je

0,350056443

$$(y_i - Y_i)^2$$

62910,63832
41135,92485
33058,4877
773,9486204
6343626,632
6328523,697
6243280,403
6104139,683
9639,325739
5883805,668
5811260,996
5782369,128
5549113,867
5297619,05
5297619,05
5228794,378
4790213,857
4785837,546
4531175,171
4492940,368
4379207,959
4164275,764
4038716,11
4014636,241
3946800,947
51322562,67
336608,9112
3903215,521
3777797,555
3650605,277
3582145,671
3469486,328
3424926,591
3376978,542
3238760,708
2761099,729
46757925,84
2344759,869
45832585
2492153,88
45346442,68
2376702,356
45010368,84
44501930,61
42797459,61
1670958,808
1569146,351
1507138,565
1443976,467
1422427,665
1249154,307
39012225,38
38340575,9

791487,2987
685013,9895
34574126,41
439112,6022
33500674,69
326790,2613
315457,1469
186326,6599
167817,8083
129352,2363
30437032,3
29778595,09
59368,10949
56480,24086
42706,58627
26783,19442
28366028,19
3209,870516
26615040,96
19416,8286
25139962,71
24929816,68
45943,47072
57285,68476
59216,43901
76366,16154
23648542,73
160275,5432
189524,6428
22297064,29
21547948,02
21399660,76
397333,9124
19846817,72
584222,1796
605819,8195
19359795,28
18369596,58
812421,5126
826907,0211
1123047,012
16875472,86
15784544,14
1496571,229
1799204,48
2203277,944
3001016,707
3007950,084
3603711,009
3645595,583
3925725,159
4093922,079
8945941,834
5149923,465
5571225,124
5752055,289

6315052,074
7227194,972
6145325,656
7591922,106
7674807,434
8406197,259
8481749,21
14065082,22
14065082,22
16308224,53
17300144,85
19823369,59
21579223,49
27482173,56
111571,5411
459715,5469
600661,0601
37434136,74
893068,97
1089984,576
3393049,706
3632924,684
4605415,851
5031153,364
5184834,946
5285507,97
6032050,275
6620448,718
8643736,794
11357056,8
14969341,02
17665404,56
19510094,52
20358353,94
25959262,06
30382400,47
32513069,31
32650061,86
34715938,15
34857490,7
40399031,74
41925926,27
46362797,81
46362797,81
52027704,61
52896867,4
55696716,24
55696716,24
61152763,85
76108581,92

2097588307,31

Název	x_i	y_i
Asus A7000/T 64MB	64	837
Inno3D GeForce2 MX400 64MB	64	885
MicroStar MX4000-T64 64MB	32	906
Asus A9200SE/T 64MB	64	1060
Sapphire Atlantis ATI Radeon 9200 SE 128MB	64	1127
MicroStar MX4000-T128 128MB	64	1130
Asus V9400 Magic 128MB	64	1147
GigaByte MAYA GV-N52128D 128MB	64	1175
Sapphire Atlantis ATI Radeon 9250 64MB	64	1186
Club3D ATI 9250 128MB(64bit)	64	1220
Sapphire Atlantis ATI Radeon 9250 128MB(64bit)	64	1235
Asus A9200SET 128MB	64	1241
Asus A9250/TD 128MB	64	1290
Asus V9520 Magic 128MB	64	1344
Asus V9520-X/TD 128MB	64	1344
Sapphire Atlantis ATI Radeon 9250 128MB	128	1359
GigaByte MAYA GV-R925128D 128MB	128	1457
Club3D ATI 9250 128MB(128bit)	128	1458
Sapphire Atlantis ATI Radeon 9250 128MB VIVO	128	1517
GigaByte MAYA GV-R955128T 128MB OEM	64	1526
Inno3D GeForce FX 5200 128MB	128	1553
Asus A9250GE/TD 128MB	128	1605
Gainward FX PowerPack Pro/665 TV/DVI 128MB	64	1636
MicroStar RX300SE-TD128E 128MB, PCI-E	64	1642
MicroStar FX5200-TD128 128MB	64	1659
Sapphire Atlantis ATI Radeon 9250 256MB	128	1664
MicroStar NX6200TC-TD64E 64MB, PCI-E	128	1668
GigaByte MAYA GV-R925128VH 128MB VIVO OEM	128	1670
Club3D ATI 9250 128MB(128bit) VIVO	128	1702
Asus V9520/TD 128MB	128	1735
Inno3D GeForce FX-5500 128MB	128	1753
MicroStar FX5500-T128 128MB	64	1783
Asus A9550/TD 128MB	64	1795
Sapphire Atlantis ATI Radeon 9600SE 128MB	64	1808
Sapphire Atlantis ATI Radeon 9550 128MB	128	1846
MicroStar RX300LE-TD128E	64	1984
Club3D FX5200 256MB	128	1990
Asus A9550GE/TD 128MB	128	1994
Club3D ATI 9550 256MB(128bit)	128	2058
Sapphire Atlantis ATI Radeon 9550 128MB VIVO	128	2067
Asus A9250GE/TD 256MB	128	2094
GigaByte MAYA GV-N55128DP 128MB	128	2104
Inno3D GeForce FX-5500 256MB	128	2119
Sapphire Atlantis ATI Radeon 9550 256MB	128	2157
Inno3D GeForce FX-5500 256MB, VIVO	128	2286
Club3D ATI 9600 128MB	128	2353
MicroStar NX6200-TD128E 128MB, PCI-E	128	2393
Sapphire Atlantis ATI Radeon 9600 128MB	128	2418
MicroStar NX6200-TD128 128MB	128	2444
Asus A9600GE/TD 128MB	128	2453
Asus V9570LE/TD 128MB	64	2528
Asus A9550GE/TD 256MB	128	2582
Abit Radeon 9550-256CDT 256MB	128	2636

Asus V9570GE/TD 128MB	128	2756
Sapphire Atlantis ATI Radeon X600Pro 128MB, PCI-E	128	2818
Club3D ATI 9600 256MB	128	2948
Asus V9520 VideoSuite 128MB	128	2983
Sapphire Atlantis ATI Radeon X600Pro 256MB, PCI-E	128	3040
Club3D ATI 9600PRO 128MB	128	3074
Inno3D GeForce 6200 128MB	128	3084
MicroStar RX600XT-TD128E II 128MB, PCI-E	128	3214
Asus N6200GE/TD 128MB	128	3236
Sapphire Atlantis ATI Radeon 9600XT 128MB	128	3286
MicroStar FX5700LE-TD256 256MB	128	3311
Asus V9570GE/TD 256MB	128	3371
Sapphire Atlantis ATI Radeon 9600XT 128MB VIVO	128	3402
Sapphire Atlantis ATI Radeon X700 128MB, PCI-E	128	3408
Sapphire Atlantis ATI Radeon 9800SE 128MB	128	3439
Inno3D GeForce 6600 128MB, AGP	128	3482
Sapphire Atlantis ATI Radeon 9600XT 256MB	128	3502
Gainward FX PowerPack Ultra/1760 XP TV-DVI 128MB	128	3589
Sapphire Atlantis ATI Radeon 9600XT 256MB VIVO	128	3669
Sapphire Atlantis ATI Radeon X700 128MB	128	3785
Inno3D GeForce 6600 256MB, AGP	128	3814
Club3D ATI 9600XT 256MB	128	3835
Gainward FX PowerPack Ultra/1760 XP Golden Sample 128MB	128	3860
MicroStar NX6600-TD128E 128MB, PCI-E	128	3885
Asus N6600/TD 128MB	128	3889
Asus V9570 TD 128MB	128	3922
Gainward FX PowerPack Ultra/1780 TV/DVI 256MB	128	3965
Club3D ATI 9600XT 128MB	128	4046
Sapphire Atlantis ATI Radeon 9800Pro 128MB	256	4081
Sapphire Atlantis ATI Radeon X700 256MB, PCI-E	128	4106
MicroStar NX6600-TD256E 256MB, PCI-E	128	4186
Sapphire Atlantis ATI Radeon X700 256MB	128	4202
MicroStar RX700 PRO-TD128E 128MB, PCI-E	128	4276
Asus V9570/TD 256MB	128	4373
Asus A9600XT 128MB	128	4410
GigaByte MAYA GV-R96X128D 128MB	128	4424
Sapphire Atlantis ATI Radeon X700 Fireblade 256MB	128	4428
Asus N6600/TD 256MB	128	4542
Club3D All-In-Wonder 9800SE 128MB	256	4547
MicroStar NX6600-VTD256EH 256MB, VIVO, PCI-E	128	4555
GigaByte MAYA GV-R96X128DU 128MB	128	4585
Sapphire Atlantis ATI Radeon X700PRO 256MB, PCI-E	128	4720
MicroStar NX6600-VTD256 256MB	128	4855
Sapphire Atlantis ATI Radeon X700PRO 128MB	128	4869
Club3D ATI 9800Pro 256bit 128MB	256	4987
MicroStar NX6600-VTD128E Diamond 128MB, VIVO, PCI-E	128	5130
Inno3D GeForce 6600GT 128MB, AGP	128	5378
Gainward FX PowerPack Ultra/1960 TV-DVI 128MB	128	5380
Inno3D GeForce 6600PE 128MB, AGP	128	5544
GigaByte MAYA GV-N66T128D 128MB	128	5555
Asus V9999LE/TD 128MB	256	5627
MicroStar NX6600GT-TD128E 128MB, PCI-E	128	5669
Sapphire Atlantis ATI Radeon X700PRO 256MB	128	5837
MicroStar NX6600-VTD128 Diamond	128	5915
MicroStar NX6600GT-VTD128 128MB	128	6006
Asus N6600GT/TD 128MB	128	6044

MicroStar RX700 PRO-TD256E 256MB, PCI-E	128	6315
GigaByte MAYA GV-N66T128VP 128MB	128	6334
Gainward FX PowerPack Ultra/1960 TV-DVI-DVI XP 256MB	128	6349
Gainward FX PowerPack Ultra/1960 XP 128MB, AGP	128	6401
Asus N6600GT TOP 128MB	128	6416
MicroStar NX6800LE-TD128 128MB	256	6545
Gainward FX PowerPack Ultra/1960 XP Golden Sample 128MB, A	128	6558
Gainward FX PowerPack Ultra/1960 XP 128MB Golden Sample G	128	7396
Gainward FX PowerPack Ultra/2100 TV/DVI 128MB	256	7396
MicroStar NX6800-TD128 128MB	256	7684
Inno3D GeForce 6800PE 128MB	256	7805
Leadtek Winfast A400 TDH 128MB	256	8098
Asus V9999/TD 128MB	256	8291
Gainward FX PowerPack Ultra/2100 TV/DVI Golden Sample 128M	256	8888
Sapphire Atlantis ATI Radeon X800Pro 256MB	256	9162
MicroStar NX6800-TD256E 256MB, PCI-E	256	9506
Sapphire Atlantis ATI Radeon X800 Pro 256 MB, PCI-E	256	9603
Asus V9999GT/TD 128MB	256	9764
Sapphire Atlantis ATI Radeon X800 XL 256 MB, PCI-E	256	9773
Excibur X800Pro IceQ II 256MB	256	9872
Asus V9999GT/TD 256MB	256	10670
Inno3D GeForce 6800GT 256MB	256	10734
Sapphire Atlantis ATI Radeon X800 XL Ultimate 256MB	256	10974
Asus AX800XL/2DTV 256MB	256	11071
MicroStar RX800 PRO VTD 256MB VIVO	256	11105
MicroStar NX6800GT-TD256 256MB	256	11127
Gainward FX PowerPack Ultra/2400 TV/DVI Golden Sample 256M	256	11284
Club3D ATI X800 Pro 256MB VIVO	256	11401
Gainward FX PowerPack Ultra/2400 TV/DVI 256MB	256	11768
Club3D ATI X800 Pro 256MB	256	12198
Asus AX800 PRO/TD 256MB	256	12697
Sapphire Atlantis ATI Radeon X800XT 256 MB VIVO	256	13031
Club3D ATI X800 XT 256MB VIVO	256	13245
Sapphire Atlantis ATI Radeon X800XT Platinum 256MB	256	13340
Sapphire Atlantis ATI Radeon X850 XT 256 MB	256	13923
Asus AX850PRO/HTVD 256MB, VIVO	256	14340
Gainward FX PowerPack Ultra/2600 TV-DVI 256MB	256	14530
Sapphire Atlantis ATI Radeon X850 XT 256 MB, VIVO, PCI-E	256	14542
Inno3D GeForce 6800 Ultra 256MB	256	14720
MicroStar NX6800Ultra-T2D256 256MB	256	14732
Asus AX800XT 256MB	256	15184
Asus AX800XT/TVD 256MB, VIVO	256	15303
Asus AX850XT/HTVD 256MB, VIVO	256	15637
Asus V9999 Ultra/2DT 256MB	256	15637
Sapphire Atlantis ATI Radeon X850 XT 256 MB, VIVO	256	16041
Gainward FX PowerPack Ultra/2600 Golden Sample 256MB	256	16101
Asus AX850XTP/HTVD 256MB, VIVO	256	16291
Asus V9999 Ultra Deluxe 256MB	256	16291
MicroStar NX6800Ultra-T2D256E 256MB, PCI-E	256	16648
Sapphire Atlantis ATI Radeon X850 XT Platinum 256 MB, VIVO	256	17552
Součty	24800	895740
Průměr		5633,585

Sy

Z tabulky je zřejma platnost vztahu $Sy = St + Sr$, protože

3227339182,60

Vzhledk k tomu, ze uvazujeme $n=159$ pozorovani a protoze regresni rovnice ma $p=3$ parametry, j

Y_i	$y_i - \bar{y}$	$Y_i - \bar{Y}$	$y_i - Y_i$	$(y_i - \bar{y})^2$	$(Y_i - \bar{Y})^2$
1426,452093	-4796,584906	-4207,132813	-589,4520926	23007226,76	17699966,51
1426,452093	-4748,584906	-4207,132813	-541,4520926	22549058,61	17699966,51
776,6296402	-4727,584906	-4856,955265	129,3703598	22350059,04	23590014,45
1426,452093	-4573,584906	-4207,132813	-366,4520926	20917678,89	17699966,51
1426,452093	-4506,584906	-4207,132813	-299,4520926	20309307,51	17699966,51
1426,452093	-4503,584906	-4207,132813	-296,4520926	20282277	17699966,51
1426,452093	-4486,584906	-4207,132813	-279,4520926	20129444,12	17699966,51
1426,452093	-4458,584906	-4207,132813	-251,4520926	19878979,36	17699966,51
1426,452093	-4447,584906	-4207,132813	-240,4520926	19781011,49	17699966,51
1426,452093	-4413,584906	-4207,132813	-206,4520926	19479731,72	17699966,51
1426,452093	-4398,584906	-4207,132813	-191,4520926	19347549,17	17699966,51
1426,452093	-4392,584906	-4207,132813	-185,4520926	19294802,15	17699966,51
1426,452093	-4343,584906	-4207,132813	-136,4520926	18866729,83	17699966,51
1426,452093	-4289,584906	-4207,132813	-82,45209259	18400538,66	17699966,51
1426,452093	-4289,584906	-4207,132813	-82,45209259	18400538,66	17699966,51
3618,100011	-4274,584906	-2015,484895	-2259,100011	18272076,12	4062179,362
3618,100011	-4176,584906	-2015,484895	-2161,100011	17443861,47	4062179,362
3618,100011	-4175,584906	-2015,484895	-2160,100011	17435509,3	4062179,362
3618,100011	-4116,584906	-2015,484895	-2101,100011	16946271,29	4062179,362
1426,452093	-4107,584906	-4207,132813	99,54790741	16872253,76	17699966,51
3618,100011	-4080,584906	-2015,484895	-2065,100011	16651173,17	4062179,362
3618,100011	-4028,584906	-2015,484895	-2013,100011	16229496,34	4062179,362
1426,452093	-3997,584906	-4207,132813	209,5479074	15980685,08	17699966,51
1426,452093	-3991,584906	-4207,132813	215,5479074	15932750,06	17699966,51
1426,452093	-3974,584906	-4207,132813	232,5479074	15797325,17	17699966,51
3618,100011	-3969,584906	-2015,484895	-1954,100011	15757604,32	4062179,362
3618,100011	-3965,584906	-2015,484895	-1950,100011	15725863,64	4062179,362
3618,100011	-3963,584906	-2015,484895	-1948,100011	15710005,3	4062179,362
3618,100011	-3931,584906	-2015,484895	-1916,100011	15457359,87	4062179,362
3618,100011	-3898,584906	-2015,484895	-1883,100011	15198964,27	4062179,362
3618,100011	-3880,584906	-2015,484895	-1865,100011	15058939,21	4062179,362
1426,452093	-3850,584906	-4207,132813	356,5479074	14827004,12	17699966,51
1426,452093	-3838,584906	-4207,132813	368,5479074	14734734,08	17699966,51
1426,452093	-3825,584906	-4207,132813	381,5479074	14635099,87	17699966,51
3618,100011	-3787,584906	-2015,484895	-1772,100011	14345799,42	4062179,362
1426,452093	-3649,584906	-4207,132813	557,5479074	13319469,98	17699966,51
3618,100011	-3643,584906	-2015,484895	-1628,100011	13275710,96	4062179,362
3618,100011	-3639,584906	-2015,484895	-1624,100011	13246578,29	4062179,362
3618,100011	-3575,584906	-2015,484895	-1560,100011	12784807,42	4062179,362
3618,100011	-3566,584906	-2015,484895	-1551,100011	12720527,89	4062179,362
3618,100011	-3539,584906	-2015,484895	-1524,100011	12528661,3	4062179,362
3618,100011	-3529,584906	-2015,484895	-1514,100011	12457969,61	4062179,362
3618,100011	-3514,584906	-2015,484895	-1499,100011	12352307,06	4062179,362
3618,100011	-3476,584906	-2015,484895	-1461,100011	12086642,61	4062179,362
3618,100011	-3347,584906	-2015,484895	-1332,100011	11206324,7	4062179,362
3618,100011	-3280,584906	-2015,484895	-1265,100011	10762237,32	4062179,362
3618,100011	-3240,584906	-2015,484895	-1225,100011	10501390,53	4062179,362
3618,100011	-3215,584906	-2015,484895	-1200,100011	10339986,29	4062179,362
3618,100011	-3189,584906	-2015,484895	-1174,100011	10173451,87	4062179,362
3618,100011	-3180,584906	-2015,484895	-1165,100011	10116120,34	4062179,362
1426,452093	-3105,584906	-4207,132813	1101,547907	9644657,606	17699966,51
3618,100011	-3051,584906	-2015,484895	-1036,100011	9312170,436	4062179,362
3618,100011	-2997,584906	-2015,484895	-982,1000106	8985515,267	4062179,362

