

Experimentální finanční trh

Marek Bryša

Masarykova Univerzita

10. dubna 2013

Experimentální ekonomie je vědní disciplína, sloužící k odpovědi na ekonomické otázky metodou experimentů konaných na lidech v kontrolovaných podmínkách. Experiment má pevná pravidla, která jsou účastníkům předem sdělena. Z chování účastníků je obvykle sestaven datový soubor, který je následně statisticky analyzován.

Experimentální trh dvojité aukce

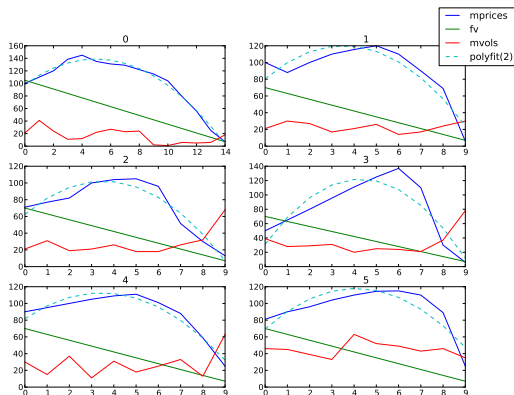
Každý účastník obdrží obnos virtuálních peněz (\$) a několik virtuálních akcií blíže nespecifikované společnosti. Experiment probíhal v předem stanoveném počtu kol r . Na začátku každého kola má účastník možnost podat tyto příkazy k obchodování s akcemi:

- Prodat nejvýše SOV (sell order volume) akcií za nejméně SOP \$ (sell order price)
- Koupit nejvýše BOV akcií za nejvýše BOP \$

Jakmile všichni účastníci zadají své příkazy, sestaví software křivky nabídky a poptávky, určí jejich průnik (tj. tržní cenu) tak, aby byl maximalizován tržní objem za dodržení všech podmínek všech příkazů.

Na konci každého kola navíc akcie vyplatí fixní dividendu d . To znamená, že akcie má jasně definovanou fundamentální hodnotu $FV = d \cdot (r - k)$, kde k je pořadí aktuálního kola. Toto, stejně jako skutečnost, že akcie jsou na konci experimentu bezcenné, ovšem není účastníkům explicitně sděleno.

Experimentální data



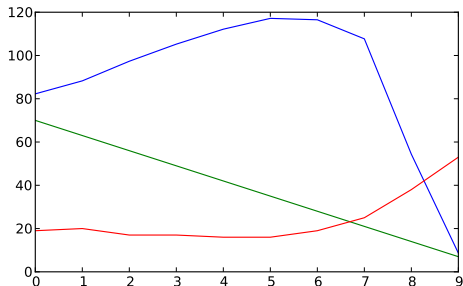
Grafy průběhu tržní ceny, fundamentální hodnoty, tržního objemu a aproximace tržní ceny kvadratickou funkcí (OLS) v šesti běžích experimentu. Je patrná cenová bublina.

Na základě analýzy dostupných dat byl sestaven software pro agentovou simulaci ekonomického experimentu finančního trhu.

Podle odpozorovaného chování účastníku experimentu byly implementovány následující čtyři obchodní strategie:

- RandomUnif Náhodné obchody
- Momentum Sledování trendu
- Fv Sledování fundamentální hodnoty
- Bubble Snaha o výtěžek na bublině

Navíc u prvních dvou strategií může náhodně v posledních 0 až 3 kolech dojít k odprodeji všech akcií za ceny blízké fundamentální hodnotě. To má simulovat prudký pokles ceny v závěru, který jsme experimentálně pozorovali.



Průběh ceny aritmeticky zprůměrovaný z 1 000 běhů simulace se zastoupením jednotlivých strategií (3,8,2,4)

Existují lepší strategie? Povede jejich použití k zefektivnění trhu?

Využijeme strojové učení.

Genetické algoritmy jsou jedno z odvětví evolučních algoritmů. Obzvláště dobře je lze využít na optimalizační problémy. Vychází z principu pozorovaného v přírodě, že zdatní jedinci se množí a předávají svoji genetickou výbavu potomkům, zatímco neúspěšní vymírají, a to vše při vlivu náhodných událostí.

Agent s LCS funguje v prostředí s diskretním časem, rozděleným na kola. LCS agenta je funkce ve smyslu části počítačového programu, jejímž vstupem je číselně zakódovaný stav prostředí, ve kterém se v daném kole nachází a výstupem číselně zakódované akce, které má agent provést. Tato funkce se časem může vyvíjet a měnit k dosažení lepších výsledků.

