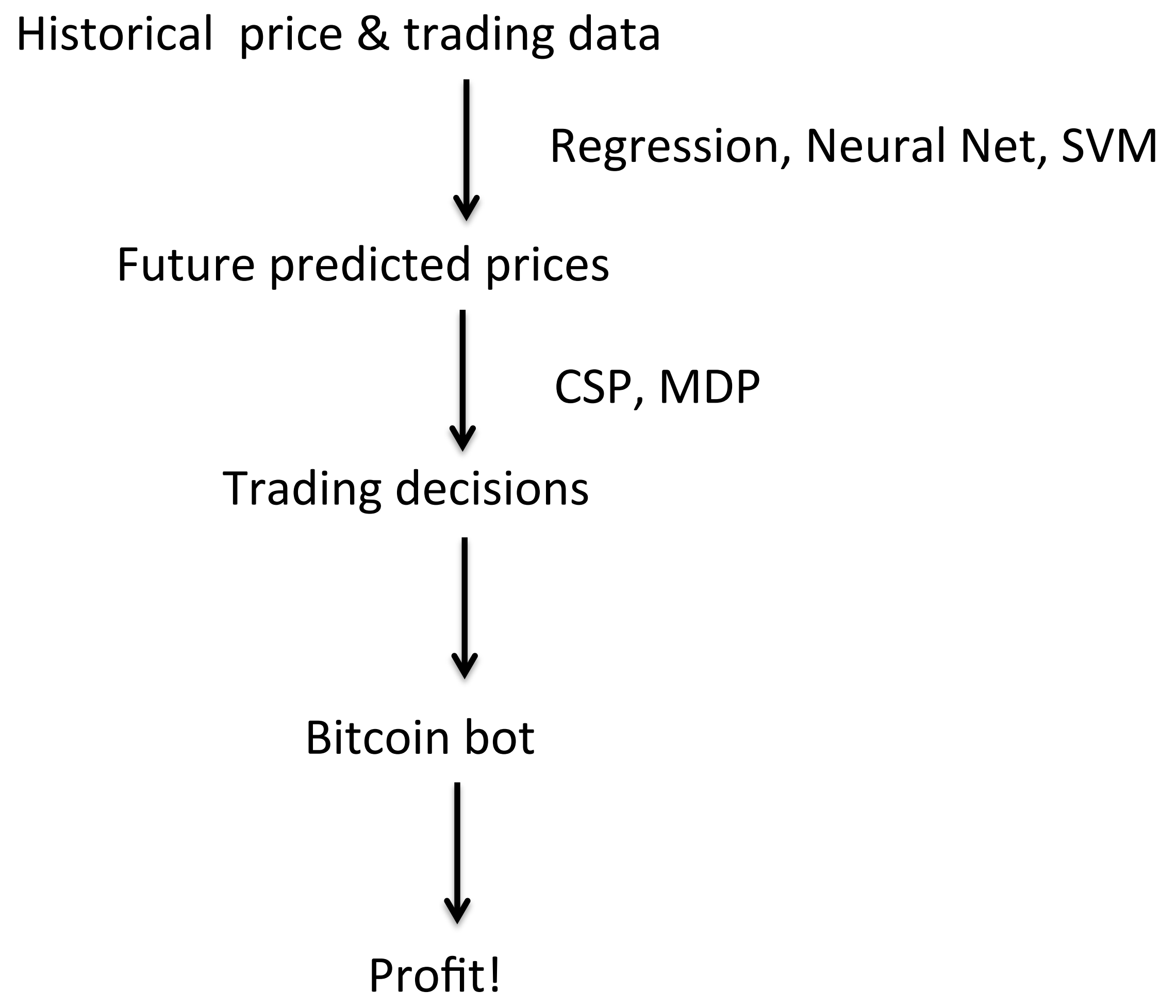
How to Succeed in BitCoin Without Really Trying

Abaho Katabarwa, Guoxing Li, Ellen Sebastian

Introduction

Bitcoin is the world's first decentralized cryptocurrency. In this project, we take advantage of the vast amount of transactional data surrounding Bitcoin to create a Bitcoin trading bot. This bot will maximize profit by predicting future Bitcoin prices.

We split the task into two phases: price prediction and trading decisions. Since trading decisions can be made easily given perfect price predictions, the main challenge in this project isto generate accurate price predictions.