3618,100011	-2877,584906	-2015,484895	-862,1000106	8280494,889	4062179,362
3618,100011	-2815,584906	-2015,484895	-800,1000106	7927518,361	4062179,362
3618,100011	-2685,584906	-2015,484895	-670,1000106	7212366,286	4062179,362
3618,100011	-2650,584906	-2015,484895	-635,1000106	7025600,342	4062179,362
3618,100011	-2593,584906	-2015,484895	-578,1000106	6726682,663	4062179,362
3618,100011	-2559,584906	-2015,484895	-544,1000106	6551474,889	4062179,362
3618,100011	-2549,584906	-2015,484895	-534,1000106	6500383,191	4062179,362
3618,100011	-2419,584906	-2015,484895	-404,1000106	5854391,116	4062179,362
3618,100011	-2397,584906	-2015,484895	-382,1000106	5748413,38	4062179,362
3618,100011	-2347,584906	-2015,484895	-332,1000106	5511154,889	4062179,362
3618,100011	-2322,584906	-2015,484895	-307,1000106	5394400,644	4062179,362
3618,100011	-2262,584906	-2015,484895	-247,1000106	5119290,455	4062179,362
3618,100011	-2231,584906	-2015,484895	-216,1000106	4979971,191	4062179,362
3618,100011	-2225,584906	-2015,484895	-210,1000106	4953228,172	4062179,362
3618,100011	-2194,584906	-2015,484895	-179,1000106	4816202,908	4062179,362
3618,100011	-2151,584906	-2015,484895	-136,1000106	4629317,606	4062179,362
3618,100011	-2131,584906	-2015,484895	-116,1000106	4543654,21	4062179,362
3618,100011	-2044,584906	-2015,484895	-29,10001063	4180327,436	4062179,362
3618,100011	-1964,584906	-2015,484895	50,89998937	3859593,852	4062179,362
3618,100011	-1848,584906	-2015,484895	166,8999894	3417266,153	4062179,362
3618,100011	-1819,584906	-2015,484895	195,8999894	3310889,229	4062179,362
3618,100011	-1798,584906	-2015,484895	216,8999894	3234907,663	4062179,362
3618,100011	-1773,584906	-2015,484895	241,8999894	3145603,418	4062179,362
3618,100011	-1748,584906	-2015,484895	266,8999894	3057549,172	4062179,362
3618,100011	-1744,584906	-2015,484895	270,8999894	3043576,493	4062179,362
3618,100011	-1711,584906	-2015,484895	303,8999894	2929522,889	4062179,362
3618,100011	-1668,584906	-2015,484895	346,8999894	2784175,587	4062179,362
3618,100011	-1587,584906	-2015,484895	427,8999894	2520425,833	4062179,362
11569,4079	-1552,584906	5935,822994	-7488,4079	2410519,889	35233994,62
3618,100011	-1527,584906	-2015,484895	487,8999894	2333515,644	4062179,362
3618,100011	-1447,584906	-2015,484895	567,8999894	2095502,059	4062179,362
3618,100011	-1431,584906	-2015,484895	583,8999894	2049435,342	4062179,362
3618,100011	-1357,584906	-2015,484895	657,8999894	1843036,776	4062179,362
3618,100011	-1260,584906	-2015,484895	754,8999894	1589074,304	4062179,362
3618,100011	-1223,584906	-2015,484895	791,8999894	1497160,021	4062179,362
3618,100011	-1209,584906	-2015,484895	805,8999894	1463095,644	4062179,362
3618,100011	-1205,584906	-2015,484895	809,8999894	1453434,965	4062179,362
3618,100011	-1091,584906	-2015,484895	923,8999894	1191557,606	4062179,362
11569,4079	-1086,584906	5935,822994	-7022,4079	1180666,757	35233994,62
3618,100011	-1078,584906	-2015,484895	936,8999894	1163345,399	4062179,362
3618,100011	-1048,584906	-2015,484895	966,8999894	1099530,304	4062179,362
3618,100011	-913,5849057	-2015,484895	1101,899989	834637,3799	4062179,362
3618,100011	-778,5849057	-2015,484895	1236,899989	606194,4553	4062179,362
3618,100011	-764,5849057	-2015,484895	1250,899989	584590,078	4062179,362
11569,4079	-646,5849057	5935,822994	-6582,4079	418072,0402	35233994,62
3618,100011	-503,5849057	-2015,484895	1511,899989	253597,7572	4062179,362
3618,100011	-255,5849057	-2015,484895	1759,899989	65323,644	4062179,362
3618,100011	-253,5849057	-2015,484895	1761,899989	64305,30438	4062179,362
3618,100011	-89,58490566	-2015,484895	1925,899989	8025,455322	4062179,362
3618,100011	-78,58490566	-2015,484895	1936,899989	6175,587398	4062179,362
11569,4079	-6,58490566	5935,822994	-5942,4079	43,36098256	35233994,62
3618,100011	35,41509434	-2015,484895	2050,899989	1254,228907	4062179,362
3618,100011	203,4150943	-2015,484895	2218,899989	41377,70061	4062179,362
3618,100011	281,4150943	-2015,484895	2296,899989	79194,45532	4062179,362
3618,100011	372,4150943	-2015,484895	2387,899989	138693,0025	4062179,362
3618,100011	410,4150943	-2015,484895	2425,899989	168440,5497	4062179,362

3618,100011	681,4150943	-2015,484895	2696,899989	464326,5308	4062179,362
3618,100011	700,4150943	-2015,484895	2715,899989	490581,3044	4062179,362
3618,100011	715,4150943	-2015,484895	2730,899989	511818,7572	4062179,362
3618,100011	767,4150943	-2015,484895	2782,899989	588925,927	4062179,362
3618,100011	782,4150943	-2015,484895	2797,899989	612173,3799	4062179,362
11569,4079	911,4150943	5935,822994	-5024,4079	830677,4742	35233994,62
3618,100011	924,4150943	-2015,484895	2939,899989	854543,2666	4062179,362
3618,100011	1762,415094	-2015,484895	3777,899989	3106106,965	4062179,362
11569,4079	1762,415094	5935,822994	-4173,4079	3106106,965	35233994,62
11569,4079	2050,415094	5935,822994	-3885,4079	4204202,059	35233994,62
11569,4079	2171,415094	5935,822994	-3764,4079	4715043,512	35233994,62
11569,4079	2464,415094	5935,822994	-3471,4079	6073341,757	35233994,62
11569,4079	2657,415094	5935,822994	-3278,4079	7061854,984	35233994,62
11569,4079	3254,415094	5935,822994	-2681,4079	10591217,61	35233994,62
11569,4079	3528,415094	5935,822994	-2407,4079	12449713,08	35233994,62
11569,4079	3872,415094	5935,822994	-2063,4079	14995598,66	35233994,62
11569,4079	3969,415094	5935,822994	-1966,4079	15756256,19	35233994,62
11569,4079	4130,415094	5935,822994	-1805,4079	17060328,85	35233994,62
11569,4079	4139,415094	5935,822994	-1796,4079	17134757,32	35233994,62
11569,4079	4238,415094	5935,822994	-1697,4079	17964162,51	35233994,62
11569,4079	5036,415094	5935,822994	-899,4078999	25365477	35233994,62
11569,4079	5100,415094	5935,822994	-835,4078999	26014234,13	35233994,62
11569,4079	5340,415094	5935,822994	-595,4078999	28520033,38	35233994,62
11569,4079	5437,415094	5935,822994	-498,4078999	29565482,91	35233994,62
11569,4079	5471,415094	5935,822994	-464,4078999	29936383,13	35233994,62
11569,4079	5493,415094	5935,822994	-442,4078999	30177609,4	35233994,62
11569,4079	5650,415094	5935,822994	-285,4078999	31927190,74	35233994,62
11569,4079	5767,415094	5935,822994	-168,4078999	33263076,87	35233994,62
11569,4079	6134,415094	5935,822994	198,5921001	37631048,55	35233994,62
11569,4079	6564,415094	5935,822994	628,5921001	43091545,53	35233994,62
11569,4079	7063,415094	5935,822994	1127,5921	49891832,79	35233994,62
11569,4079	7397,415094	5935,822994	1461,5921	54721750,08	35233994,62
11569,4079	7611,415094	5935,822994	1675,5921	57933639,74	35233994,62
11569,4079	7706,415094	5935,822994	1770,5921	59388833,61	35233994,62
11569,4079	8289,415094	5935,822994	2353,5921	68714402,61	35233994,62
11569,4079	8706,415094	5935,822994	2770,5921	75801663,79	35233994,62
11569,4079	8896,415094	5935,822994	2960,5921	79146201,53	35233994,62
11569,4079	8908,415094	5935,822994	2972,5921	79359859,49	35233994,62
11569,4079	9086,415094	5935,822994	3150,5921	82562939,27	35233994,62
11569,4079	9098,415094	5935,822994	3162,5921	82781157,23	35233994,62
11569,4079	9550,415094	5935,822994	3614,5921	91210428,47	35233994,62
11569,4079	9669,415094	5935,822994	3733,5921	93497588,27	35233994,62
11569,4079	10003,41509	5935,822994	4067,5921	100068313,5	35233994,62
11569,4079	10003,41509	5935,822994	4067,5921	100068313,5	35233994,62
11569,4079	10407,41509	5935,822994	4471,5921	108314288,9	35233994,62
11569,4079	10467,41509	5935,822994	4531,5921	109566778,8	35233994,62
11569,4079	10657,41509	5935,822994	4721,5921	113580496,5	35233994,62
11569,4079	10657,41509	5935,822994	4721,5921	113580496,5	35233994,62
11569,4079	11014,41509	5935,822994	5078,5921	121317339,9	35233994,62
11569,4079	11918,41509	5935,822994	5982,5921	142048618,4	35233994,62
895740	0,00	0,00	0,00	3227339182,60	2444158775,06

St	Sr	Vysledek
2444158775,06	783180407,55	3227339182,60

je rezidualni rozptyl

$$S_R^2 = \frac{S_R}{n-p}$$

5020387,228

Determinacni index je

0,757329378

$$(y_i - Y_i)^2$$

347453,7695
293170,3686
16736,68999
134287,1362
89671,55576
87883,8432
78093,47205
63228,15487
57817,20883
42622,46654
36653,90376
34392,47865
18619,17357
6798,347573
6798,347573
5103532,858
4670353,256
4666032,056
4414621,255
9909,785869
4264638,054
4052571,653
43910,3255
46460,90039
54078,52924
3818506,852
3802890,051
3795093,651
3671439,251
3546065,65
3478598,05
127126,4103
135827,5601
145578,8056
3140338,448
310859,6691
2650709,645
2637700,845
2433912,043
2405911,243
2322880,842
2292498,842
2247300,842
2134813,241
1774490,438
1600478,037
1500870,036
1440240,036
1378510,835
1357458,035
1213407,792
1073503,232
964520,4309

743216,4283
640160,027
449034,0242
403352,0235
334199,6223
296044,8216
285262,8214
163296,8186
146000,4181
110290,4171
94310,41653
61058,41525
46699,21459
44142,01447
32076,81381
18523,21289
13479,21247
846,8106185
2590,808918
27855,60645
38376,80584
47045,60539
58515,60486
71235,60433
73386,80424
92355,20354
120339,6026
183098,4009
56076252,88
238046,3996
322510,3979
340939,1976
432832,396
569873,994
627105,5932
649474,7929
655937,9928
853591,1904
49314212,71
877781,5901
934895,5894
1214183,587
1529921,584
1564750,783
43328093,76
2285841,578
3097247,973
3104291,573
3709090,769
3751581,569
35312211,65
4206190,766
4923517,163
5275749,561
5702066,359
5884990,758

7273269,553
7376112,752
7457814,752
7744532,351
7828244,351
25244674,74
8643011,948
14272528,33
17417333,5
15096394,55
14170766,84
12050672,81
10747958,36
7189948,326
5795612,796
4257652,161
3866760,029
3259497,685
3227081,343
2881193,579
808934,5704
697906,3592
354510,5673
248410,4347
215674,6975
195724,7499
81457,66932
28361,22075
39438,82222
395128,0283
1271463,944
2136251,467
2807608,886
3134996,385
5539395,774
7676180,585
8765105,583
8836303,794
9926230,581
10001988,79
13065276,05
13939709,97
16545305,49
16545305,49
19995135,91
20535326,96
22293431,96
22293431,96
25792097,72
35791408,24
783180407,55

Název	x_i	y_i
Asus A7000/T 64MB	150	837
Inno3D GeForce2 MX400 64MB	200	885
MicroStar MX4000-T64 64MB	250	906
Asus A9200SE/T 64MB	200	1060
Sapphire Atlantis ATI Radeon 9200 SE 128MB	250	1127
MicroStar MX4000-T128 128MB	250	1130
Asus V9400 Magic 128MB	250	1147
GigaByte MAYA GV-N52128D 128MB	250	1175
Sapphire Atlantis ATI Radeon 9250 64MB	240	1186
Club3D ATI 9250 128MB(64bit)	240	1220
Sapphire Atlantis ATI Radeon 9250 128MB(64bit)	240	1235
Asus A9200SET 128MB	200	1241
Asus A9250/TD 128MB	240	1290
Asus V9520 Magic 128MB	250	1344
Asus V9520-X/TD 128MB	250	1344
Sapphire Atlantis ATI Radeon 9250 128MB	240	1359
GigaByte MAYA GV-R925128D 128MB	240	1457
Club3D ATI 9250 128MB(128bit)	240	1458
Sapphire Atlantis ATI Radeon 9250 128MB VIVO	240	1517
GigaByte MAYA GV-R955128T 128MB OEM	250	1526
Inno3D GeForce FX 5200 128MB	250	1553
Asus A9250GE/TD 128MB	240	1605
Gainward FX PowerPack Pro/665 TV/DVI 128MB	270	1636
MicroStar RX300SE-TD128E 128MB, PCI-E	325	1642
MicroStar FX5200-TD128 128MB	250	1659
Sapphire Atlantis ATI Radeon 9250 256MB	240	1664
MicroStar NX6200TC-TD64E 64MB, PCI-E	350	1668
GigaByte MAYA GV-R925128VH 128MB VIVO OEM	240	1670
Club3D ATI 9250 128MB(128bit) VIVO	240	1702
Asus V9520/TD 128MB	250	1735
Inno3D GeForce FX-5500 128MB	270	1753
MicroStar FX5500-T128 128MB	200	1783
Asus A9550/TD 128MB	250	1795
Sapphire Atlantis ATI Radeon 9600SE 128MB	325	1808
Sapphire Atlantis ATI Radeon 9550 128MB	250	1846
MicroStar RX300LE-TD128E	325	1984
Club3D FX5200 256MB	250	1990
Asus A9550GE/TD 128MB	250	1994
Club3D ATI 9550 256MB(128bit)	250	2058
Sapphire Atlantis ATI Radeon 9550 128MB VIVO	250	2067
Asus A9250GE/TD 256MB	240	2094
GigaByte MAYA GV-N55128DP 128MB	300	2104
Inno3D GeForce FX-5500 256MB	270	2119
Sapphire Atlantis ATI Radeon 9550 256MB	250	2157
Inno3D GeForce FX-5500 256MB, VIVO	270	2286
Club3D ATI 9600 128MB	400	2353
MicroStar NX6200-TD128E 128MB, PCI-E	300	2393
Sapphire Atlantis ATI Radeon 9600 128MB	325	2418
MicroStar NX6200-TD128 128MB	300	2444
Asus A9600GE/TD 128MB	325	2453
Asus V9570LE/TD 128MB	250	2528
Asus A9550GE/TD 256MB	250	2582
Abit Radeon 9550-256CDT 256MB	250	2636

Asus V9570GE/TD 128MB	250	2756
Sapphire Atlantis ATI Radeon X600Pro 128MB, PCI-E	400	2818
Club3D ATI 9600 256MB	400	2948
Asus V9520 VideoSuite 128MB	250	2983
Sapphire Atlantis ATI Radeon X600Pro 256MB, PCI-E	400	3040
Club3D ATI 9600PRO 128MB	400	3074
Inno3D GeForce 6200 128MB	300	3084
MicroStar RX600XT-TD128E II 128MB, PCI-E	500	3214
Asus N6200GE/TD 128MB	300	3236
Sapphire Atlantis ATI Radeon 9600XT 128MB	500	3286
MicroStar FX5700LE-TD256 256MB	250	3311
Asus V9570GE/TD 256MB	250	3371
Sapphire Atlantis ATI Radeon 9600XT 128MB VIVO	500	3402
Sapphire Atlantis ATI Radeon X700 128MB, PCI-E	400	3408
Sapphire Atlantis ATI Radeon 9800SE 128MB	325	3439
Inno3D GeForce 6600 128MB, AGP	300	3482
Sapphire Atlantis ATI Radeon 9600XT 256MB	500	3502
Gainward FX PowerPack Ultra/1760 XP TV-DVI 128MB	300	3589
Sapphire Atlantis ATI Radeon 9600XT 256MB VIVO	500	3669
Sapphire Atlantis ATI Radeon X700 128MB	400	3785
Inno3D GeForce 6600 256MB, AGP	300	3814
Club3D ATI 9600XT 256MB	500	3835
Gainward FX PowerPack Ultra/1760 XP Golden Sample 128MB	400	3860
MicroStar NX6600-TD128E 128MB, PCI-E	300	3885
Asus N6600/TD 128MB	300	3889
Asus V9570 TD 128MB	425	3922
Gainward FX PowerPack Ultra/1780 TV/DVI 256MB	300	3965
Club3D ATI 9600XT 128MB	500	4046
Sapphire Atlantis ATI Radeon 9800Pro 128MB	380	4081
Sapphire Atlantis ATI Radeon X700 256MB, PCI-E	400	4106
MicroStar NX6600-TD256E 256MB, PCI-E	300	4186
Sapphire Atlantis ATI Radeon X700 256MB	400	4202
MicroStar RX700 PRO-TD128E 128MB, PCI-E	420	4276
Asus V9570/TD 256MB	425	4373
Asus A9600XT 128MB	500	4410
GigaByte MAYA GV-R96X128D 128MB	500	4424
Sapphire Atlantis ATI Radeon X700 Fireblade 256MB	400	4428
Asus N6600/TD 256MB	300	4542
Club3D All-In-Wonder 9800SE 128MB	380	4547
MicroStar NX6600-VTD256EH 256MB, VIVO, PCI-E	300	4555
GigaByte MAYA GV-R96X128DU 128MB	550	4585
Sapphire Atlantis ATI Radeon X700PRO 256MB, PCI-E	420	4720
MicroStar NX6600-VTD256 256MB	400	4855
Sapphire Atlantis ATI Radeon X700PRO 128MB	420	4869
Club3D ATI 9800Pro 256bit 128MB	380	4987
MicroStar NX6600-VTD128E Diamond 128MB, VIVO, PCI-E	400	5130
Inno3D GeForce 6600GT 128MB, AGP	500	5378
Gainward FX PowerPack Ultra/1960 TV-DVI 128MB	500	5380
Inno3D GeForce 6600PE 128MB, AGP	400	5544
GigaByte MAYA GV-N66T128D 128MB	500	5555
Asus V9999LE/TD 128MB	300	5627
MicroStar NX6600GT-TD128E 128MB, PCI-E	500	5669
Sapphire Atlantis ATI Radeon X700PRO 256MB	420	5837
MicroStar NX6600-VTD128 Diamond	400	5915
MicroStar NX6600GT-VTD128 128MB	500	6006
Asus N6600GT/TD 128MB	500	6044