Předpis funkce LCS je zejména tvořen konečnou množinou pravidel *Rules*. Každé pravidlo $r \in Rules$ je uspořádaná trojice $r = (Input, Output, Fitness)$, kde *Input* je uspořádaná i -tice vstupních bitů, *Output* uspořádaná o -tice výstupních bitů a $Fitness \in \mathbb{R}$ zdatnost pravidla. Každý bit může nabývat hodnot z předem definovaného oboru, typicky množiny $\{1, 0\}$, konečné podmnožiny množiny celých čísel, popř. pouze u vstupních bitů speciální hodnoty $\#$, která znamená *libovolná hodnota*. Obecně se v množině pravidel může vyskytovat více pravidel se stejnou n -ticí vstupních i výstupních bitů.

$$R_1 = \{((1, 0, -5, \#), (20, -1, 0), 1500), \quad (r1)$$

$$((1, \#, -5, 3), (18, 0, 1), 2000), \quad (r2)$$

$$((0, 1, -5, \#), (3, -6, 0), 3000)\}. \quad (r3)$$

Pak funkce LCS reaguje na ukázkové vstupy (neboli stavy) takto:

- $(1, 0, -5, 8) \rightarrow r1:(20, -1, 0)$, $\#$ funguje jako zástupce libovolné hodnoty.
- $(1, 0, -5, 3)$ Tomuto vstupu odpovídají pravidla $r1$ i $r2$. Je tedy třeba rozhodnout, které z nich se zvolí. Možných postupů je několik: vždy vybrat to s nejvyšší zdatností, vybrat náhodné nebo náhodné proporcionálně ke zdatnosti.
- $(0, 0, 10, 3)$ Tomuto vstupu neodpovídá žádné pravidlo. Je možno buď vrátit předem definovanou akci *nedělej nic*, nebo vytvořit nové pravidlo s tímto vstupem a náhodným výstupem – tzv. *automatch*. V modulu LCS je *automatch* implementován, ale zde nikde není využit.

Agent v každém kole provede následující:

- 1 Vypočte stav, ve kterém se trh nachází.
- 2 Předá jej jako vstup funkci LCS, která vrátí výstup ve formě zakódované akce.
- 3 Provede akci jako operaci na trhu.

Vstupní bity:

- 1 round Kolo, ve kterém se tržní experiment nachází $\in \{0, \dots, 9\}$ a hodnota u pravidla # není povolena.
- 2 `linfit_moving` $\in \{-3, \dots, 3, \#\}$
- 3 `linfit_whole` $\in \{-3, \dots, 3, \#\}$
- 4 `last_p` Poslední tržní cena, rovnoměrně zobrazena na hodnoty z intervalu $\langle 0, 200 \rangle$ na množinu $\{0, \dots, 40\}$. U pravidla je povolena hodnota #.

Např. stav $(7, -2, 1, 16)$ odpovídá situaci v 8. kole, kde pohyblivý průměr tržních cen výrazněji klesá, celkový stále ještě mírně roste a cena je v intervalu $\langle 77.5, 82.5 \rangle$.

Výstupní bity:

- 1 BOP Maximální nákupní cena. Hodnota náleží množině $\{0, \dots, 40\}$. Pro získání maximální nákupní ceny příkazu se hodnota vynásobí 5.
- 2 BOV Nákupní objem. Hodnota náleží množině $\{0, 1, 2\}$. Protože agent nemá ve vstupních bitech informaci o stavu držných peněz ani akcií, je tato hodnota relativní. Agent vně své LCS funkce ovšem tyto informace má. 0 znamená nekupuj nic, 1 nakup podle BOP a aktuálního držení peněz za polovinu maxima, 2 za celé maximum.
- 3 SOP Minimální prodejní cena, interpretace je analogická BOP.
- 4 SOV Prodejní objem, interpretace je analogická BOV.

Zdatnost je měřítkem kvality pravidla. Nepřímo slouží k určení toho, které pravidlo bude eliminováno a nahrazeno nějakou kombinací zdatnějších. Ve zdatnostní funkci tak musí být zakódován cíl optimalizačního problému, který řešíme. Tím je v našem případě dlouhodobá maximalizace množství peněz na konci, tedy zisku. Lze si taktéž představit i jiné cíle, např. maximalizaci zisku za určitého omezení rizika, snahu neprodělat atd. Je třeba ale mít na paměti, že tyto případné protichůdné cíle je nutno transformovat do jednoho reálného čísla tak, aby se pravidla dala lineárně uspořádat.

Profit sharing plan – pro každé pravidlo r_i , které bylo v běhu použito:

$$Fitness_i := Fitness_i - b \cdot Fitness_i + b \cdot (Profit \cdot (1 - hash_cost)^{num_hash}),$$

kde $b = 0.025$, $hash_cost = 0.03$ je cena za $\#$ a num_hash je počet $\#$ ve vstupních bitech.

Úprava zdatnosti pravidel popsaná probíhá po každém běhu experimentu, ale křížení a mutace, které ze zdatnosti vychází, se koná až po řádově několika tisících bžích tak, aby se mohla zdatnost pravidel dostatečně projevit a zpřesnit. Konkrétně probíhá každých průměrně 2 000 bžů v intervalech určených exponenciálním rozdělením:

$$W_{GA} \sim \text{Exp}(\lambda = \frac{1}{2000}) \Rightarrow E[W_{GA}] = 2000$$

Počet pravidel byl empiricky zvolen na 200. V literatuře je také popsána možnost změny tohoto počtu v průběhu evoluce, ale to se neukázalo jako nutné.

