# TD and ETD for time series predictions

## Reinforcement Learning Course

Claudio Sole - Greta Laage

Avril 2017

# Framework

## Methods

- Learning To Predict approach (Sutton,1988)
- TD and supervised learning comparison
- ETD trys

Observation-outcome sequences : $\mathbf{x}_1, .., \mathbf{x}_m$,z
Estimates of $z$ at each time step: $P_1, .., P_m$
Prediction function: $P_t = P(\mathbf{x_t}, \omega)$

Learning: update $\omega$

$$\omega = \omega + \sum_{t=1}^{m} \Delta\omega_t$$

# Learning To Predict approach

## Supervised learning

$\Delta\omega_t = \alpha(z - P_t)\nabla_w P_t$

## TD method

$(z - P_t) = \sum_{k=t}^{m}(P_{k+1} - P_k)$ with $P_{m+1} = z$
$TD(\lambda)$: Exponential weighting with recency
$\Delta\omega_t = \alpha(P_{t+1} - P_t)\sum_{k=1}^{t}\lambda^{t-k}\nabla_\omega P_k$

## TD update rules

$$\omega = \omega + \alpha(P_{t+1} - P_t)\mathbf{e}_t$$

$$\mathbf{e}_t = \nabla_\omega P_t + \lambda\mathbf{e}_{t-1}$$

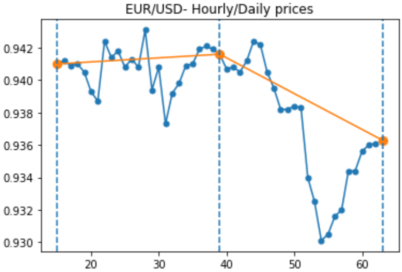
## Convergence results

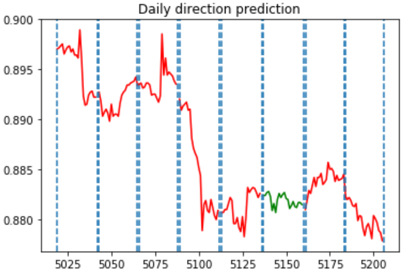
$$P(x_i, w_n) \longrightarrow \mathbb{E}[z|i]$$

Convergence to the true expected value of outcome $z$ starting from $i$

# Financial time series

- Fractal nature of financial time series $\Leftrightarrow$ easy to switch from single-step to multi-step prediction view (and viceversa)

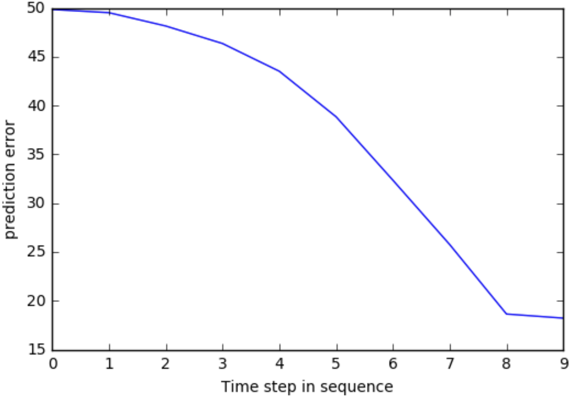



- **Approach:** Neural network for function approximation. INPUT include: past prices, past values of MAs 10-20,50,200, and other derivated from MA

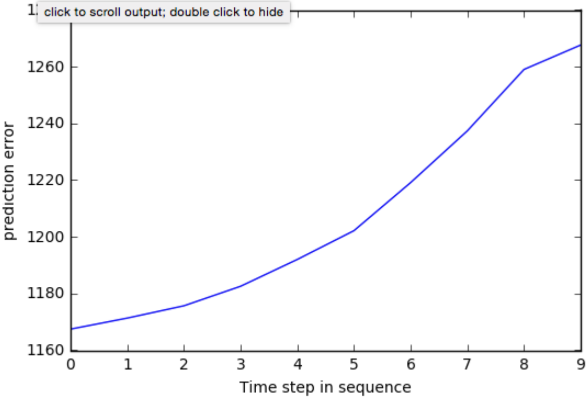
# Predict the opening price

- Linear function approximation: $P(x_t, \omega) = \omega^T \phi(s)$
- Basic features and more *complicated* features
- Adapted ETD update rules



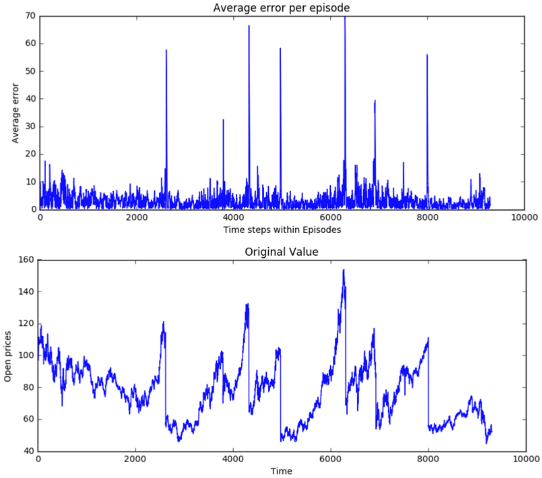
# Predict the opening price

- Linear function approximation: $P(x_t, \omega) = \omega^T \phi(s)$
- Basic features and more *complicated* features
- Adapted ETD update rules



# Predict the opening price

- Linear function approximation: $P(x_t, \omega) = \omega^T \phi(s)$
- Basic features and more *complicated* features
- Adapted ETD update rules



?