MicroStar RX700 PRO-TD256E 256MB, PCI-E	420	6315
GigaByte MAYA GV-N66T128VP 128MB	500	6334
Gainward FX PowerPack Ultra/1960 TV-DVI-DVI XP 256MB	500	6349
Gainward FX PowerPack Ultra/1960 XP 128MB, AGP	500	6401
Asus N6600GT TOP 128MB	520	6416
MicroStar NX6800LE-TD128 128MB	300	6545
Gainward FX PowerPack Ultra/1960 XP Golden Sample 128MB, A	540	6558
Gainward FX PowerPack Ultra/1960 XP 128MB Golden Sample G	540	7396
Gainward FX PowerPack Ultra/2100 TV/DVI 128MB	325	7396
MicroStar NX6800-TD128 128MB	325	7684
Inno3D GeForce 6800PE 128MB	350	7805
Leadtek Winfast A400 TDH 128MB	325	8098
Asus V9999/TD 128MB	325	8291
Gainward FX PowerPack Ultra/2100 TV/DVI Golden Sample 128M	350	8888
Sapphire Atlantis ATI Radeon X800Pro 256MB	475	9162
MicroStar NX6800-TD256E 256MB, PCI-E	325	9506
Sapphire Atlantis ATI Radeon X800 Pro 256 MB, PCI-E	475	9603
Asus V9999GT/TD 128MB	350	9764
Sapphire Atlantis ATI Radeon X800 XL 256 MB, PCI-E	400	9773
Excibur X800Pro IceQ II 256MB	475	9872
Asus V9999GT/TD 256MB	350	10670
Inno3D GeForce 6800GT 256MB	350	10734
Sapphire Atlantis ATI Radeon X800 XL Ultimate 256MB	400	10974
Asus AX800XL/2DTV 256MB	400	11071
MicroStar RX800 PRO VTD 256MB VIVO	475	11105
MicroStar NX6800GT-TD256 256MB	350	11127
Gainward FX PowerPack Ultra/2400 TV/DVI Golden Sample 256M	400	11284
Club3D ATI X800 Pro 256MB VIVO	475	11401
Gainward FX PowerPack Ultra/2400 TV/DVI 256MB	350	11768
Club3D ATI X800 Pro 256MB	475	12198
Asus AX800 PRO/TD 256MB	475	12697
Sapphire Atlantis ATI Radeon X800XT 256 MB VIVO	500	13031
Club3D ATI X800 XT 256MB VIVO	500	13245
Sapphire Atlantis ATI Radeon X800XT Platinum 256MB	520	13340
Sapphire Atlantis ATI Radeon X850 XT 256 MB	520	13923
Asus AX850PRO/HTVD 256MB, VIVO	520	14340
Gainward FX PowerPack Ultra/2600 TV-DVI 256MB	400	14530
Sapphire Atlantis ATI Radeon X850 XT 256 MB, VIVO, PCI-E	520	14542
Inno3D GeForce 6800 Ultra 256MB	400	14720
MicroStar NX6800Ultra-T2D256 256MB	400	14732
Asus AX800XT 256MB	520	15184
Asus AX800XT/TVD 256MB, VIVO	520	15303
Asus AX850XT/HTVD 256MB, VIVO	520	15637
Asus V9999 Ultra/2DT 256MB	425	15637
Sapphire Atlantis ATI Radeon X850 XT 256 MB, VIVO	520	16041
Gainward FX PowerPack Ultra/2600 Golden Sample 256MB	430	16101
Asus AX850XTP/HTVD 256MB, VIVO	540	16291
Asus V9999 Ultra Deluxe 256MB	400	16291
MicroStar NX6800Ultra-T2D256E 256MB, PCI-E	400	16648
Sapphire Atlantis ATI Radeon X850 XT Platinum 256 MB, VIVO	540	17552
Součty	57835	895740
Průměr		5633,585

Sy

Z tabulky je zřejma platnost vztahu $Sy = St + Sr$, protože

3227339182,60

Vzhledk k tomu, ze uvazujeme $n=159$ pozorovani a protoze regresni rovnice ma $p=3$ parametry, j

Y_i	$y_i - \bar{y}$	$Y_i - \bar{Y}$	$y_i - Y_i$	$(y_i - \bar{y})^2$	$(Y_i - \bar{Y})^2$
-2127,830218	-4796,584906	-7761,415124	2964,830218	23007226,76	60239564,73
204,3333758	-4748,584906	-5429,25153	680,6666242	22549058,61	29476772,17
2294,650547	-4727,584906	-3338,934358	-1388,650547	22350059,04	11148482,65
204,3333758	-4573,584906	-5429,25153	855,6666242	20917678,89	29476772,17
2294,650547	-4506,584906	-3338,934358	-1167,650547	20309307,51	11148482,65
2294,650547	-4503,584906	-3338,934358	-1164,650547	20282277	11148482,65
2294,650547	-4486,584906	-3338,934358	-1147,650547	20129444,12	11148482,65
2294,650547	-4458,584906	-3338,934358	-1119,650547	19878979,36	11148482,65
1895,934827	-4447,584906	-3737,650079	-709,9348268	19781011,49	13970028,11
1895,934827	-4413,584906	-3737,650079	-675,9348268	19479731,72	13970028,11
1895,934827	-4398,584906	-3737,650079	-660,9348268	19347549,17	13970028,11
204,3333758	-4392,584906	-5429,25153	1036,666624	19294802,15	29476772,17
1895,934827	-4343,584906	-3737,650079	-605,9348268	18866729,83	13970028,11
2294,650547	-4289,584906	-3338,934358	-950,6505473	18400538,66	11148482,65
2294,650547	-4289,584906	-3338,934358	-950,6505473	18400538,66	11148482,65
1895,934827	-4274,584906	-3737,650079	-536,9348268	18272076,12	13970028,11
1895,934827	-4176,584906	-3737,650079	-438,9348268	17443861,47	13970028,11
1895,934827	-4175,584906	-3737,650079	-437,9348268	17435509,3	13970028,11
1895,934827	-4116,584906	-3737,650079	-378,9348268	16946271,29	13970028,11
2294,650547	-4107,584906	-3338,934358	-768,6505473	16872253,76	11148482,65
2294,650547	-4080,584906	-3338,934358	-741,6505473	16651173,17	11148482,65
1895,934827	-4028,584906	-3737,650079	-290,9348268	16229496,34	13970028,11
3063,060418	-3997,584906	-2570,524488	-1427,060418	15980685,08	6607596,144
4976,664262	-3991,584906	-656,9206436	-3334,664262	15932750,06	431544,7319
2294,650547	-3974,584906	-3338,934358	-635,6505473	15797325,17	11148482,65
1895,934827	-3969,584906	-3737,650079	-231,9348268	15757604,32	13970028,11
5749,745622	-3965,584906	116,1607167	-4081,745622	15725863,64	13493,31211
1895,934827	-3963,584906	-3737,650079	-225,9348268	15710005,3	13970028,11
1895,934827	-3931,584906	-3737,650079	-193,9348268	15457359,87	13970028,11
2294,650547	-3898,584906	-3338,934358	-559,6505473	15198964,27	11148482,65
3063,060418	-3880,584906	-2570,524488	-1310,060418	15058939,21	6607596,144
204,3333758	-3850,584906	-5429,25153	1578,666624	14827004,12	29476772,17
2294,650547	-3838,584906	-3338,934358	-499,6505473	14734734,08	11148482,65
4976,664262	-3825,584906	-656,9206436	-3168,664262	14635099,87	431544,7319
2294,650547	-3787,584906	-3338,934358	-448,6505473	14345799,42	11148482,65
4976,664262	-3649,584906	-656,9206436	-2992,664262	13319469,98	431544,7319
2294,650547	-3643,584906	-3338,934358	-304,6505473	13275710,96	11148482,65
2294,650547	-3639,584906	-3338,934358	-300,6505473	13246578,29	11148482,65
2294,650547	-3575,584906	-3338,934358	-236,6505473	12784807,42	11148482,65
2294,650547	-3566,584906	-3338,934358	-227,6505473	12720527,89	11148482,65
1895,934827	-3539,584906	-3737,650079	198,0651732	12528661,3	13970028,11
4143,121296	-3529,584906	-1490,46361	-2039,121296	12457969,61	2221481,771
3063,060418	-3514,584906	-2570,524488	-944,0604175	12352307,06	6607596,144
2294,650547	-3476,584906	-3338,934358	-137,6505473	12086642,61	11148482,65
3063,060418	-3347,584906	-2570,524488	-777,0604175	11206324,7	6607596,144
7114,523526	-3280,584906	1480,93862	-4761,523526	10762237,32	2193179,197
4143,121296	-3240,584906	-1490,46361	-1750,121296	10501390,53	2221481,771
4976,664262	-3215,584906	-656,9206436	-2558,664262	10339986,29	431544,7319
4143,121296	-3189,584906	-1490,46361	-1699,121296	10173451,87	2221481,771
4976,664262	-3180,584906	-656,9206436	-2523,664262	10116120,34	431544,7319
2294,650547	-3105,584906	-3338,934358	233,3494527	9644657,606	11148482,65
2294,650547	-3051,584906	-3338,934358	287,3494527	9312170,436	11148482,65
2294,650547	-2997,584906	-3338,934358	341,3494527	8985515,267	11148482,65

2294,650547	-2877,584906	-3338,934358	461,3494527	8280494,889	11148482,65
7114,523526	-2815,584906	1480,93862	-4296,523526	7927518,361	2193179,197
7114,523526	-2685,584906	1480,93862	-4166,523526	7212366,286	2193179,197
2294,650547	-2650,584906	-3338,934358	688,3494527	7025600,342	11148482,65
7114,523526	-2593,584906	1480,93862	-4074,523526	6726682,663	2193179,197
7114,523526	-2559,584906	1480,93862	-4040,523526	6551474,889	2193179,197
4143,121296	-2549,584906	-1490,46361	-1059,121296	6500383,191	2221481,771
9118,540065	-2419,584906	3484,95516	-5904,540065	5854391,116	12144912,47
4143,121296	-2397,584906	-1490,46361	-907,1212961	5748413,38	2221481,771
9118,540065	-2347,584906	3484,95516	-5832,540065	5511154,889	12144912,47
2294,650547	-2322,584906	-3338,934358	1016,349453	5394400,644	11148482,65
2294,650547	-2262,584906	-3338,934358	1076,349453	5119290,455	11148482,65
9118,540065	-2231,584906	3484,95516	-5716,540065	4979971,191	12144912,47
7114,523526	-2225,584906	1480,93862	-3706,523526	4953228,172	2193179,197
4976,664262	-2194,584906	-656,9206436	-1537,664262	4816202,908	431544,7319
4143,121296	-2151,584906	-1490,46361	-661,1212961	4629317,606	2221481,771
9118,540065	-2131,584906	3484,95516	-5616,540065	4543654,21	12144912,47
4143,121296	-2044,584906	-1490,46361	-554,1212961	4180327,436	2221481,771
9118,540065	-1964,584906	3484,95516	-5449,540065	3859593,852	12144912,47
7114,523526	-1848,584906	1480,93862	-3329,523526	3417266,153	2193179,197
4143,121296	-1819,584906	-1490,46361	-329,1212961	3310889,229	2221481,771
9118,540065	-1798,584906	3484,95516	-5283,540065	3234907,663	12144912,47
7114,523526	-1773,584906	1480,93862	-3254,523526	3145603,418	2193179,197
4143,121296	-1748,584906	-1490,46361	-258,1212961	3057549,172	2221481,771
4143,121296	-1744,584906	-1490,46361	-254,1212961	3043576,493	2221481,771
7706,220069	-1711,584906	2072,635164	-3784,220069	2929522,889	4295816,522
4143,121296	-1668,584906	-1490,46361	-178,1212961	2784175,587	2221481,771
9118,540065	-1587,584906	3484,95516	-5072,540065	2520425,833	12144912,47
6597,633935	-1552,584906	964,0490296	-2516,633935	2410519,889	929390,5315
7114,523526	-1527,584906	1480,93862	-3008,523526	2333515,644	2193179,197
4143,121296	-1447,584906	-1490,46361	42,87870386	2095502,059	2221481,771
7114,523526	-1431,584906	1480,93862	-2912,523526	2049435,342	2193179,197
7592,717689	-1357,584906	1959,132783	-3316,717689	1843036,776	3838201,263
7706,220069	-1260,584906	2072,635164	-3333,220069	1589074,304	4295816,522
9118,540065	-1223,584906	3484,95516	-4708,540065	1497160,021	12144912,47
9118,540065	-1209,584906	3484,95516	-4694,540065	1463095,644	12144912,47
7114,523526	-1205,584906	1480,93862	-2686,523526	1453434,965	2193179,197
4143,121296	-1091,584906	-1490,46361	398,8787039	1191557,606	2221481,771
6597,633935	-1086,584906	964,0490296	-2050,633935	1180666,757	929390,5315
4143,121296	-1078,584906	-1490,46361	411,8787039	1163345,399	2221481,771
9757,778701	-1048,584906	4124,193795	-5172,778701	1099530,304	17008974,46
7592,717689	-913,5849057	1959,132783	-2872,717689	834637,3799	3838201,263
7114,523526	-778,5849057	1480,93862	-2259,523526	606194,4553	2193179,197
7592,717689	-764,5849057	1959,132783	-2723,717689	584590,078	3838201,263
6597,633935	-646,5849057	964,0490296	-1610,633935	418072,0402	929390,5315
7114,523526	-503,5849057	1480,93862	-1984,523526	253597,7572	2193179,197
9118,540065	-255,5849057	3484,95516	-3740,540065	65323,644	12144912,47
9118,540065	-253,5849057	3484,95516	-3738,540065	64305,30438	12144912,47
7114,523526	-89,58490566	1480,93862	-1570,523526	8025,455322	2193179,197
9118,540065	-78,58490566	3484,95516	-3563,540065	6175,587398	12144912,47
4143,121296	-6,58490566	-1490,46361	1483,878704	43,36098256	2221481,771
9118,540065	35,41509434	3484,95516	-3449,540065	1254,228907	12144912,47
7592,717689	203,4150943	1959,132783	-1755,717689	41377,70061	3838201,263
7114,523526	281,4150943	1480,93862	-1199,523526	79194,45532	2193179,197
9118,540065	372,4150943	3484,95516	-3112,540065	138693,0025	12144912,47
9118,540065	410,4150943	3484,95516	-3074,540065	168440,5497	12144912,47

7592,717689	681,4150943	1959,132783	-1277,717689	464326,5308	3838201,263
9118,540065	700,4150943	3484,95516	-2784,540065	490581,3044	12144912,47
9118,540065	715,4150943	3484,95516	-2769,540065	511818,7572	12144912,47
9118,540065	767,4150943	3484,95516	-2717,540065	588925,927	12144912,47
9403,25709	782,4150943	3769,672185	-2987,25709	612173,3799	14210428,38
4143,121296	911,4150943	-1490,46361	2401,878704	830677,4742	2221481,771
9649,278688	924,4150943	4015,693782	-3091,278688	854543,2666	16125796,55
9649,278688	1762,415094	4015,693782	-2253,278688	3106106,965	16125796,55
4976,664262	1762,415094	-656,9206436	2419,335738	3106106,965	431544,7319
4976,664262	2050,415094	-656,9206436	2707,335738	4204202,059	431544,7319
5749,745622	2171,415094	116,1607167	2055,254378	4715043,512	13493,31211
4976,664262	2464,415094	-656,9206436	3121,335738	6073341,757	431544,7319
4976,664262	2657,415094	-656,9206436	3314,335738	7061854,984	431544,7319
5749,745622	3254,415094	116,1607167	3138,254378	10591217,61	13493,31211
8708,228339	3528,415094	3074,643433	453,771661	12449713,08	9453432,242
4976,664262	3872,415094	-656,9206436	4529,335738	14995598,66	431544,7319
8708,228339	3969,415094	3074,643433	894,771661	15756256,19	9453432,242
5749,745622	4130,415094	116,1607167	4014,254378	17060328,85	13493,31211
7114,523526	4139,415094	1480,93862	2658,476474	17134757,32	2193179,197
8708,228339	4238,415094	3074,643433	1163,771661	17964162,51	9453432,242
5749,745622	5036,415094	116,1607167	4920,254378	25365477	13493,31211
5749,745622	5100,415094	116,1607167	4984,254378	26014234,13	13493,31211
7114,523526	5340,415094	1480,93862	3859,476474	28520033,38	2193179,197
7114,523526	5437,415094	1480,93862	3956,476474	29565482,91	2193179,197
8708,228339	5471,415094	3074,643433	2396,771661	29936383,13	9453432,242
5749,745622	5493,415094	116,1607167	5377,254378	30177609,4	13493,31211
7114,523526	5650,415094	1480,93862	4169,476474	31927190,74	2193179,197
8708,228339	5767,415094	3074,643433	2692,771661	33263076,87	9453432,242
5749,745622	6134,415094	116,1607167	6018,254378	37631048,55	13493,31211
8708,228339	6564,415094	3074,643433	3489,771661	43091545,53	9453432,242
8708,228339	7063,415094	3074,643433	3988,771661	49891832,79	9453432,242
9118,540065	7397,415094	3484,95516	3912,459935	54721750,08	12144912,47
9118,540065	7611,415094	3484,95516	4126,459935	57933639,74	12144912,47
9403,25709	7706,415094	3769,672185	3936,74291	59388833,61	14210428,38
9403,25709	8289,415094	3769,672185	4519,74291	68714402,61	14210428,38
9403,25709	8706,415094	3769,672185	4936,74291	75801663,79	14210428,38
7114,523526	8896,415094	1480,93862	7415,476474	79146201,53	2193179,197
9403,25709	8908,415094	3769,672185	5138,74291	79359859,49	14210428,38
7114,523526	9086,415094	1480,93862	7605,476474	82562939,27	2193179,197
7114,523526	9098,415094	1480,93862	7617,476474	82781157,23	2193179,197
9403,25709	9550,415094	3769,672185	5780,74291	91210428,47	14210428,38
9403,25709	9669,415094	3769,672185	5899,74291	93497588,27	14210428,38
9403,25709	10003,41509	3769,672185	6233,74291	100068313,5	14210428,38
7706,220069	10003,41509	2072,635164	7930,779931	100068313,5	4295816,522
9403,25709	10407,41509	3769,672185	6637,74291	108314288,9	14210428,38
7817,303985	10467,41509	2183,71908	8283,696015	109566778,8	4768629,019
9649,278688	10657,41509	4015,693782	6641,721312	113580496,5	16125796,55
7114,523526	10657,41509	1480,93862	9176,476474	113580496,5	2193179,197
7114,523526	11014,41509	1480,93862	9533,476474	121317339,9	2193179,197
9649,278688	11918,41509	4015,693782	7902,721312	142048618,4	16125796,55
895740	0,00	0,00	0,00	3227339182,60	1316047964,90

