

平成 28 年度 3 回生後期実験 (エージェント)

課題 4 入札エージェントの作成

村田 叡

2016/11/18

1 プログラム概要

今回のレポートではオークションで予測値を元に入札するエージェントを作成した。

1.1 プログラムの起動方法及び実行例

readme.md の項 requirements を参照のこと

2 外部仕様

エージェントは Python3 で実装した。以下、その Python3 の外部インターフェイスについて述べる

2.1 py3/multiagent.py

このコードでは、多エージェントによるオークションシミュレートを実装している。オークションデータの csv ファイル、エージェントの種類を引数にとり、オークションを行う。

エージェントは以下の 4 種類を用意している。

1. 一日目の購入価格の中央値で購入し続ける単純エージェント @simple
2. 一日目の購入価格の中央値の 5 倍の価格で購入し続ける貪欲エージェント @greedy
3. 一つ前の購入価格を次の価格の予測値として利用して戦略的に購入するエージェント @sorena
4. SVR に基づいて次の価格を予測し、戦略的に購入するエージェント @svr

例えば、id0001.csv を用いて、貪欲エージェントと SVR エージェントを戦わせるには

python3 py3/multi_agent.py sample_data/id0001.csv @greedy @svr のようにすればよい。

--show-process を引数に加えると、購入過程を逐一表示する。この購入過程は、例えば 0 145.16 13.138 については、左から 購入者 id(0)、購入された時間 (145.16)、落札価格 (13.138) を表す。購入者 id は、csv のデータの人が買えば -1 を、0 以上であれば、それぞれエージェントを順に表す。例えば、python3 py3/multi_agent.py sample_data/id0001.csv @greedy @svr @simple --show-process によって実行すれば、@greedy は 0 番、@svr は 1 番、@simple は 2 版である。SVR の予測値については、ガウスカーネルで、一日目のデータを元に求めた C , σ の値を元にして、最近のデータから毎度 SVR を作成

して予測している。

3 内部仕様

3.1 py3/visualize.py

このコードでは、データをプロットして可視化するために必要な関数を実装している。CUI インターフェースは無い。3D データを表示する関数、SVR の予測がどのようなものなのかを検証する関数を提供する。

3.2 py3/auction.py

このコードでは、オークションデータを取り扱う Auction クラスを実装している。CUI インターフェースは無い。Auction クラスは、コンストラクタに CSV ファイルをとってデータを読み込み、可視化する関数や、価格の配列を取り出す関数などを提供している。

3.3 py3/multiagent.py

このコードでは、内部的には購入に携わる Buyer クラス及び購入戦略に基づいて購入する Agent クラスを実装している。

3.3.1 Buyer クラス

最初に所持金額を与え、buy メソッドにより購入を行うことができるような抽象化をしている。

3.3.2 Agent クラス

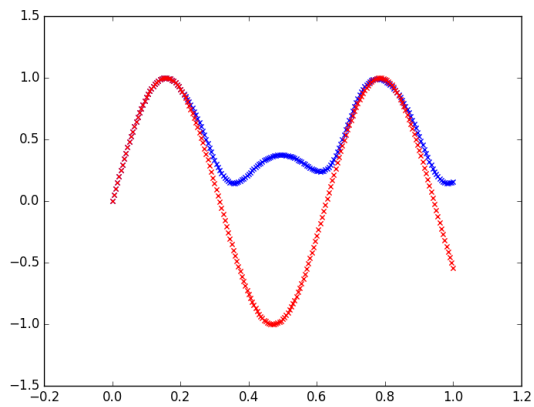
一日目のデータ、一日目の購入価格の中央値、Buyer クラスインスタンスを元にして作成する。buy_から始まる関数では、各エージェントがその関数が表す戦略に基づいた購入価格を出力する。do_multi_auction 関数によって、実際に多エージェントによるオークションをシミュレートする。

4 考察

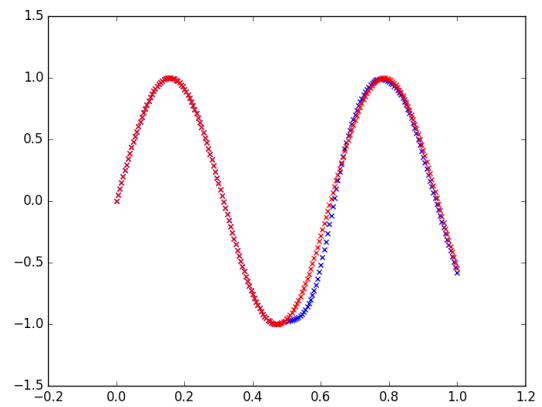
4.1 SVR の予測精度について

SVR 自体の能力を過信せず、一度検証してみることが大事である。予測精度を可視化するために、ある関数 $f(x)$ を定義し ($x \in [0, 1]$), 与えるデータの範囲を $[0, b](0 < b < 1)$ とすることで予測がどのように変化するかを考察する。

4.1.1 sin 波

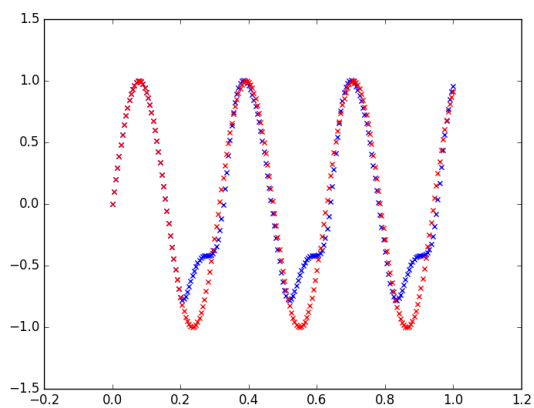


(a) $b = 0.2$

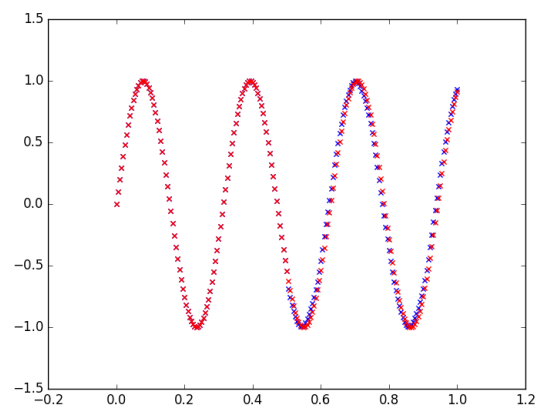


(b) $b = 0.5$

図 1: $f(x) = \sin(10x)$ による予測値 (赤:実値, 青:予測値)



(a) $b = 0.2$



(b) $b = 0.5$

図 2: $f(x) = \sin(10x)$ による予測値 (赤:実値, 青:予測値)

sin 波は、とてもわかりやすい規則性があるため、少ないサンプル数でもほどよい予測ができる事がわかる