Historical price & trading data

Regression, Neural Net, SVM

Future predicted prices

CSP, MDP

Trading decisions

Bitcoin bot

Profit!

Data Acquisition & Processing

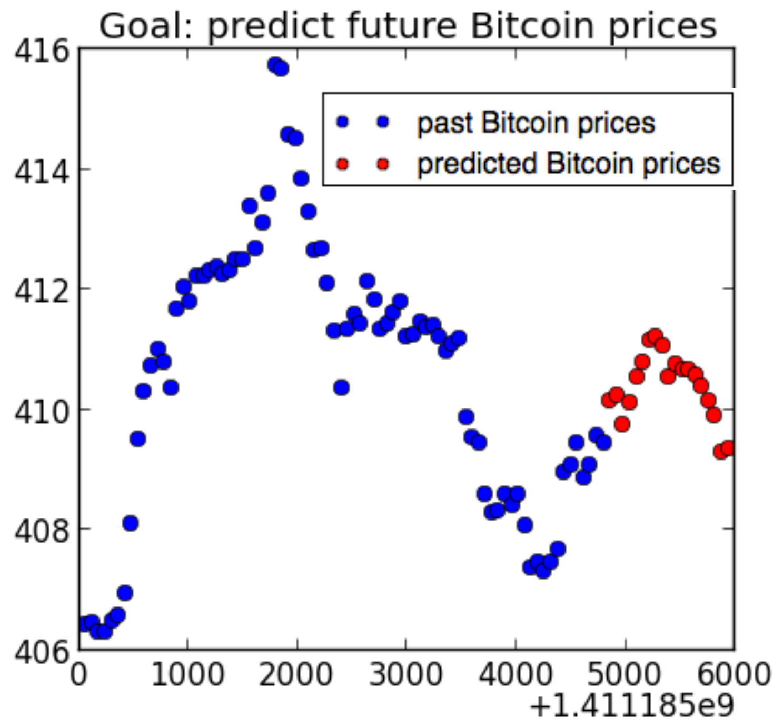
- Minute-granularity historical prices
- Various Bitcoin statistics, e.g. market volume
- Sampled order book data → buy/sell ratios
- Missing data was cubic-spline interpolated or imputed when appropriate

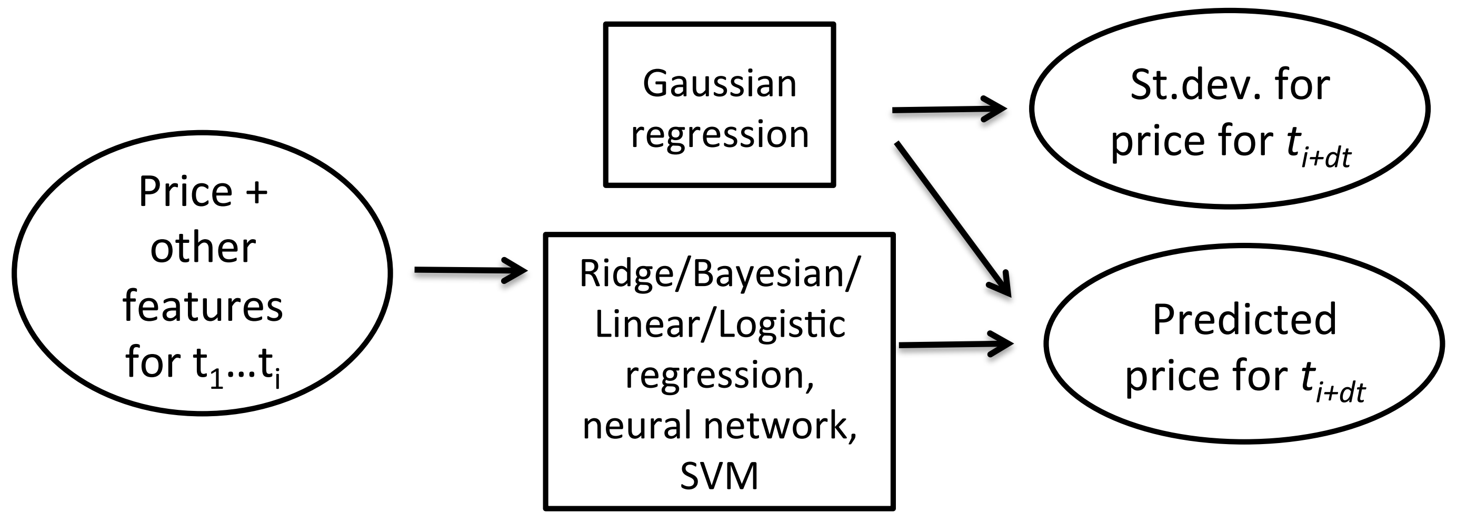
Analysis

- CSP chooses the optimal action given perfect price predictions
- Neural network is the best price prediction method
- Price prediction is poor overall, achieving maximum 56% accuracy
- Price prediction continues to be the most difficult and most important aspect of bitcoin bot creation