St Sr Vysledek
1316047964,90 1911291217,70 3227339182,60

je rezidualni rozptyl

$$S_R^2 = \frac{S_R}{n - p}$$

12251866,78

Determinacni index je

0,407781113

$$(y_i - Y_i)^2$$

8790218,223
463307,0533
1928350,342
732165,3718
1363407,801
1356410,897
1317101,779
1253617,348
504007,4583
456887,8901
436834,8453
1074677,69
367157,0143
903736,463
903736,463
288299,0082
192663,7822
191786,9125
143591,603
590823,6638
550045,5343
84643,07344
2036501,435
11119985,74
404051,6183
53793,76388
16660647,33
51046,54596
37610,71704
313208,7351
1716258,298
2492188,31
249650,6694
10040433,21
201287,3136
8956039,386
92811,95596
90390,75158
56003,48153
51824,77168
39229,81284
4158015,66
891250,072
18947,67317
603822,8925
22672106,29
3062924,551
6546762,806
2887013,179
6368881,308
54451,96709
82569,70798
116519,4489

212843,3175
18460114,41
17359918,29
473824,9691
16601741,96
16325830,36
1121737,92
34863593,38
822869,0459
34018523,61
1032966,21
1158528,144
32678830,32
13738316,65
2364411,383
437081,3682
31545522,31
307050,4108
29697486,92
11085726,91
108320,8276
27915795,62
10591923,38
66626,60352
64577,63315
14320321,53
31727,19614
25730662,71
6333446,364
9051213,806
1838,583245
8482793,29
11000616,23
11110356,03
22170349,55
22038706,43
7217408,656
159104,2204
4205099,536
169644,0667
26757639,49
8252506,921
5105446,565
7418638,05
2594141,673
3938333,625
13991639,98
13976681,82
2466544,146
12698817,8
2201896,008
11899326,66
3082544,604
1438856,689
9687905,658
9452796,614

1632562,493
7753663,376
7670352,174
7385024,007
8923704,924
5769021,308
9556003,925
5077264,845
5853185,413
7329666,798
4224070,557
9742736,789
10984821,38
9848640,539
205908,7203
20514882,23
800616,3253
16114238,21
7067497,163
1354364,479
24208903,14
24842791,7
14895558,65
15653706,09
5744514,395
28914864,64
17384534,07
7251019,218
36219385,75
12178506,25
15910299,36
15307342,74
17027671,59
15497944,74
20428075,97
24371430,56
54989291,34
26406678,69
57843272,4
58025947,83
33416988,59
34806966,4
38859550,66
62897270,31
44059630,93
68619619,66
44112461,99
84207720,48
90887173,68
62453004,14
1911291217,70

regresní parametr	čip	šířka sběrnice
a	-0,048369285	0,145182782
b	63,57252147	6,369404554
c	-10575,39954	424,1415256

Název	cip	sbernice
	Rychlost graf. čipu [MHz]	Šířka paměťové sběrnice [bit]
Asus A7000/T 64MB	150	64
Inno3D GeForce2 MX400 64MB	200	64
MicroStar MX4000-T64 64MB	250	32
Asus A9200SE/T 64MB	200	64
Sapphire Atlantis ATI Radeon 9200 SE 128MB	250	64
MicroStar MX4000-T128 128MB	250	64
Asus V9400 Magic 128MB	250	64
GigaByte MAYA GV-N52128D 128MB	250	64
Sapphire Atlantis ATI Radeon 9250 64MB	240	64
Club3D ATI 9250 128MB(64bit)	240	64
Sapphire Atlantis ATI Radeon 9250 128MB(64bit)	240	64
Asus A9200SET 128MB	200	64
Asus A9250/TD 128MB	240	64
Asus V9520 Magic 128MB	250	64
Asus V9520-X/TD 128MB	250	64
Sapphire Atlantis ATI Radeon 9250 128MB	240	128
GigaByte MAYA GV-R925128D 128MB	240	128
Club3D ATI 9250 128MB(128bit)	240	128
Sapphire Atlantis ATI Radeon 9250 128MB VIVO	240	128
GigaByte MAYA GV-R955128T 128MB OEM	250	64
Inno3D GeForce FX 5200 128MB	250	128
Asus A9250GE/TD 128MB	240	128
Gainward FX PowerPack Pro/665 TV/DVI 128MB	270	64
MicroStar RX300SE-TD128E 128MB, PCI-E	325	64
MicroStar FX5200-TD128 128MB	250	64
Sapphire Atlantis ATI Radeon 9250 256MB	240	128
MicroStar NX6200TC-TD64E 64MB, PCI-E	350	128
GigaByte MAYA GV-R925128VH 128MB VIVO OEM	240	128
Club3D ATI 9250 128MB(128bit) VIVO	240	128
Asus V9520/TD 128MB	250	128
Inno3D GeForce FX-5500 128MB	270	128
MicroStar FX5500-T128 128MB	200	64
Asus A9550/TD 128MB	250	64
Sapphire Atlantis ATI Radeon 9600SE 128MB	325	64
Sapphire Atlantis ATI Radeon 9550 128MB	250	128
MicroStar RX300LE-TD128E	325	64
Club3D FX5200 256MB	250	128
Asus A9550GE/TD 128MB	250	128
Club3D ATI 9550 256MB(128bit)	250	128
Sapphire Atlantis ATI Radeon 9550 128MB VIVO	250	128
Asus A9250GE/TD 256MB	240	128
GigaByte MAYA GV-N55128DP 128MB	300	128
Inno3D GeForce FX-5500 256MB	270	128
Sapphire Atlantis ATI Radeon 9550 256MB	250	128
Inno3D GeForce FX-5500 256MB, VIVO	270	128
Club3D ATI 9600 128MB	400	128
MicroStar NX6200-TD128E 128MB, PCI-E	300	128
Sapphire Atlantis ATI Radeon 9600 128MB	325	128

MicroStar NX6200-TD128 128MB	300	128
Asus A9600GE/TD 128MB	325	128
Asus V9570LE/TD 128MB	250	64
Asus A9550GE/TD 256MB	250	128
Abit Radeon 9550-256CDT 256MB	250	128
Asus V9570GE/TD 128MB	250	128
Sapphire Atlantis ATI Radeon X600Pro 128MB, PCI-E	400	128
Club3D ATI 9600 256MB	400	128
Asus V9520 VideoSuite 128MB	250	128
Sapphire Atlantis ATI Radeon X600Pro 256MB, PCI-E	400	128
Club3D ATI 9600PRO 128MB	400	128
Inno3D GeForce 6200 128MB	300	128
MicroStar RX600XT-TD128E II 128MB, PCI-E	500	128
Asus N6200GE/TD 128MB	300	128
Sapphire Atlantis ATI Radeon 9600XT 128MB	500	128
MicroStar FX5700LE-TD256 256MB	250	128
Asus V9570GE/TD 256MB	250	128
Sapphire Atlantis ATI Radeon 9600XT 128MB VIVO	500	128
Sapphire Atlantis ATI Radeon X700 128MB, PCI-E	400	128
Sapphire Atlantis ATI Radeon 9800SE 128MB	325	128
Inno3D GeForce 6600 128MB, AGP	300	128
Sapphire Atlantis ATI Radeon 9600XT 256MB	500	128
Gainward FX PowerPack Ultra/1760 XP TV-DVI 128MB	300	128
Sapphire Atlantis ATI Radeon 9600XT 256MB VIVO	500	128
Sapphire Atlantis ATI Radeon X700 128MB	400	128
Inno3D GeForce 6600 256MB, AGP	300	128
Club3D ATI 9600XT 256MB	500	128
Gainward FX PowerPack Ultra/1760 XP Golden Sample	400	128
MicroStar NX6600-TD128E 128MB, PCI-E	300	128
Asus N6600/TD 128MB	300	128
Asus V9570 TD 128MB	425	128
Gainward FX PowerPack Ultra/1780 TV/DVI 256MB	300	128
Club3D ATI 9600XT 128MB	500	128
Sapphire Atlantis ATI Radeon 9800Pro 128MB	380	256
Sapphire Atlantis ATI Radeon X700 256MB, PCI-E	400	128
MicroStar NX6600-TD256E 256MB, PCI-E	300	128
Sapphire Atlantis ATI Radeon X700 256MB	400	128
MicroStar RX700 PRO-TD128E 128MB, PCI-E	420	128
Asus V9570/TD 256MB	425	128
Asus A9600XT 128MB	500	128
GigaByte MAYA GV-R96X128D 128MB	500	128
Sapphire Atlantis ATI Radeon X700 Fireblade 256MB	400	128
Asus N6600/TD 256MB	300	128
Club3D All-In-Wonder 9800SE 128MB	380	256
MicroStar NX6600-VTD256EH 256MB, VIVO, PCI-E	300	128
GigaByte MAYA GV-R96X128DU 128MB	550	128
Sapphire Atlantis ATI Radeon X700PRO 256MB, PCI-E	420	128
MicroStar NX6600-VTD256 256MB	400	128
Sapphire Atlantis ATI Radeon X700PRO 128MB	420	128
Club3D ATI 9800Pro 256bit 128MB	380	256
MicroStar NX6600-VTD128E Diamond 128MB, VIVO, PC	400	128
Inno3D GeForce 6600GT 128MB, AGP	500	128
Gainward FX PowerPack Ultra/1960 TV-DVI 128MB	500	128
Inno3D GeForce 6600PE 128MB, AGP	400	128
GigaByte MAYA GV-N66T128D 128MB	500	128
Asus V9999LE/TD 128MB	300	256

MicroStar NX6600GT-TD128E 128MB, PCI-E	500	128
Sapphire Atlantis ATI Radeon X700PRO 256MB	420	128
MicroStar NX6600-VTD128 Diamond	400	128
MicroStar NX6600GT-VTD128 128MB	500	128
Asus N6600GT/TD 128MB	500	128
MicroStar RX700 PRO-TD256E 256MB, PCI-E	420	128
GigaByte MAYA GV-N66T128VP 128MB	500	128
Gainward FX PowerPack Ultra/1960 TV-DVI-DVI XP 256	500	128
Gainward FX PowerPack Ultra/1960 XP 128MB, AGP	500	128
Asus N6600GT TOP 128MB	520	128
MicroStar NX6800LE-TD128 128MB	300	256
Gainward FX PowerPack Ultra/1960 XP Golden Sample	540	128
Gainward FX PowerPack Ultra/1960 XP 128MB Golden :	540	128
Gainward FX PowerPack Ultra/2100 TV/DVI 128MB	325	256
MicroStar NX6800-TD128 128MB	325	256
Inno3D GeForce 6800PE 128MB	350	256
Leadtek Winfast A400 TDH 128MB	325	256
Asus V9999/TD 128MB	325	256
Gainward FX PowerPack Ultra/2100 TV/DVI Golden San	350	256
Sapphire Atlantis ATI Radeon X800Pro 256MB	475	256
MicroStar NX6800-TD256E 256MB, PCI-E	325	256
Sapphire Atlantis ATI Radeon X800 Pro 256 MB, PCI-E	475	256
Asus V9999GT/TD 128MB	350	256
Sapphire Atlantis ATI Radeon X800 XL 256 MB, PCI-E	400	256
Excalibur X800Pro IceQ II 256MB	475	256
Asus V9999GT/TD 256MB	350	256
Inno3D GeForce 6800GT 256MB	350	256
Sapphire Atlantis ATI Radeon X800 XL Ultimate 256MB	400	256
Asus AX800XL/2DTV 256MB	400	256
MicroStar RX800 PRO VTD 256MB VIVO	475	256
MicroStar NX6800GT-TD256 256MB	350	256
Gainward FX PowerPack Ultra/2400 TV/DVI Golden San	400	256
Club3D ATI X800 Pro 256MB VIVO	475	256
Gainward FX PowerPack Ultra/2400 TV/DVI 256MB	350	256
Club3D ATI X800 Pro 256MB	475	256
Asus AX800 PRO/TD 256MB	475	256
Sapphire Atlantis ATI Radeon X800XT 256 MB VIVO	500	256
Club3D ATI X800 XT 256MB VIVO	500	256
Sapphire Atlantis ATI Radeon X800XT Platinum 256MB	520	256
Sapphire Atlantis ATI Radeon X850 XT 256 MB	520	256
Asus AX850PRO/HTVD 256MB, VIVO	520	256
Gainward FX PowerPack Ultra/2600 TV-DVI 256MB	400	256
Sapphire Atlantis ATI Radeon X850 XT 256 MB, VIVO, P	520	256
Inno3D GeForce 6800 Ultra 256MB	400	256
MicroStar NX6800Ultra-T2D256 256MB	400	256
Asus AX800XT 256MB	520	256
Asus AX800XT/TV 256MB, VIVO	520	256
Asus AX850XT/HTVD 256MB, VIVO	520	256
Asus V9999 Ultra/2DT 256MB	425	256
Sapphire Atlantis ATI Radeon X850 XT 256 MB, VIVO	520	256
Gainward FX PowerPack Ultra/2600 Golden Sample 256	430	256
Asus AX850XTP/HTVD 256MB, VIVO	540	256
Asus V9999 Ultra Deluxe 256MB	400	256
MicroStar NX6800Ultra-T2D256E 256MB, PCI-E	400	256
Sapphire Atlantis ATI Radeon X850 XT Platinum 256 MB	540	256

velikost paměti	rychlost paměti
0,002711973	0,010284215
39,44548541	0,043605894
-1447,799374	422,1239626

velikost	rychlost	cena_vyr					
Velikost graf. paměti [MB]	Rychlost graf. paměti [MHz]	Cena s DPH [Kč]	rcip	rsbernice	rvelikost	rrychlost	REGRESE
64	300	837	-2127,83	1426,452	1087,82	1360,785	436,80673
64	166	885	204,3334	1426,452	1087,82	712,7544	857,83995
64	400	906	2294,651	776,6296	1087,82	2085,041	1561,0352
64	333	1060	204,3334	1426,452	1087,82	1577,051	1073,9141
128	333	1127	2294,651	1426,452	3645,656	1577,051	2235,9524
128	332	1130	2294,651	1426,452	3645,656	1570,168	2234,2317
128	332	1147	2294,651	1426,452	3645,656	1570,168	2234,2317
128	400	1175	2294,651	1426,452	3645,656	2085,041	2362,9498
64	400	1186	1895,935	1426,452	1087,82	2085,041	1623,8119
128	400	1220	1895,935	1426,452	3645,656	2085,041	2263,2709
128	400	1235	1895,935	1426,452	3645,656	2085,041	2263,2709
128	333	1241	204,3334	1426,452	3645,656	1577,051	1713,3731
128	400	1290	1895,935	1426,452	3645,656	2085,041	2263,2709
128	332	1344	2294,651	1426,452	3645,656	1570,168	2234,2317
128	400	1344	2294,651	1426,452	3645,656	2085,041	2362,9498
128	400	1359	1895,935	3618,1	3645,656	2085,041	2811,1828
128	400	1457	1895,935	3618,1	3645,656	2085,041	2811,1828
128	400	1458	1895,935	3618,1	3645,656	2085,041	2811,1828
128	400	1517	1895,935	3618,1	3645,656	2085,041	2811,1828
128	400	1526	2294,651	1426,452	3645,656	2085,041	2362,9498
128	333	1553	2294,651	3618,1	3645,656	1577,051	2783,8643
128	400	1605	1895,935	3618,1	3645,656	2085,041	2811,1828
128	400	1636	3063,06	1426,452	3645,656	2085,041	2555,0523
128	400	1642	4976,664	1426,452	3645,656	2085,041	3033,4532
128	400	1659	2294,651	1426,452	3645,656	2085,041	2362,9498
256	400	1664	1895,935	3618,1	8827,977	2085,041	4106,7631
64	550	1668	5749,746	3618,1	1087,82	3557,082	3503,187
128	400	1670	1895,935	3618,1	3645,656	2085,041	2811,1828
128	400	1702	1895,935	3618,1	3645,656	2085,041	2811,1828
128	400	1735	2294,651	3618,1	3645,656	2085,041	2910,8618
128	400	1753	3063,06	3618,1	3645,656	2085,041	3102,9642
128	400	1783	204,3334	1426,452	3645,656	2085,041	1840,3705
128	400	1795	2294,651	1426,452	3645,656	2085,041	2362,9498
128	400	1808	4976,664	1426,452	3645,656	2085,041	3033,4532
128	400	1846	2294,651	3618,1	3645,656	2085,041	2910,8618
128	400	1984	4976,664	1426,452	3645,656	2085,041	3033,4532
256	286	1990	2294,651	3618,1	8827,977	1275,803	4004,1326
125	400	1994	2294,651	3618,1	3525,261	2085,041	2880,7631
256	400	2058	2294,651	3618,1	8827,977	2085,041	4206,442
128	400	2067	2294,651	3618,1	3645,656	2085,041	2910,8618
256	400	2094	1895,935	3618,1	8827,977	2085,041	4106,7631
128	500	2104	4143,121	3618,1	3645,656	3014,981	3605,4645
256	400	2119	3063,06	3618,1	8827,977	2085,041	4398,5445
256	400	2157	2294,651	3618,1	8827,977	2085,041	4206,442
256	400	2286	3063,06	3618,1	8827,977	2085,041	4398,5445
128	400	2353	7114,524	3618,1	3645,656	2085,041	4115,83
128	550	2393	4143,121	3618,1	3645,656	3557,082	3740,9899
128	400	2418	4976,664	3618,1	3645,656	2085,041	3581,3652