Nezdatná pravidla

Pravidla jsou nejprve seřazena vzestupně podle zdatnosti. Pak je jejich nejméně zdatná část, určená poměrem `unfit_ratio`, označena jako nezdatná a vstupuje do fáze křížení a mutace.

LeBaron et al. pracují s konstantním `unfit_ratio = 0.2`, to znamená, že by bylo zvoleno vždy 40 nejhorších pravidel. Ukázalo se, že tento přístup vede k nestabilitě množiny pravidel, jakmile se začne přibližovat svému potenciálu, tedy jistému maximu zisku, kterého je možno s daným LCS dosáhnout. Pak totiž i dolních 20% pravidel hraje svou roli a jejich neustálé změny výsledky nezlepšují. K vyřešení toho problému byla vytvořena jednoduchá metoda adaptivního `unfit_ratio`. Ta funguje na principu lineárního poklesu `unfit_ratio` např. z 0.2 při průměrném zisku pod 2 000 za posledních 1000 kol na 0.0 při průměrném zisku nad 2 900, zapsáno jako čtveřice (2000, 0.2, 2900, 0.0).

Pro každé pravidlo, které bylo označeno jako nezdatné, je s pravděpodobností 0.9 provedeno křížení a 0.1 mutace. Princip křížení spočívá v tom, že se ze dvou zdatných pravidel (rodičů) vytvoří jedno (potomek), u kterého je šance, že bude zdatnější než rodiče. Mutace je jistý prvek náhody, který zajišťuje pestrost pravidel a objevení zcela nových prvků.

Při křížení jsou dva rodiče vybráni ze zdatných pravidel pomocí normalizovaného proporcionálního náhodného výběru. Vstupní i výstupní bity nezdatného pravidla jsou jeden po druhém zcela nahrazeny odpovídajícími bity z pravidel jednoho z rodičů, a to se stejnou pravděpodobností v případě binárních bitů. V případě celočíselných bitů, což jsou zde všechny, je postup o něco složitější:

- S $p = 0.3$ vezmi bit z prvního rodiče.
- S $p = 0.3$ vezmi bit z druhého rodiče.
- S $p = 0.4$ dále:
 - Pokud je bit alespoň jednoho z rodičů roven #, vezmi bit jednoho z rodičů se stejnou pravděpodobností.
 - Pokud jsou odpovídající bity obou rodičů různé od #, vezmi jejich aritmetický průměr a zaokrouhli ho.

V případě mutace je ze zdatných pravidel vybrán jeden rodič pomocí normalizovaného proporcionálního náhodného výběru. Vstupní a výstupní bity nezdatného pravidla jsou nahrazeny těmi z rodiče a dále je postupně každý bit s pravděpodobností 0.5 nahrazen náhodnou hodnotou z jeho oboru hodnot, popř. #.

V případě křížení i mutace je zdatnost nezdatného pravidla ponechána. Pravidlo tak musí dokázat svou zdatnost následujících bězích, jinak bude opět vyřazeno při příštím spuštění genetického algoritmu.

Za zmínku dále stojí, že mutace musí být poměrně vzácným jevem. Byla testována i možnost pravděpodobnosti mutace 0.5, což vedlo k celkovému selhání tohoto evolučního postupu.

Zobecnění nepoužívaných pravidel

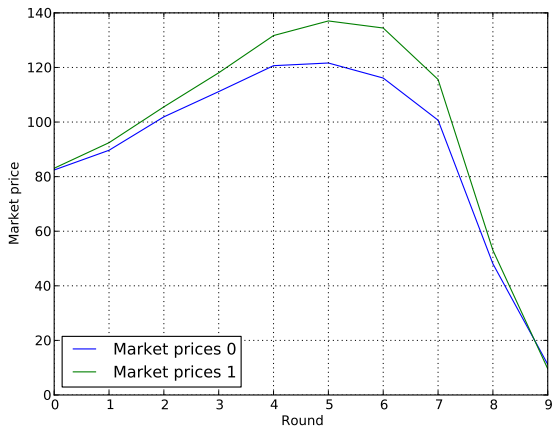
Nežádoucím jevem je výskyt pravidel, jejichž vstupní bity zřídka odpovídají stavu systému a jsou tedy nepoužívána. Pokud by se jich systém nezbavoval, po čase se stane, že žádná pravidla nejsou využívána a evoluce ustrne. Proto je každý bit pravidla, které nebylo použito déle než 300 běhů, s pravděpodobností 0.2 nahrazen náhodnou hodnotou z jeho oboru hodnot, popř. #. Při testování se tato hranice 300 běhů jeví jako optimální – menší číslo vede k eliminaci užitečných, ale zřídka používaných pravidel, větší pak ke zpomalení, či částečnému ustrnutí evoluce.

Lepší strategie?



Nejvyšší udržitelný zisk až na úrovni 2 950, přičemž nejlepší neevoluční strategie dosahuje na 2 300.

Efektivnější trh?



Způsobuje ještě větší bublinu. Důvodem je přítomnost trendových agentů, kteří by na reálném trhu zkrachovali.

To be continued...
Děkuji za pozornost!