Price prediction algorithms & results

Goal: predict *n* future Bitcoin price changes over *dt* from past bitcoin prices and other data





Price + other features for $t_1…t_i$

Gaussian regression

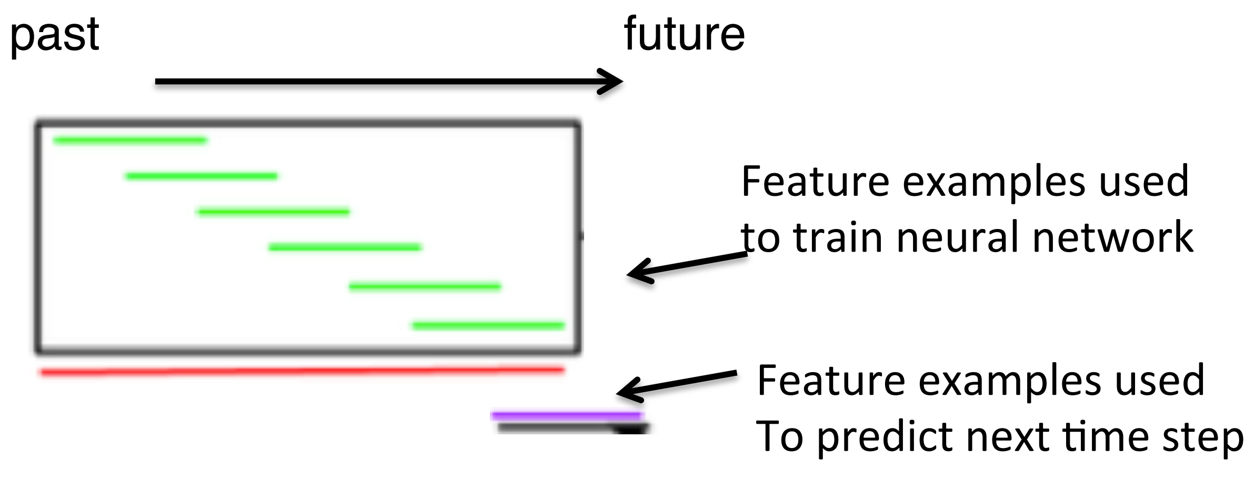
Ridge/Bayesian/ Linear/Logistic regression, neural network, SVM

St.dev. for price for t_{i+dt}

Predicted price for t_{i+dt}

Neural Networks

Our most effective method of price Prediction was a Neural Network In which we trained the network on Multiple windows of past prices and Used a final window to predict an Unseen future price.