128	400	2444	4143,121	3618,1	3645,656	2085,041	3372,9795
128	400	2453	4976,664	3618,1	3645,656	2085,041	3581,3652
128	400	2528	2294,651	1426,452	3645,656	2085,041	2362,9498
256	400	2582	2294,651	3618,1	8827,977	2085,041	4206,442
256	400	2636	2294,651	3618,1	8827,977	2085,041	4206,442
128	400	2756	2294,651	3618,1	3645,656	2085,041	2910,8618
128	600	2818	7114,524	3618,1	3645,656	4150,605	4632,2211
256	400	2948	7114,524	3618,1	8827,977	2085,041	5411,4103
128	400	2983	2294,651	3618,1	3645,656	2085,041	2910,8618
256	600	3040	7114,524	3618,1	8827,977	4150,605	5927,8013
128	600	3074	7114,524	3618,1	3645,656	4150,605	4632,2211
128	400	3084	4143,121	3618,1	3645,656	2085,041	3372,9795
128	550	3214	9118,54	3618,1	3645,656	3557,082	4984,8445
128	550	3236	4143,121	3618,1	3645,656	3557,082	3740,9899
128	600	3286	9118,54	3618,1	3645,656	4150,605	5133,2252
256	400	3311	2294,651	3618,1	8827,977	2085,041	4206,442
256	400	3371	2294,651	3618,1	8827,977	2085,041	4206,442
128	600	3402	9118,54	3618,1	3645,656	4150,605	5133,2252
128	700	3408	7114,524	3618,1	3645,656	5491,914	4967,5482
128	580	3439	4976,664	3618,1	3645,656	3907,025	4036,8614
128	550	3482	4143,121	3618,1	3645,656	3557,082	3740,9899
256	600	3502	9118,54	3618,1	8827,977	4150,605	6428,8055
128	500	3589	4143,121	3618,1	3645,656	3014,981	3605,4645
256	600	3669	9118,54	3618,1	8827,977	4150,605	6428,8055
128	500	3785	7114,524	3618,1	3645,656	3014,981	4348,315
256	500	3814	4143,121	3618,1	8827,977	3014,981	4901,0447
256	600	3835	9118,54	3618,1	8827,977	4150,605	6428,8055
128	560	3860	7114,524	3618,1	3645,656	3671,673	4512,4881
128	550	3885	4143,121	3618,1	3645,656	3557,082	3740,9899
128	550	3889	4143,121	3618,1	3645,656	3557,082	3740,9899
128	500	3922	7706,22	3618,1	3645,656	3014,981	4496,2391
256	500	3965	4143,121	3618,1	8827,977	3014,981	4901,0447
128	680	4046	9118,54	3618,1	3645,656	5207,197	5397,3733
128	680	4081	6597,634	11569,41	3645,656	5207,197	6754,9737
256	700	4106	7114,524	3618,1	8827,977	5491,914	6263,1285
256	500	4186	4143,121	3618,1	8827,977	3014,981	4901,0447
256	500	4202	7114,524	3618,1	8827,977	3014,981	5643,8953
128	860	4276	7592,718	3618,1	3645,656	8065,831	5730,5761
256	500	4373	7706,22	3618,1	8827,977	3014,981	5791,8194
128	600	4410	9118,54	3618,1	3645,656	4150,605	5133,2252
128	600	4424	9118,54	3618,1	3645,656	4150,605	5133,2252
256	600	4428	7114,524	3618,1	8827,977	4150,605	5927,8013
256	500	4542	4143,121	3618,1	8827,977	3014,981	4901,0447
128	680	4547	6597,634	11569,41	3645,656	5207,197	6754,9737
128	500	4555	4143,121	3618,1	3645,656	3014,981	3605,4645
125	600	4585	9757,779	3618,1	3525,261	4150,605	5262,9362
256	860	4720	7592,718	3618,1	8827,977	8065,831	7026,1563
256	600	4855	7114,524	3618,1	8827,977	4150,605	5927,8013
128	860	4869	7592,718	3618,1	3645,656	8065,831	5730,5761
128	680	4987	6597,634	11569,41	3645,656	5207,197	6754,9737
128	800	5130	7114,524	3618,1	3645,656	7038,907	5354,2965
128	900	5378	9118,54	3618,1	3645,656	8791,584	6293,4699
128	900	5380	9118,54	3618,1	3645,656	8791,584	6293,4699
128	800	5544	7114,524	3618,1	3645,656	7038,907	5354,2965
128	1000	5555	9118,54	3618,1	3645,656	10749,95	6783,0603
128	700	5627	4143,121	11569,41	3645,656	5491,914	6212,5246

128	1000	5669	9118,54	3618,1	3645,656	10749,95	6783,0603
256	860	5837	7592,718	3618,1	8827,977	8065,831	7026,1563
128	800	5915	7114,524	3618,1	3645,656	7038,907	5354,2965
128	900	6006	9118,54	3618,1	3645,656	8791,584	6293,4699
128	900	6044	9118,54	3618,1	3645,656	8791,584	6293,4699
256	860	6315	7592,718	3618,1	8827,977	8065,831	7026,1563
128	1120	6334	9118,54	3618,1	3645,656	13371,48	7438,4446
256	900	6349	9118,54	3618,1	8827,977	8791,584	7589,0502
128	900	6401	9118,54	3618,1	3645,656	8791,584	6293,4699
128	1100	6416	9403,257	3618,1	3645,656	12913,99	7395,251
128	700	6545	4143,121	11569,41	3645,656	5491,914	6212,5246
128	1050	6558	9649,279	3618,1	3645,656	11806,26	7179,823
128	1150	7396	9649,279	3618,1	3645,656	14073,15	7746,545
128	700	7396	4976,664	11569,41	3645,656	5491,914	6420,9104
128	700	7684	4976,664	11569,41	3645,656	5491,914	6420,9104
128	800	7805	5749,746	11569,41	3645,656	7038,907	7000,929
128	700	8098	4976,664	11569,41	3645,656	5491,914	6420,9104
128	700	8291	4976,664	11569,41	3645,656	5491,914	6420,9104
128	800	8888	5749,746	11569,41	3645,656	7038,907	7000,929
256	900	9162	8708,228	11569,41	8827,977	8791,584	9474,2992
256	600	9506	4976,664	11569,41	8827,977	4150,605	7381,1635
256	900	9603	8708,228	11569,41	8827,977	8791,584	9474,2992
128	700	9764	5749,746	11569,41	3645,656	5491,914	6614,1807
256	980	9773	7114,524	11569,41	8827,977	10341,82	9463,4316
256	900	9872	8708,228	11569,41	8827,977	8791,584	9474,2992
256	700	10670	5749,746	11569,41	8827,977	5491,914	7909,761
256	1100	10734	5749,746	11569,41	8827,977	12913,99	9765,2804
256	980	10974	7114,524	11569,41	8827,977	10341,82	9463,4316
256	980	11071	7114,524	11569,41	8827,977	10341,82	9463,4316
256	900	11105	8708,228	11569,41	8827,977	8791,584	9474,2992
256	1000	11127	5749,746	11569,41	8827,977	10749,95	9224,2689
256	1100	11284	7114,524	11569,41	8827,977	12913,99	10106,475
256	900	11401	8708,228	11569,41	8827,977	8791,584	9474,2992
256	1000	11768	5749,746	11569,41	8827,977	10749,95	9224,2689
256	900	12198	8708,228	11569,41	8827,977	8791,584	9474,2992
256	900	12697	8708,228	11569,41	8827,977	8791,584	9474,2992
256	1000	13031	9118,54	11569,41	8827,977	10749,95	10066,468
256	1000	13245	9118,54	11569,41	8827,977	10749,95	10066,468
256	1120	13340	9403,257	11569,41	8827,977	13371,48	10793,031
256	1080	13923	9403,257	11569,41	8827,977	12464,73	10566,342
256	1080	14340	9403,257	11569,41	8827,977	12464,73	10566,342
256	1100	14530	7114,524	11569,41	8827,977	12913,99	10106,475
256	1080	14542	9403,257	11569,41	8827,977	12464,73	10566,342
256	1000	14720	7114,524	11569,41	8827,977	10749,95	9565,4634
256	1100	14732	7114,524	11569,41	8827,977	12913,99	10106,475
256	1120	15184	9403,257	11569,41	8827,977	13371,48	10793,031
256	1120	15303	9403,257	11569,41	8827,977	13371,48	10793,031
256	1080	15637	9403,257	11569,41	8827,977	12464,73	10566,342
256	1100	15637	7706,22	11569,41	8827,977	12913,99	10254,399
256	1080	16041	9403,257	11569,41	8827,977	12464,73	10566,342
256	1200	16101	7817,304	11569,41	8827,977	15283,72	10874,602
256	1180	16291	9649,279	11569,41	8827,977	14793,32	11209,996
256	1100	16291	7114,524	11569,41	8827,977	12913,99	10106,475
256	1100	16648	7114,524	11569,41	8827,977	12913,99	10106,475
256	1180	17552	9649,279	11569,41	8827,977	14793,32	11209,996

cip

a	b	c	y
4201947543125	9529819375	22731125	162198721725
9529819375	22731125	57835	372496155
22731125	57835	159	895740
162198721725	9529819375	22731125	
372496155	22731125	57835	a=
895740	57835	159	
4201947543125	162198721725	22731125	
9529819375	372496155	57835	b=
22731125	895740	159	
4201947543125	9529819375	162198721725	
9529819375	22731125	372496155	c=
22731125	57835	895740	

Odhad ceny

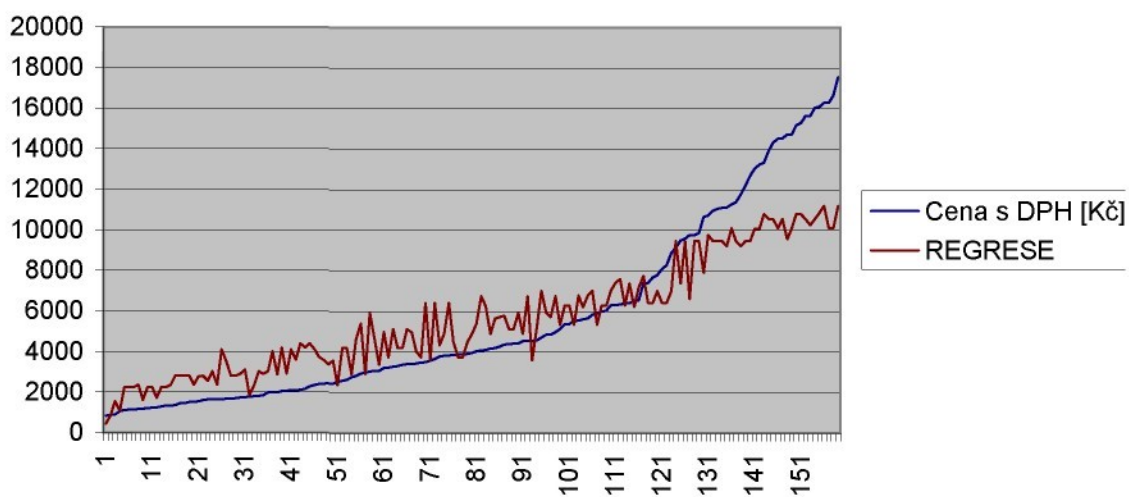
Grafická karta

200

64

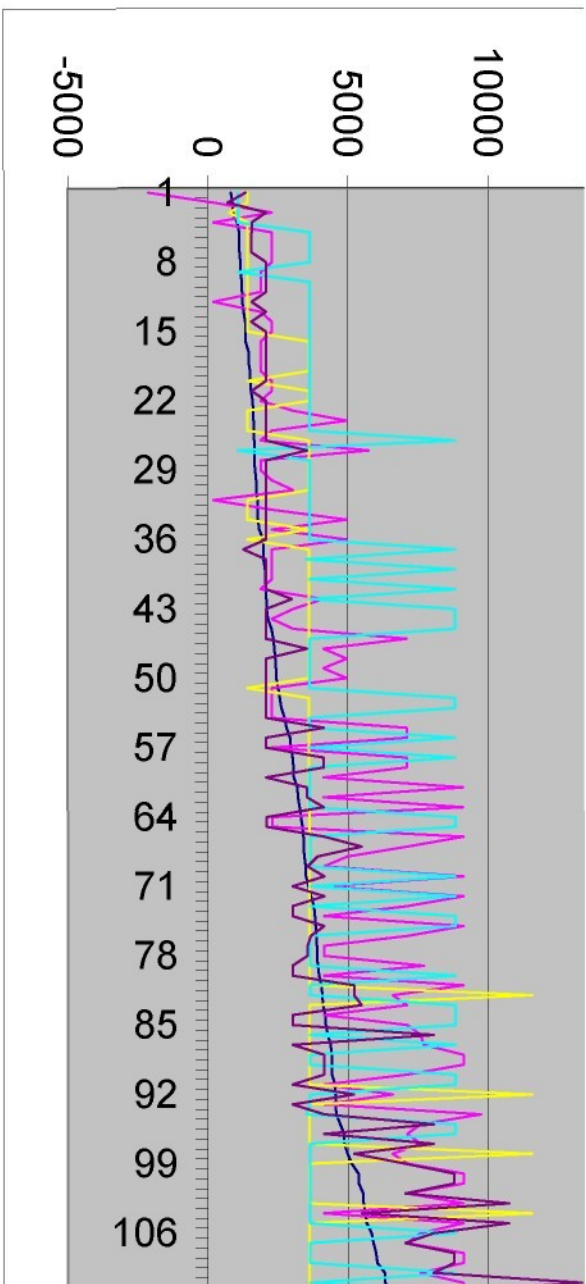
64

166



20000

15000

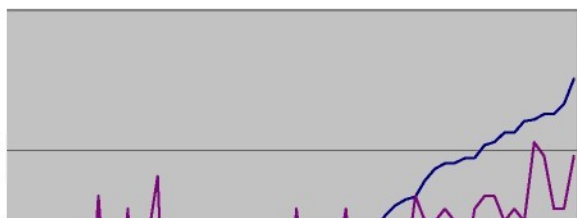


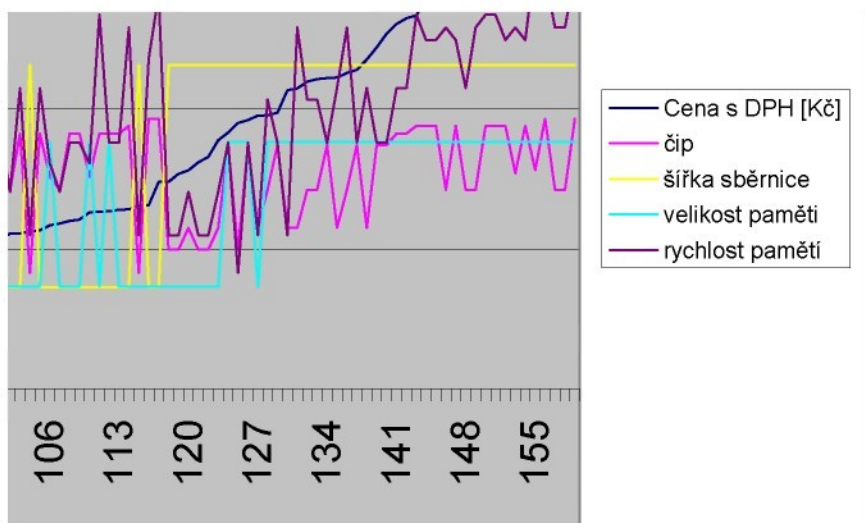
sbernice

	a	b	c	y
3,45E+18	225872707584	979140608	4617216	40987723776
	979140608	4617216	24800	182081984
	4617216	24800	159	895740
-1,67E+17	40987723776	979140608	4617216	
-0,048369	182081984	4617216	24800	a=
	895740	24800	159	
2,19E+20	225872707584	40987723776	4617216	
63,57252	979140608	182081984	24800	b=
	4617216	895740	159	
-3,64E+22	225872707584	979140608	40987723776	
-10575,4	979140608	4617216	182081984	c=
	4617216	24800	895740	

Cena				
204,3334	1426,452093	1087,819932	712,7543829	857,8399458

PH [Kč]
SE





velikost

	a	b	c	y	
2,69E+17	298820736162	1261673162	5675538	42360663627	8,45E+16
	1261673162	5675538	28154	186534631	
	5675538	28154	159	895740	
3,9E+16	42360663627	1261673162	5675538		2,29E+14
0,145183	186534631	5675538	28154	a=	0,002712
	895740	28154	159		
1,71E+18	298820736162	42360663627	5675538		3,33E+18
6,369405	1261673162	186534631	28154	b=	39,44549
	5675538	895740	159		
1,14E+20	298820736162	1261673162	42360663627		-1,22E+20
424,1415	1261673162	5675538	186534631	c=	-1447,799
	5675538	28154	895740		

rychlost

a	b	c	y	
64693427640564	69178228204	80129280	702162218013	9,63E+20
69178228204	80129280	104470	759037205	
80129280	104470	159	895740	
702162218013	69178228204	80129280		9,9E+18
759037205	80129280	104470	a=	0,010284
895740	104470	159		
64693427640564	702162218013	80129280		4,2E+19
69178228204	759037205	104470	b=	0,043606
80129280	895740	159		
64693427640564	69178228204	702162218013		4,07E+23
69178228204	80129280	759037205	c=	422,124
80129280	104470	895740		

	Cena	Pipelines	Rychlost graf. čipu	Šířka pamět'. sběrnice	Velikost graf. paměti
Cena	1				
Pipelines	0,943430183	1			
Rychlost graf. čipu	0,631274858	0,52752198	1		
Šířka pamět'. sběrnice	0,861735998	0,86589148	0,47496365	1	
Velikost graf. paměti	0,591654197	0,5581557	0,3450282	0,525260748	1
Rychlost graf. paměti	0,885468762	0,85122858	0,78908156	0,730332579	0,449036313

Pozor! Mezi vysvětlujícími proměnnými (pipelines a šířka sběrnice; pipelines a rychlost graf. paměti

Rychlost graf. paměti
1

) existuje vysoká korelace

	Cena	Rychlost graf. čipu	Šířka pamět'. sběrnice	Velikost graf. paměti	Rychlost graf. paměti
Cena	1				
Rychlost graf. čipu	0,631275	1			
Šířka pamět'. sběrnice	0,861736	0,474963648	1		
Velikost graf. paměti	0,591654	0,345028203	0,525260748	1	
Rychlost graf. paměti	0,885469	0,789081557	0,730332579	0,449036313	1

Po vyřazení proměnné **pipelines** se již v modelu multikolinearita nevyskytuje. Cena je stále vysoce k

Název	Cena s DPH [Kč]	Rychlost graf. čipu [MHz]
Asus A7000/T 64MB	837	150
Inno3D GeForce2 MX400 64MB	885	200
MicroStar MX4000-T64 64MB	906	250
Asus A9200SE/T 64MB	1060	200
Sapphire Atlantis ATI Radeon 9200 SE 128MB	1127	250
MicroStar MX4000-T128 128MB	1130	250
Asus V9400 Magic 128MB	1147	250
GigaByte MAYA GV-N52128D 128MB	1175	250
Sapphire Atlantis ATI Radeon 9250 64MB	1186	240
Club3D ATI 9250 128MB(64bit)	1220	240
Sapphire Atlantis ATI Radeon 9250 128MB(64bit)	1235	240
Asus A9200SET 128MB	1241	200
Asus A9250/TD 128MB	1290	240
Asus V9520 Magic 128MB	1344	250
Asus V9520-X/TD 128MB	1344	250
Sapphire Atlantis ATI Radeon 9250 128MB	1359	240
GigaByte MAYA GV-R925128D 128MB	1457	240
Club3D ATI 9250 128MB(128bit)	1458	240
Sapphire Atlantis ATI Radeon 9250 128MB VIVO	1517	240
GigaByte MAYA GV-R955128T 128MB OEM	1526	250

Inno3D GeForce FX 5200 128MB	1553	250
Asus A9250GE/TD 128MB	1605	240
Gainward FX PowerPack Pro/665 TV/DVI 128MB	1636	270
MicroStar RX300SE-TD128E 128MB, PCI-E	1642	325
MicroStar FX5200-TD128 128MB	1659	250
Sapphire Atlantis ATI Radeon 9250 256MB	1664	240
MicroStar NX6200TC-TD64E 64MB, PCI-E	1668	350
GigaByte MAYA GV-R925128VH 128MB VIVO OEM	1670	240
Club3D ATI 9250 128MB(128bit) VIVO	1702	240
Asus V9520/TD 128MB	1735	250
Inno3D GeForce FX-5500 128MB	1753	270
MicroStar FX5500-T128 128MB	1783	200
Asus A9550/TD 128MB	1795	250
Sapphire Atlantis ATI Radeon 9600SE 128MB	1808	325
Sapphire Atlantis ATI Radeon 9550 128MB	1846	250
MicroStar RX300LE-TD128E	1984	325
Club3D FX5200 256MB	1990	250
Asus A9550GE/TD 128MB	1994	250
Club3D ATI 9550 256MB(128bit)	2058	250
Sapphire Atlantis ATI Radeon 9550 128MB VIVO	2067	250
Asus A9250GE/TD 256MB	2094	240
GigaByte MAYA GV-N55128DP 128MB	2104	300
Inno3D GeForce FX-5500 256MB	2119	270

Sapphire Atlantis ATI Radeon 9550 256MB	2157	250
Inno3D GeForce FX-5500 256MB, VIVO	2286	270
Club3D ATI 9600 128MB	2353	400
MicroStar NX6200-TD128E 128MB, PCI-E	2393	300
Sapphire Atlantis ATI Radeon 9600 128MB	2418	325
MicroStar NX6200-TD128 128MB	2444	300
Asus A9600GE/TD 128MB	2453	325
Asus V9570LE/TD 128MB	2528	250
Asus A9550GE/TD 256MB	2582	250
Abit Radeon 9550-256CDT 256MB	2636	250
Asus V9570GE/TD 128MB	2756	250
Sapphire Atlantis ATI Radeon X600Pro 128MB, PCI-E	2818	400
Club3D ATI 9600 256MB	2948	400
Asus V9520 VideoSuite 128MB	2983	250
Sapphire Atlantis ATI Radeon X600Pro 256MB, PCI-E	3040	400
Club3D ATI 9600PRO 128MB	3074	400
Inno3D GeForce 6200 128MB	3084	300
MicroStar RX600XT-TD128E II 128MB, PCI-E	3214	500
Asus N6200GE/TD 128MB	3236	300
Sapphire Atlantis ATI Radeon 9600XT 128MB	3286	500
MicroStar FX5700LE-TD256 256MB	3311	250
Asus V9570GE/TD 256MB	3371	250
Sapphire Atlantis ATI Radeon 9600XT 128MB VIVO	3402	500

Sapphire Atlantis ATI Radeon X700 128MB, PCI-E	3408	400
Sapphire Atlantis ATI Radeon 9800SE 128MB	3439	325
Inno3D GeForce 6600 128MB, AGP	3482	300
Sapphire Atlantis ATI Radeon 9600XT 256MB	3502	500
Gainward FX PowerPack Ultra/1760 XP TV-DVI 128MB	3589	300
Sapphire Atlantis ATI Radeon 9600XT 256MB VIVO	3669	500
Sapphire Atlantis ATI Radeon X700 128MB	3785	400
Inno3D GeForce 6600 256MB, AGP	3814	300
Club3D ATI 9600XT 256MB	3835	500
Gainward FX PowerPack Ultra/1760 XP Golden Sample 128MB	3860	400
MicroStar NX6600-TD128E 128MB, PCI-E	3885	300
Asus N6600/TD 128MB	3889	300
Asus V9570 TD 128MB	3922	425
Gainward FX PowerPack Ultra/1780 TV/DVI 256MB	3965	300
Club3D ATI 9600XT 128MB	4046	500
Sapphire Atlantis ATI Radeon 9800Pro 128MB	4081	380
Sapphire Atlantis ATI Radeon X700 256MB, PCI-E	4106	400
MicroStar NX6600-TD256E 256MB, PCI-E	4186	300
Sapphire Atlantis ATI Radeon X700 256MB	4202	400
MicroStar RX700 PRO-TD128E 128MB, PCI-E	4276	420
Asus V9570/TD 256MB	4373	425
Asus A9600XT 128MB	4410	500
GigaByte MAYA GV-R96X128D 128MB	4424	500

Sapphire Atlantis ATI Radeon X700 Fireblade 256MB	4428	400
Asus N6600/TD 256MB	4542	300
Club3D All-In-Wonder 9800SE 128MB	4547	380
MicroStar NX6600-VTD256EH 256MB, VIVO, PCI-E	4555	300
GigaByte MAYA GV-R96X128DU 128MB	4585	550
Sapphire Atlantis ATI Radeon X700PRO 256MB, PCI-E	4720	420
MicroStar NX6600-VTD256 256MB	4855	400
Sapphire Atlantis ATI Radeon X700PRO 128MB	4869	420
Club3D ATI 9800Pro 256bit 128MB	4987	380
MicroStar NX6600-VTD128E Diamond 128MB, VIVO, PCI-E	5130	400
Inno3D GeForce 6600GT 128MB, AGP	5378	500
Gainward FX PowerPack Ultra/1960 TV-DVI 128MB	5380	500
Inno3D GeForce 6600PE 128MB, AGP	5544	400
GigaByte MAYA GV-N66T128D 128MB	5555	500
Asus V9999LE/TD 128MB	5627	300
MicroStar NX6600GT-TD128E 128MB, PCI-E	5669	500
Sapphire Atlantis ATI Radeon X700PRO 256MB	5837	420
MicroStar NX6600-VTD128 Diamond	5915	400
MicroStar NX6600GT-VTD128 128MB	6006	500
Asus N6600GT/TD 128MB	6044	500
MicroStar RX700 PRO-TD256E 256MB, PCI-E	6315	420
GigaByte MAYA GV-N66T128VP 128MB	6334	500
Gainward FX PowerPack Ultra/1960 TV-DVI-DVI XP 256MB	6349	500

Gainward FX PowerPack Ultra/1960 XP 128MB, AGP	6401	500
Asus N6600GT TOP 128MB	6416	520
MicroStar NX6800LE-TD128 128MB	6545	300
Gainward FX PowerPack Ultra/1960 XP Golden Sample 128MB	6558	540
Gainward FX PowerPack Ultra/1960 XP 128MB Golden Sample	7396	540
Gainward FX PowerPack Ultra/2100 TV/DVI 128MB	7396	325
MicroStar NX6800-TD128 128MB	7684	325
Inno3D GeForce 6800PE 128MB	7805	350
Leadtek Winfast A400 TDH 128MB	8098	325
Asus V9999/TD 128MB	8291	325
Gainward FX PowerPack Ultra/2100 TV/DVI Golden Sample	8888	350
Sapphire Atlantis ATI Radeon X800Pro 256MB	9162	475
MicroStar NX6800-TD256E 256MB, PCI-E	9506	325
Sapphire Atlantis ATI Radeon X800 Pro 256 MB, PCI-E	9603	475
Asus V9999GT/TD 128MB	9764	350
Sapphire Atlantis ATI Radeon X800 XL 256 MB, PCI-E	9773	400
Excabur X800Pro IceQ II 256MB	9872	475
Asus V9999GT/TD 256MB	10670	350
Inno3D GeForce 6800GT 256MB	10734	350
Sapphire Atlantis ATI Radeon X800 XL Ultimate 256MB	10974	400
Asus AX800XL/2DTV 256MB	11071	400
MicroStar RX800 PRO VTD 256MB VIVO	11105	475
MicroStar NX6800GT-TD256 256MB	11127	350

Gainward FX PowerPack Ultra/2400 TV/DVI Golden Sample	11284	400
Club3D ATI X800 Pro 256MB VIVO	11401	475
Gainward FX PowerPack Ultra/2400 TV/DVI 256MB	11768	350
Club3D ATI X800 Pro 256MB	12198	475
Asus AX800 PRO/TD 256MB	12697	475
Sapphire Atlantis ATI Radeon X800XT 256 MB VIVO	13031	500
Club3D ATI X800 XT 256MB VIVO	13245	500
Sapphire Atlantis ATI Radeon X800XT Platinum 256MB	13340	520
Sapphire Atlantis ATI Radeon X850 XT 256 MB	13923	520
Asus AX850PRO/HTVD 256MB, VIVO	14340	520
Gainward FX PowerPack Ultra/2600 TV-DVI 256MB	14530	400
Sapphire Atlantis ATI Radeon X850 XT 256 MB, VIVO, PCI-I	14542	520
Inno3D GeForce 6800 Ultra 256MB	14720	400
MicroStar NX6800Ultra-T2D256 256MB	14732	400
Asus AX800XT 256MB	15184	520
Asus AX800XT/TVD 256MB, VIVO	15303	520
Asus AX850XT/HTVD 256MB, VIVO	15637	520
Asus V9999 Ultra/2DT 256MB	15637	425
Sapphire Atlantis ATI Radeon X850 XT 256 MB, VIVO	16041	520
Gainward FX PowerPack Ultra/2600 Golden Sample 256ME	16101	430
Asus AX850XTP/HTVD 256MB, VIVO	16291	540
Asus V9999 Ultra Deluxe 256MB	16291	400
MicroStar NX6800Ultra-T2D256E 256MB, PCI-E	16648	400

Sapphire Atlantis ATI Radeon X850 XT Platinum 256 MB, V	17552	540
--	--------------	------------

Šířka paměťové sběrnice [bit]	Velikost graf. paměti [MB]	Rychlost graf. paměti [MHz]	Počet pipelines
64	64	300	1
64	64	166	2
32	64	400	2
64	64	333	4
64	128	333	4
64	128	332	2
64	128	332	2
64	128	400	4
64	64	400	4
64	128	400	4
64	128	400	4
64	128	333	4
64	128	400	4
64	128	332	4
64	128	400	4
128	128	400	4
128	128	400	4
128	128	400	4
128	128	400	4
64	128	400	4

128	128	333	4
128	128	400	4
64	128	400	4
64	128	400	4
64	128	400	4
128	256	400	4
128	64	550	4
128	128	400	4
128	128	400	4
128	128	400	4
128	128	400	4
64	128	400	4
64	128	400	4
64	128	400	4
128	128	400	4
64	128	400	4
128	256	286	4
128	125	400	4
128	256	400	4
128	128	400	4
128	256	400	4
128	128	500	4
128	256	400	4

128	256	400	4
128	256	400	4
128	128	400	4
128	128	550	4
128	128	400	4
128	128	400	4
128	128	400	4
64	128	400	4
128	256	400	4
128	256	400	4
128	128	400	4
128	128	600	4
128	256	400	4
128	128	400	4
128	256	600	4
128	128	600	4
128	128	400	4
128	128	550	4
128	128	550	4
128	128	600	4
128	256	400	4
128	256	400	4
128	128	600	4

128	128	700	8
128	128	580	4
128	128	550	8
128	256	600	4
128	128	500	8
128	256	600	4
128	128	500	8
128	256	500	8
128	256	600	4
128	128	560	8
128	128	550	8
128	128	550	8
128	128	500	4
128	256	500	8
128	128	680	4
256	128	680	8
128	256	700	8
128	256	500	8
128	256	500	8
128	128	860	8
128	256	500	4
128	128	600	4
128	128	600	4

128	256	600	8
128	256	500	8
256	128	680	4
128	128	500	8
128	125	600	4
128	256	860	8
128	256	600	8
128	128	860	8
256	128	680	8
128	128	800	8
128	128	900	8
128	128	900	8
128	128	800	8
128	128	1000	8
256	128	700	8
128	128	1000	8
128	256	860	8
128	128	800	8
128	128	900	8
128	128	900	8
128	256	860	8
128	128	1120	8
128	256	900	8

128	128	900	8
128	128	1100	8
256	128	700	8
128	128	1050	8
128	128	1150	8
256	128	700	12
256	128	700	12
256	128	800	12
256	128	700	12
256	128	700	12
256	128	800	12
256	256	900	12
256	256	600	12
256	256	900	12
256	128	700	16
256	256	980	16
256	256	900	12
256	256	700	16
256	256	1100	16
256	256	980	16
256	256	980	16
256	256	900	12
256	256	1000	16

256	256	1100	16
256	256	900	12
256	256	1000	16
256	256	900	12
256	256	900	12
256	256	1000	16
256	256	1000	16
256	256	1120	16
256	256	1080	16
256	256	1080	12
256	256	1100	16
256	256	1080	16
256	256	1000	16
256	256	1100	16
256	256	1120	16
256	256	1120	16
256	256	1080	16
256	256	1100	16
256	256	1080	16
256	256	1200	16
256	256	1180	16
256	256	1100	16
256	256	1100	16

256

256

1180

16

regresní parametr	pipelines	čip	šířka sběrnice	velikost paměti	rychlost paměti
a	26,23961612	-0,048369285	0,145182782	0,002711973	0,010284215
b	434,5506097	63,57252147	6,369404554	39,44548541	0,043605894
c	162,1633636	-10575,39954	424,1415256	-1447,799374	422,1239626

	pipelines	cip	sbernice	velikost	rychlost	cena_vyr
Název	Počet pipelines	Rychlost graf. čipu [MHz]	Šířka paměťové sběrnice [bit]	Velikost graf. paměti [MB]	Rychlost graf. paměti [MHz]	Cena s DPH [Kč]
Asus A7000/T 64MB	1	150	64	64	300	837
Inno3D GeForce2 MX400 64MB	2	200	64	64	166	885
MicroStar MX4000-T64 64MB	2	250	32	64	400	906
Asus A9200SE/T 64MB	4	200	64	64	333	1060
Sapphire Atlantis ATI Radeon 9200 SE 128MB	4	250	64	128	333	1127
MicroStar MX4000-T128 128MB	2	250	64	128	332	1130
Asus V9400 Magic 128MB	2	250	64	128	332	1147
GigaByte MAYA GV-N52128D 128MB	4	250	64	128	400	1175
Sapphire Atlantis ATI Radeon 9250 64MB	4	240	64	64	400	1186
Club3D ATI 9250 128MB(64bit)	4	240	64	128	400	1220
Sapphire Atlantis ATI Radeon 9250 128MB(64bit)	4	240	64	128	400	1235
Asus A9200SET 128MB	4	200	64	128	333	1241
Asus A9250/TD 128MB	4	240	64	128	400	1290
Asus V9520 Magic 128MB	4	250	64	128	332	1344
Asus V9520-X/TD 128MB	4	250	64	128	400	1344
Sapphire Atlantis ATI Radeon 9250 128MB	4	240	128	128	400	1359
GigaByte MAYA GV-R925128D 128MB	4	240	128	128	400	1457
Club3D ATI 9250 128MB(128bit)	4	240	128	128	400	1458
Sapphire Atlantis ATI Radeon 9250 128MB VIVO	4	240	128	128	400	1517
GigaByte MAYA GV-R955128T 128MB OEM	4	250	64	128	400	1526
Inno3D GeForce FX 5200 128MB	4	250	128	128	333	1553
Asus A9250GE/TD 128MB	4	240	128	128	400	1605
Gainward FX PowerPack Pro/665 TV/DVI 128MB	4	270	64	128	400	1636
MicroStar RX300SE-TD128E 128MB, PCI-E	4	325	64	128	400	1642
MicroStar FX5200-TD128 128MB	4	250	64	128	400	1659
Sapphire Atlantis ATI Radeon 9250 256MB	4	240	128	256	400	1664
MicroStar NX6200TC-TD64E 64MB, PCI-E	4	350	128	64	550	1668
GigaByte MAYA GV-R925128VH 128MB VIVO OEM	4	240	128	128	400	1670
Club3D ATI 9250 128MB(128bit) VIVO	4	240	128	128	400	1702
Asus V9520/TD 128MB	4	250	128	128	400	1735
Inno3D GeForce FX-5500 128MB	4	270	128	128	400	1753
MicroStar FX5500-T128 128MB	4	200	64	128	400	1783
Asus A9550/TD 128MB	4	250	64	128	400	1795