past

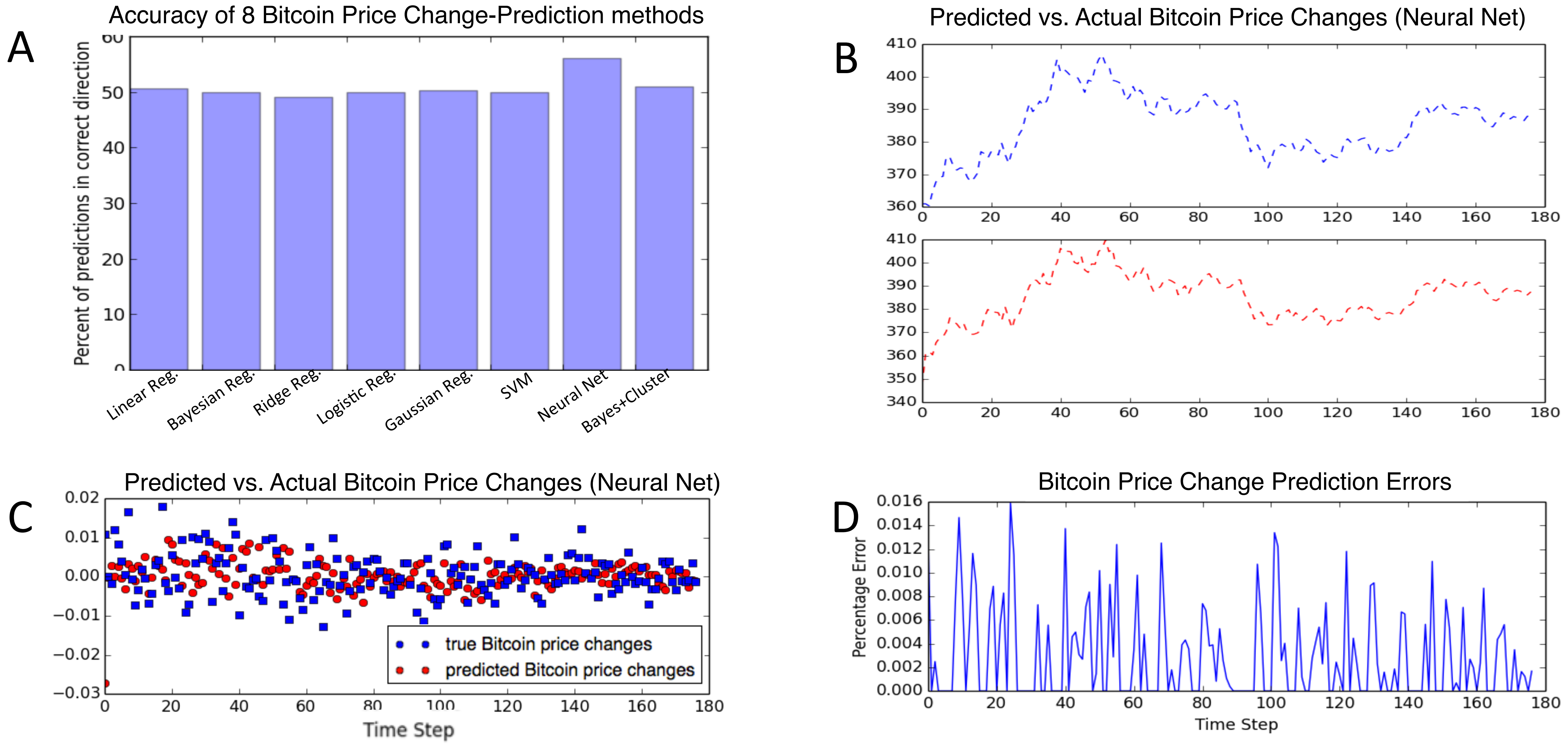
future

Feature examples used to train neural network

Feature examples used To predict next time step

Regression

- Gaussian regression outputs a standard deviation for the prediction (useful for MDP transition probabilities).
- We combined Bayesian regression with k-means trend clustering learn from hidden Bitcoin price trends.



Bitcoin price prediction errors. (A) Different methods of price prediction achieve 50-56% accuracy, where an "accurate" prediction has the same sign as the actual price change. (B) Price predictions (derived from price change predictions) vs. Actual bitcoin price. They follow the same trend due to the small change at each time step. (C) Predicted vs. actual price changes for SVM; other algorithms are similar. (D) Price change prediction errors; other algorithms are similar. Errors are roughly uniform across examples.