Sapphire Atlantis ATI Radeon 9600SE 128MB	4	325	64	128	400	1808
Sapphire Atlantis ATI Radeon 9550 128MB	4	250	128	128	400	1846
MicroStar RX300LE-TD128E	4	325	64	128	400	1984
Club3D FX5200 256MB	4	250	128	256	286	1990
Asus A9550GE/TD 128MB	4	250	128	125	400	1994
Club3D ATI 9550 256MB(128bit)	4	250	128	256	400	2058
Sapphire Atlantis ATI Radeon 9550 128MB VIVO	4	250	128	128	400	2067
Asus A9250GE/TD 256MB	4	240	128	256	400	2094
GigaByte MAYA GV-N55128DP 128MB	4	300	128	128	500	2104
Inno3D GeForce FX-5500 256MB	4	270	128	256	400	2119
Sapphire Atlantis ATI Radeon 9550 256MB	4	250	128	256	400	2157
Inno3D GeForce FX-5500 256MB, VIVO	4	270	128	256	400	2286
Club3D ATI 9600 128MB	4	400	128	128	400	2353
MicroStar NX6200-TD128E 128MB, PCI-E	4	300	128	128	550	2393
Sapphire Atlantis ATI Radeon 9600 128MB	4	325	128	128	400	2418
MicroStar NX6200-TD128 128MB	4	300	128	128	400	2444
Asus A9600GE/TD 128MB	4	325	128	128	400	2453
Asus V9570LE/TD 128MB	4	250	64	128	400	2528
Asus A9550GE/TD 256MB	4	250	128	256	400	2582
Abit Radeon 9550-256CDT 256MB	4	250	128	256	400	2636
Asus V9570GE/TD 128MB	4	250	128	128	400	2756
Sapphire Atlantis ATI Radeon X600Pro 128MB, PCI-E	4	400	128	128	600	2818
Club3D ATI 9600 256MB	4	400	128	256	400	2948
Asus V9520 VideoSuite 128MB	4	250	128	128	400	2983
Sapphire Atlantis ATI Radeon X600Pro 256MB, PCI-E	4	400	128	256	600	3040
Club3D ATI 9600PRO 128MB	4	400	128	128	600	3074
Inno3D GeForce 6200 128MB	4	300	128	128	400	3084
MicroStar RX600XT-TD128E II 128MB, PCI-E	4	500	128	128	550	3214
Asus N6200GE/TD 128MB	4	300	128	128	550	3236
Sapphire Atlantis ATI Radeon 9600XT 128MB	4	500	128	128	600	3286
MicroStar FX5700LE-TD256 256MB	4	250	128	256	400	3311
Asus V9570GE/TD 256MB	4	250	128	256	400	3371
Sapphire Atlantis ATI Radeon 9600XT VIVO	4	500	128	128	600	3402
Sapphire Atlantis ATI Radeon X700 128MB, PCI-E	8	400	128	128	700	3408
Sapphire Atlantis ATI Radeon 9800SE 128MB	4	325	128	128	580	3439
Inno3D GeForce 6600 128MB, AGP	8	300	128	128	550	3482
Sapphire Atlantis ATI Radeon 9600XT 256MB	4	500	128	256	600	3502
Gainward FX PowerPack Ultra/1760 XP TV-DVI 128MB	8	300	128	128	500	3589
Sapphire Atlantis ATI Radeon 9600XT 256MB VIVO	4	500	128	256	600	3669
Sapphire Atlantis ATI Radeon X700 128MB	8	400	128	128	500	3785
Inno3D GeForce 6600 256MB, AGP	8	300	128	256	500	3814

Club3D ATI 9600XT 256MB	4	500	128	256	600	3835
Gainward FX PowerPack Ultra/1760 XP Golden Sample	8	400	128	128	560	3860
MicroStar NX6600-TD128E 128MB, PCI-E	8	300	128	128	550	3885
Asus N6600/TD 128MB	8	300	128	128	550	3889
Asus V9570 TD 128MB	4	425	128	128	500	3922
Gainward FX PowerPack Ultra/1780 TV/DVI 256MB	8	300	128	256	500	3965
Club3D ATI 9600XT 128MB	4	500	128	128	680	4046
Sapphire Atlantis ATI Radeon 9800Pro 128MB	8	380	256	128	680	4081
Sapphire Atlantis ATI Radeon X700 256MB, PCI-E	8	400	128	256	700	4106
MicroStar NX6600-TD256E 256MB, PCI-E	8	300	128	256	500	4186
Sapphire Atlantis ATI Radeon X700 256MB	8	400	128	256	500	4202
MicroStar RX700 PRO-TD128E 128MB, PCI-E	8	420	128	128	860	4276
Asus V9570/TD 256MB	4	425	128	256	500	4373
Asus A9600XT 128MB	4	500	128	128	600	4410
GigaByte MAYA GV-R96X128D 128MB	4	500	128	128	600	4424
Sapphire Atlantis ATI Radeon X700 Fireblade 256MB	8	400	128	256	600	4428
Asus N6600/TD 256MB	8	300	128	256	500	4542
Club3D All-In-Wonder 9800SE 128MB	4	380	256	128	680	4547
MicroStar NX6600-VTD256EH 256MB, VIVO, PCI-E	8	300	128	128	500	4555
GigaByte MAYA GV-R96X128DU 128MB	4	550	128	125	600	4585
Sapphire Atlantis ATI Radeon X700PRO 256MB, PCI-E	8	420	128	256	860	4720
MicroStar NX6600-VTD256 256MB	8	400	128	256	600	4855
Sapphire Atlantis ATI Radeon X700PRO 128MB	8	420	128	128	860	4869
Club3D ATI 9800Pro 256bit 128MB	8	380	256	128	680	4987
MicroStar NX6600-VTD128E Diamond 128MB, VIVO, PC	8	400	128	128	800	5130
Inno3D GeForce 6600GT 128MB, AGP	8	500	128	128	900	5378
Gainward FX PowerPack Ultra/1960 TV-DVI 128MB	8	500	128	128	900	5380
Inno3D GeForce 6600PE 128MB, AGP	8	400	128	128	800	5544
GigaByte MAYA GV-N66T128D 128MB	8	500	128	128	1000	5555
Asus V9999LE/TD 128MB	8	300	256	128	700	5627
MicroStar NX6600GT-TD128E 128MB, PCI-E	8	500	128	128	1000	5669
Sapphire Atlantis ATI Radeon X700PRO 256MB	8	420	128	256	860	5837
MicroStar NX6600-VTD128 Diamond	8	400	128	128	800	5915
MicroStar NX6600GT-VTD128 128MB	8	500	128	128	900	6006
Asus N6600GT/TD 128MB	8	500	128	128	900	6044
MicroStar RX700 PRO-TD256E 256MB, PCI-E	8	420	128	256	860	6315
GigaByte MAYA GV-N66T128VP 128MB	8	500	128	128	1120	6334
Gainward FX PowerPack Ultra/1960 TV-DVI-DVI XP 256	8	500	128	256	900	6349
Gainward FX PowerPack Ultra/1960 XP 128MB, AGP	8	500	128	128	900	6401
Asus N6600GT TOP 128MB	8	520	128	128	1100	6416
MicroStar NX6800LE-TD128 128MB	8	300	256	128	700	6545

Gainward FX PowerPack Ultra/1960 XP Golden Sample	8	540	128	128	1050	6558
Gainward FX PowerPack Ultra/1960 XP 128MB Golden :	8	540	128	128	1150	7396
Gainward FX PowerPack Ultra/2100 TV/DVI 128MB	12	325	256	128	700	7396
MicroStar NX6800-TD128 128MB	12	325	256	128	700	7684
Inno3D GeForce 6800PE 128MB	12	350	256	128	800	7805
Leadtek Winfast A400 TDH 128MB	12	325	256	128	700	8098
Asus V9999/TD 128MB	12	325	256	128	700	8291
Gainward FX PowerPack Ultra/2100 TV/DVI Golden San	12	350	256	128	800	8888
Sapphire Atlantis ATI Radeon X800Pro 256MB	12	475	256	256	900	9162
MicroStar NX6800-TD256E 256MB, PCI-E	12	325	256	256	600	9506
Sapphire Atlantis ATI Radeon X800 Pro 256 MB, PCI-E	12	475	256	256	900	9603
Asus V9999GT/TD 128MB	16	350	256	128	700	9764
Sapphire Atlantis ATI Radeon X800 XL 256 MB, PCI-E	16	400	256	256	980	9773
Excalibur X800Pro IceQ II 256MB	12	475	256	256	900	9872
Asus V9999GT/TD 256MB	16	350	256	256	700	10670
Inno3D GeForce 6800GT 256MB	16	350	256	256	1100	10734
Sapphire Atlantis ATI Radeon X800 XL Ultimate 256MB	16	400	256	256	980	10974
Asus AX800XL/2DTV 256MB	16	400	256	256	980	11071
MicroStar RX800 PRO VTD 256MB VIVO	12	475	256	256	900	11105
MicroStar NX6800GT-TD256 256MB	16	350	256	256	1000	11127
Gainward FX PowerPack Ultra/2400 TV/DVI Golden San	16	400	256	256	1100	11284
Club3D ATI X800 Pro 256MB VIVO	12	475	256	256	900	11401
Gainward FX PowerPack Ultra/2400 TV/DVI 256MB	16	350	256	256	1000	11768
Club3D ATI X800 Pro 256MB	12	475	256	256	900	12198
Asus AX800 PRO/TD 256MB	12	475	256	256	900	12697
Sapphire Atlantis ATI Radeon X800XT 256 MB VIVO	16	500	256	256	1000	13031
Club3D ATI X800 XT 256MB VIVO	16	500	256	256	1000	13245
Sapphire Atlantis ATI Radeon X800XT Platinum 256MB	16	520	256	256	1120	13340
Sapphire Atlantis ATI Radeon X850 XT 256 MB	16	520	256	256	1080	13923
Asus AX850PRO/HTVD 256MB, VIVO	12	520	256	256	1080	14340
Gainward FX PowerPack Ultra/2600 TV-DVI 256MB	16	400	256	256	1100	14530
Sapphire Atlantis ATI Radeon X850 XT 256 MB, VIVO, P	16	520	256	256	1080	14542
Inno3D GeForce 6800 Ultra 256MB	16	400	256	256	1000	14720
MicroStar NX6800Ultra-T2D256 256MB	16	400	256	256	1100	14732
Asus AX800XT 256MB	16	520	256	256	1120	15184
Asus AX800XT/TV 256MB, VIVO	16	520	256	256	1120	15303
Asus AX850XT/HTVD 256MB, VIVO	16	520	256	256	1080	15637
Asus V9999 Ultra/2DT 256MB	16	425	256	256	1100	15637
Sapphire Atlantis ATI Radeon X850 XT 256 MB, VIVO	16	520	256	256	1080	16041
Gainward FX PowerPack Ultra/2600 Golden Sample 256	16	430	256	256	1200	16101
Asus AX850XTP/HTVD 256MB, VIVO	16	540	256	256	1180	16291

Asus V9999 Ultra Deluxe 256MB	16	400	256	256	1100	16291
MicroStar NX6800Ultra-T2D256E 256MB, PCI-E	16	400	256	256	1100	16648
Sapphire Atlantis ATI Radeon X850 XT Platinum 256 MI	16	540	256	256	1180	17552

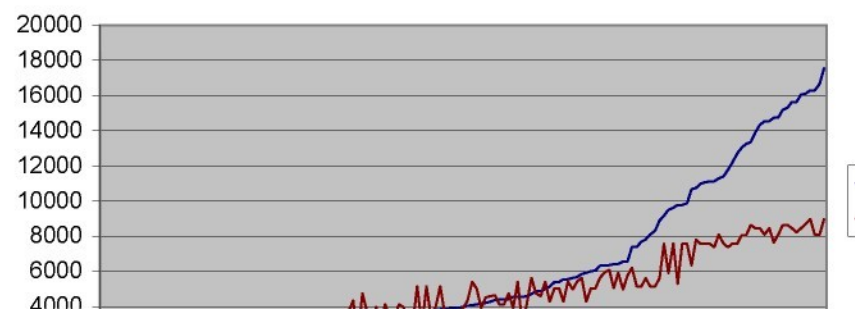
rpipelines	rcip	rsbernice	rvelikost	rrychlost	REGRESE
622,9535894	-2127,830218	1426,452093	1087,819932	1360,785125	349,4453862
1136,223047	204,3333758	1426,452093	1087,819932	712,7543829	686,2719567
1136,223047	2294,650547	776,6296402	1087,819932	2085,040798	1248,828183
2320,19966	204,3333758	1426,452093	1087,819932	1577,051096	859,1312993
2320,19966	2294,650547	1426,452093	3645,655719	1577,051096	1788,761891
1136,223047	2294,650547	1426,452093	3645,655719	1570,168487	1787,385369
1136,223047	2294,650547	1426,452093	3645,655719	1570,168487	1787,385369
2320,19966	2294,650547	1426,452093	3645,655719	2085,040798	1890,359831
2320,19966	1895,934827	1426,452093	1087,819932	2085,040798	1299,04953
2320,19966	1895,934827	1426,452093	3645,655719	2085,040798	1810,616687
2320,19966	1895,934827	1426,452093	3645,655719	2085,040798	1810,616687
2320,19966	204,3333758	1426,452093	3645,655719	1577,051096	1370,698457
2320,19966	1895,934827	1426,452093	3645,655719	2085,040798	1810,616687
2320,19966	2294,650547	1426,452093	3645,655719	1570,168487	1787,385369
2320,19966	2294,650547	1426,452093	3645,655719	2085,040798	1890,359831
2320,19966	1895,934827	3618,100011	3645,655719	2085,040798	2248,946271
2320,19966	1895,934827	3618,100011	3645,655719	2085,040798	2248,946271
2320,19966	1895,934827	3618,100011	3645,655719	2085,040798	2248,946271
2320,19966	1895,934827	3618,100011	3645,655719	2085,040798	2248,946271
2320,19966	2294,650547	1426,452093	3645,655719	2085,040798	1890,359831
2320,19966	2294,650547	3618,100011	3645,655719	1577,051096	2227,091475
2320,19966	1895,934827	3618,100011	3645,655719	2085,040798	2248,946271
2320,19966	3063,060418	1426,452093	3645,655719	2085,040798	2044,041805
2320,19966	4976,664262	1426,452093	3645,655719	2085,040798	2426,762574
2320,19966	2294,650547	1426,452093	3645,655719	2085,040798	1890,359831
2320,19966	1895,934827	3618,100011	8827,976736	2085,040798	3285,410474
2320,19966	5749,745622	3618,100011	1087,819932	3557,082389	2802,549591
2320,19966	1895,934827	3618,100011	3645,655719	2085,040798	2248,946271
2320,19966	1895,934827	3618,100011	3645,655719	2085,040798	2248,946271
2320,19966	2294,650547	3618,100011	3645,655719	2085,040798	2328,689415
2320,19966	3063,060418	3618,100011	3645,655719	2085,040798	2482,371389
2320,19966	204,3333758	1426,452093	3645,655719	2085,040798	1472,296397
2320,19966	2294,650547	1426,452093	3645,655719	2085,040798	1890,359831

pipelines

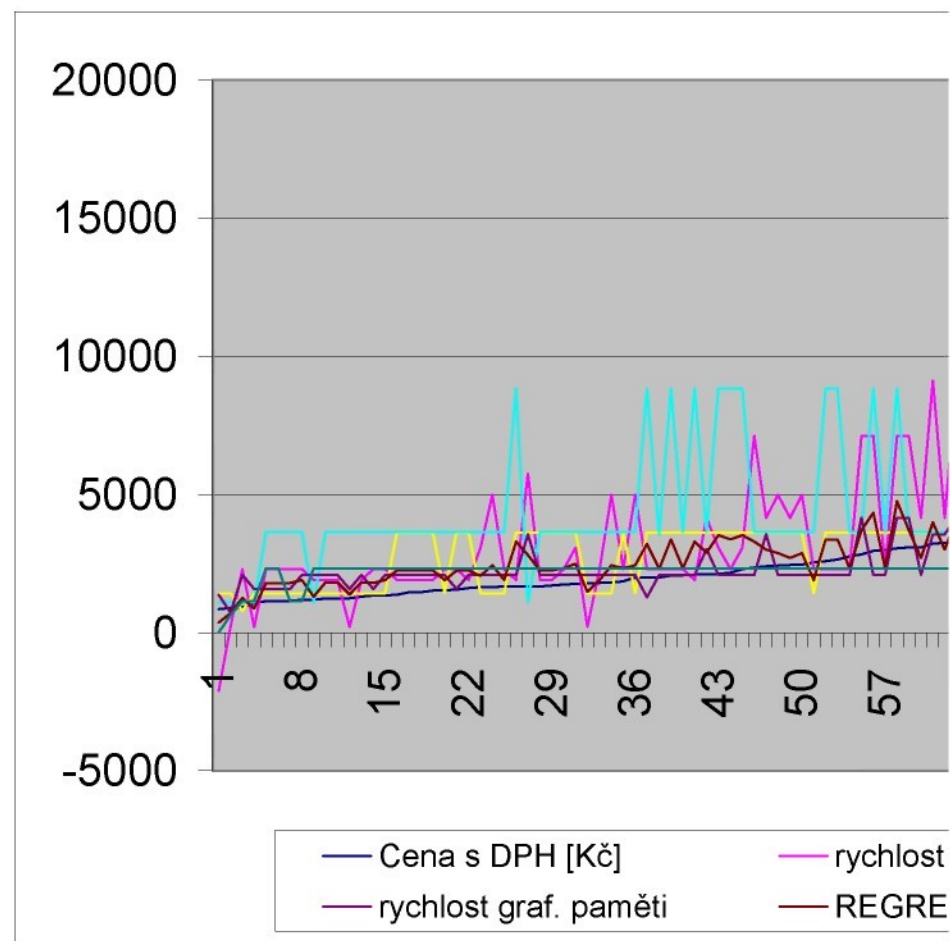
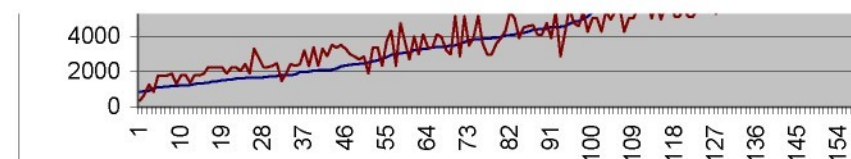
a	b	c	y	
2262849	161633	12801	131689861	22014962544
161633	12801	1229	10003169	
12801	1229	159	895740	
131689861	161633	12801		5,77664E+11
10003169	12801	1229	a=	26,23961612
895740	1229	159		
2262849	131689861	12801		9,56662E+12
161633	10003169	1229	b=	434,5506097
12801	895740	159		
2262849	161633	131689861		3,57002E+12
161633	12801	10003169	c=	162,1633636
12801	1229	895740		

Odhad ceny

Grafická karta	8	500	128	128	900
----------------	---	-----	-----	-----	-----



2320,19966	4976,664262	1426,452093	3645,655719	2085,040798	2426,762574
2320,19966	2294,650547	3618,100011	3645,655719	2085,040798	2328,689415
2320,19966	4976,664262	1426,452093	3645,655719	2085,040798	2426,762574
2320,19966	2294,650547	3618,100011	8827,976736	1275,802938	3203,306046
2320,19966	2294,650547	3618,100011	3525,260876	2085,040798	2304,610446
2320,19966	2294,650547	3618,100011	8827,976736	2085,040798	3365,153618
2320,19966	2294,650547	3618,100011	3645,655719	2085,040798	2328,689415
2320,19966	1895,934827	3618,100011	8827,976736	2085,040798	3285,410474
2320,19966	4143,121296	3618,100011	3645,655719	3014,980781	2884,371561
2320,19966	3063,060418	3618,100011	8827,976736	2085,040798	3518,835592
2320,19966	2294,650547	3618,100011	8827,976736	2085,040798	3365,153618
2320,19966	3063,060418	3618,100011	8827,976736	2085,040798	3518,835592
2320,19966	7114,523526	3618,100011	3645,655719	2085,040798	3292,664011
2320,19966	4143,121296	3618,100011	3645,655719	3557,082389	2992,791883
2320,19966	4976,664262	3618,100011	3645,655719	2085,040798	2865,092158
2320,19966	4143,121296	3618,100011	3645,655719	2085,040798	2698,383565
2320,19966	4976,664262	3618,100011	3645,655719	2085,040798	2865,092158
2320,19966	2294,650547	1426,452093	3645,655719	2085,040798	1890,359831
2320,19966	2294,650547	3618,100011	8827,976736	2085,040798	3365,153618
2320,19966	2294,650547	3618,100011	8827,976736	2085,040798	3365,153618
2320,19966	2294,650547	3618,100011	3645,655719	2085,040798	2328,689415
2320,19966	7114,523526	3618,100011	3645,655719	4150,605074	3705,776866
2320,19966	7114,523526	3618,100011	8827,976736	2085,040798	4329,128214
2320,19966	2294,650547	3618,100011	3645,655719	2085,040798	2328,689415
2320,19966	7114,523526	3618,100011	8827,976736	4150,605074	4742,241069
2320,19966	7114,523526	3618,100011	3645,655719	4150,605074	3705,776866
2320,19966	4143,121296	3618,100011	3645,655719	2085,040798	2698,383565
2320,19966	9118,540065	3618,100011	3645,655719	3557,082389	3987,875637
2320,19966	4143,121296	3618,100011	3645,655719	3557,082389	2992,791883
2320,19966	9118,540065	3618,100011	3645,655719	4150,605074	4106,580174
2320,19966	2294,650547	3618,100011	8827,976736	2085,040798	3365,153618
2320,19966	2294,650547	3618,100011	8827,976736	2085,040798	3365,153618
2320,19966	9118,540065	3618,100011	3645,655719	4150,605074	4106,580174
5317,903672	7114,523526	3618,100011	3645,655719	5491,913676	3974,038586
2320,19966	4976,664262	3618,100011	3645,655719	3907,02547	3229,489092
5317,903672	4143,121296	3618,100011	3645,655719	3557,082389	2992,791883
2320,19966	9118,540065	3618,100011	8827,976736	4150,605074	5143,044377
5317,903672	4143,121296	3618,100011	3645,655719	3014,980781	2884,371561
2320,19966	9118,540065	3618,100011	8827,976736	4150,605074	5143,044377
5317,903672	7114,523526	3618,100011	3645,655719	3014,980781	3478,652007
5317,903672	4143,121296	3618,100011	8827,976736	3014,980781	3920,835765



2320,19966	9118,540065	3618,100011	8827,976736	4150,605074	5143,044377
5317,903672	7114,523526	3618,100011	3645,655719	3671,673239	3609,990499
5317,903672	4143,121296	3618,100011	3645,655719	3557,082389	2992,791883
5317,903672	4143,121296	3618,100011	3645,655719	3557,082389	2992,791883
2320,19966	7706,220069	3618,100011	3645,655719	3014,980781	3596,991316
5317,903672	4143,121296	3618,100011	8827,976736	3014,980781	3920,835765
2320,19966	9118,540065	3618,100011	3645,655719	5207,197211	4317,898601
5317,903672	6597,633935	11569,4079	3645,655719	5207,197211	5403,978953
5317,903672	7114,523526	3618,100011	8827,976736	5491,913676	5010,50279
5317,903672	4143,121296	3618,100011	8827,976736	3014,980781	3920,835765
5317,903672	7114,523526	3618,100011	8827,976736	3014,980781	4515,116211
5317,903672	7592,717689	3618,100011	3645,655719	8065,830804	4584,460845
2320,19966	7706,220069	3618,100011	8827,976736	3014,980781	4633,455519
2320,19966	9118,540065	3618,100011	3645,655719	4150,605074	4106,580174
2320,19966	9118,540065	3618,100011	3645,655719	4150,605074	4106,580174
5317,903672	7114,523526	3618,100011	8827,976736	4150,605074	4742,241069
5317,903672	4143,121296	3618,100011	8827,976736	3014,980781	3920,835765
2320,19966	6597,633935	11569,4079	3645,655719	5207,197211	5403,978953
5317,903672	4143,121296	3618,100011	3645,655719	3014,980781	2884,371561
2320,19966	9757,778701	3618,100011	3525,260876	4150,605074	4210,348932
5317,903672	7592,717689	3618,100011	8827,976736	8065,830804	5620,925048
5317,903672	7114,523526	3618,100011	8827,976736	4150,605074	4742,241069
5317,903672	7592,717689	3618,100011	3645,655719	8065,830804	4584,460845
5317,903672	6597,633935	11569,4079	3645,655719	5207,197211	5403,978953
5317,903672	7114,523526	3618,100011	3645,655719	7038,906588	4283,437169
5317,903672	9118,540065	3618,100011	3645,655719	8791,58381	5034,775921
5317,903672	9118,540065	3618,100011	3645,655719	8791,58381	5034,775921
5317,903672	7114,523526	3618,100011	3645,655719	7038,906588	4283,437169
5317,903672	9118,540065	3618,100011	3645,655719	10749,94534	5426,448227
5317,903672	4143,121296	11569,4079	3645,655719	5491,913676	4970,019718
5317,903672	9118,540065	3618,100011	3645,655719	10749,94534	5426,448227
5317,903672	7592,717689	3618,100011	8827,976736	8065,830804	5620,925048
5317,903672	7114,523526	3618,100011	3645,655719	7038,906588	4283,437169
5317,903672	9118,540065	3618,100011	3645,655719	8791,58381	5034,775921
5317,903672	9118,540065	3618,100011	3645,655719	8791,58381	5034,775921
5317,903672	7592,717689	3618,100011	8827,976736	8065,830804	5620,925048
5317,903672	9118,540065	3618,100011	3645,655719	13371,48247	5950,755653
5317,903672	9118,540065	3618,100011	8827,976736	8791,58381	6071,240124
5317,903672	9118,540065	3618,100011	3645,655719	8791,58381	5034,775921
5317,903672	9403,25709	3618,100011	3645,655719	12913,99118	5916,200801
5317,903672	4143,121296	11569,4079	3645,655719	5491,913676	4970,019718

5317,903672	9649,278688	3618,100011	3645,655719	11806,25772	5743,858428
5317,903672	9649,278688	3618,100011	3645,655719	14073,14572	6197,236027
9155,2754	4976,664262	11569,4079	3645,655719	5491,913676	5136,728311
9155,2754	4976,664262	11569,4079	3645,655719	5491,913676	5136,728311
9155,2754	5749,745622	11569,4079	3645,655719	7038,906588	5600,743166
9155,2754	4976,664262	11569,4079	3645,655719	5491,913676	5136,728311
9155,2754	4976,664262	11569,4079	3645,655719	5491,913676	5136,728311
9155,2754	5749,745622	11569,4079	3645,655719	7038,906588	5600,743166
9155,2754	8708,228339	11569,4079	8827,976736	8791,58381	7579,439357
9155,2754	4976,664262	11569,4079	8827,976736	4150,605074	5904,930794
9155,2754	8708,228339	11569,4079	8827,976736	8791,58381	7579,439357
13832,31484	5749,745622	11569,4079	3645,655719	5491,913676	5291,344584
13832,31484	7114,523526	11569,4079	8827,976736	10341,81829	7570,74529
9155,2754	8708,228339	11569,4079	8827,976736	8791,58381	7579,439357
13832,31484	5749,745622	11569,4079	8827,976736	5491,913676	6327,808787
13832,31484	5749,745622	11569,4079	8827,976736	12913,99118	7812,224288
13832,31484	7114,523526	11569,4079	8827,976736	10341,81829	7570,74529
13832,31484	7114,523526	11569,4079	8827,976736	10341,81829	7570,74529
9155,2754	8708,228339	11569,4079	8827,976736	8791,58381	7579,439357
13832,31484	5749,745622	11569,4079	8827,976736	10749,94534	7379,41512
13832,31484	7114,523526	11569,4079	8827,976736	12913,99118	8085,179869
9155,2754	8708,228339	11569,4079	8827,976736	8791,58381	7579,439357
13832,31484	5749,745622	11569,4079	8827,976736	10749,94534	7379,41512
9155,2754	8708,228339	11569,4079	8827,976736	8791,58381	7579,439357
9155,2754	8708,228339	11569,4079	8827,976736	8791,58381	7579,439357
13832,31484	9118,540065	11569,4079	8827,976736	10749,94534	8053,174009
13832,31484	9118,540065	11569,4079	8827,976736	10749,94534	8053,174009
13832,31484	9403,25709	11569,4079	8827,976736	13371,48247	8634,424839
13832,31484	9403,25709	11569,4079	8827,976736	12464,72727	8453,073799
9155,2754	9403,25709	11569,4079	8827,976736	12464,72727	8453,073799
13832,31484	7114,523526	11569,4079	8827,976736	12913,99118	8085,179869
13832,31484	9403,25709	11569,4079	8827,976736	12464,72727	8453,073799
13832,31484	7114,523526	11569,4079	8827,976736	10749,94534	7652,370701
13832,31484	7114,523526	11569,4079	8827,976736	12913,99118	8085,179869
13832,31484	9403,25709	11569,4079	8827,976736	13371,48247	8634,424839
13832,31484	9403,25709	11569,4079	8827,976736	13371,48247	8634,424839
13832,31484	9403,25709	11569,4079	8827,976736	12464,72727	8453,073799
13832,31484	7706,220069	11569,4079	8827,976736	12913,99118	8203,519178
13832,31484	9403,25709	11569,4079	8827,976736	12464,72727	8453,073799
13832,31484	7817,303985	11569,4079	8827,976736	15283,72133	8699,681991
13832,31484	9649,278688	11569,4079	8827,976736	14793,32056	8967,996776

13832,31484	7114,523526	11569,4079	8827,976736	12913,99118	8085,179869
13832,31484	7114,523526	11569,4079	8827,976736	12913,99118	8085,179869
13832,31484	9649,278688	11569,4079	8827,976736	14793,32056	8967,996776

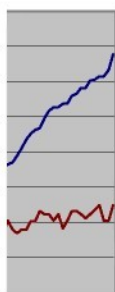
cip

a	b	c	y	
4201947543125	9529819375	22731125	162198721725	3,44563E+18
9529819375	22731125	57835	372496155	
22731125	57835	159	895740	
162198721725	9529819375	22731125		-1,66663E+17
372496155	22731125	57835	a=	-0,048369285
895740	57835	159		
4201947543125	162198721725	22731125		2,19048E+20
9529819375	372496155	57835	b=	63,57252147
22731125	895740	159		
4201947543125	9529819375	162198721725		-3,6439E+22
9529819375	22731125	372496155	c=	-10575,39954
22731125	57835	895740		

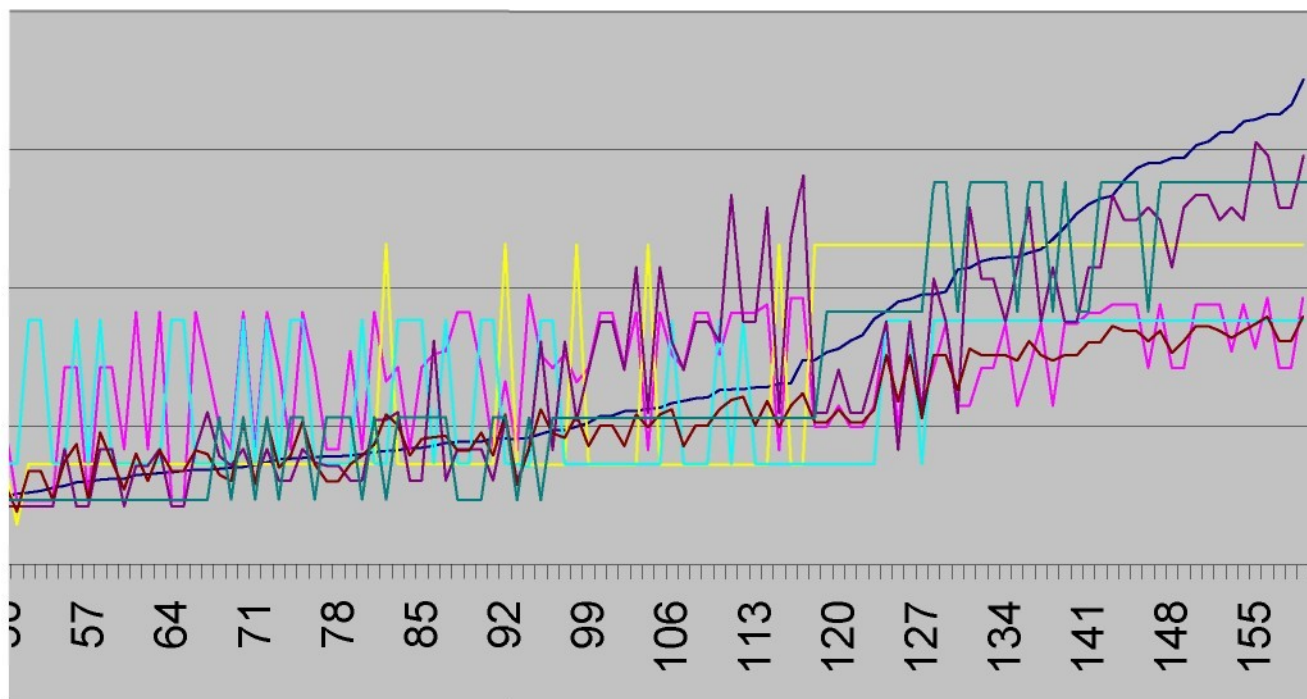
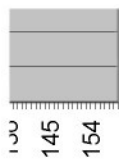
sbernice

a	b	c	y	
225872707584	979140608	4617216	40987723776	2,68799E+17
979140608	4617216	24800	182081984	
4617216	24800	159	895740	
40987723776	979140608	4617216		3,90249E+16
182081984	4617216	24800	a=	0,145182782
895740	24800	159		
225872707584	40987723776	4617216		1,71209E+18
979140608	182081984	24800	b=	6,369404554
4617216	895740	159		
225872707584	979140608	40987723776		1,14009E+20
979140608	4617216	182081984	c=	424,1415256
4617216	24800	895740		

Cena				
5317,903672	9118,540065	3618,100011	3645,655719	8791,58381 6098,356656



— Cena s DPH [Kč]
— REGRESE



— rychlost graf. čipu	— šířka pamět. sběrnice	— velikost graf. paměti
— REGRESE	— pipelines	

velikost

a	b	c	y	
298820736162	1261673162	5675538	42360663627	8,44843E+16
1261673162	5675538	28154	186534631	
5675538	28154	159	895740	
42360663627	1261673162	5675538		2,29119E+14
186534631	5675538	28154	a=	0,002711973
895740	28154	159		
298820736162	42360663627	5675538		3,33252E+18
1261673162	186534631	28154	b=	39,44548541
5675538	895740	159		
298820736162	1261673162	42360663627		-1,22316E+20
1261673162	5675538	186534631	c=	-1447,799374
5675538	28154	895740		

rychlost

a	b	c	y	
64693427640564	69178228204	80129280	702162218013	9,62996E+20
69178228204	80129280	104470	759037205	
80129280	104470	159	895740	
702162218013	69178228204	80129280		9,90366E+18
759037205	80129280	104470	a=	0,010284215
895740	104470	159		
64693427640564	702162218013	80129280		4,19923E+19
69178228204	759037205	104470	b=	0,043605894
80129280	895740	159		
64693427640564	69178228204	702162218013		4,06504E+23
69178228204	80129280	759037205	c=	422,1239626
80129280	104470	895740		