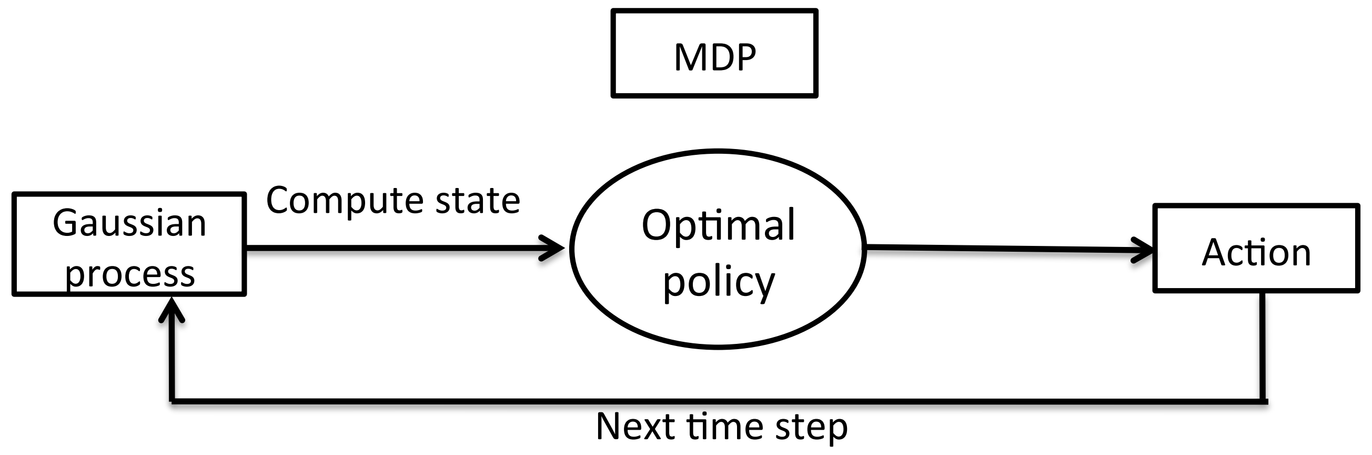
Trading Decision Algorithms

We framed the trading decision as a recommendation of the optimal time(s) to sell a fixed number of Bitcoins within an allotted amount of time.

Markov Decision Process

Standard deviations come from Gaussian regression in the Price Prediction stage.

- States: (time remaining, bitcoins remaining, pricediff (predicted price – bought price), stdev of predicted price)
- Actions: {sell i bitcoins, do nothing}
- Rewards(s, a, s): -#bitcoins sold in action a * s[pricediff]
- Transition probabilities: T(s,a,s) ~ N(s[pricediff],s[stdev]^2)
- IsEnd(s): s[time remaining] = 0



MDP

Gaussian process

Compute state

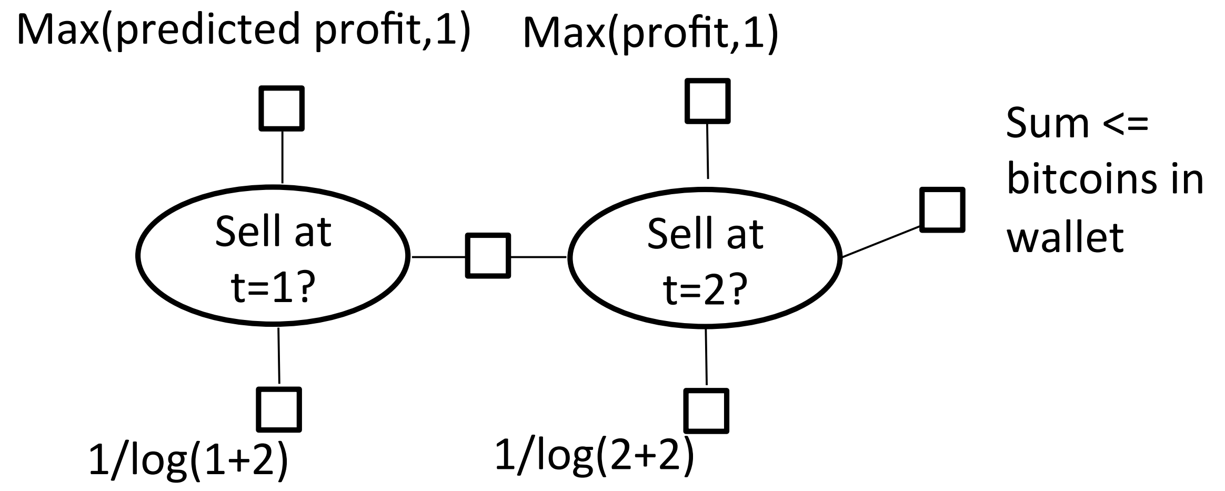
Optimal policy

Action

Next time step

Constraint Satisfaction Problem

- Variables: number of bitcoins to sell at each timestamp.
- Potential: Do not sell more Bitcoins than you own.
- Potential: timestamps with more certain (earlier) predictions are weighted higher: 1/log(timestamp+2)
- Potential: timestamps with higher predicted prices have higher weight=max(predicted profit,1)
- Re-calculate CSP when new data received.



Max(predicted profit,1)

Max(profit,1)

Sum <= bitcoins in wallet

Sell at t=1?

Sell at t=2?

1/log(1+2)

1/log(2+2)

Future work

- The main objective of future work will be to improve price prediction.
- Take the second derivative of price in order to predict its first derivative.
- Use a binary classifier to predict negative or positive changes.
- Include more non-price data; continue searching for granular data.
- Integrate with Bitcoin trading APIs to create bot