Introduction to Quant Trading

Summary

Brief Overview of Methodology

- Alpha Model (Forecasting returns for the next time horizon) using available datasets:

  (Actual earnings (company reports),

  EPS - projections (IBES),

  Price Volume data (TAQ US)

- Risk Model - Statistical Approach (PCA), Fundamental Approach (BARRA)
- Market Impact Model
- Portfolio Construction Optimization
- Attribution Analysis

Alpha Models in Real World (1)

- Finding an edge:

  New data sources

  Better algorithms

  Speed of processing information (short term momentum factors)

  Providing service (liquidity, reversion factors)

Alpha Models in Real World (2)

Analyzing new data sets

- Identify features of the data
- Sanity check, look for outliers
- Understand methodology of data collection and availability in real time

  IBES → 1) monthly updates (academicians) , 2) intraday updates (actual trading)

- Coverage of the trading universe
- Survivorship bias

Alpha Model in Real World (3)

Example of Alpha Factor Mean Reversion:

“i” - stock index i=1...N

Alpha[i, t] = rank[log[P(i,t-1)/P(i, t)]] (robust)

Alpha[i,t] -> Alpha[i,t] - <Alpha[i,t]>

Alpha[i, t] → 2*Alpha[i, t] / Sum[Abs[Alpha[i, t]] , i = 1...N]

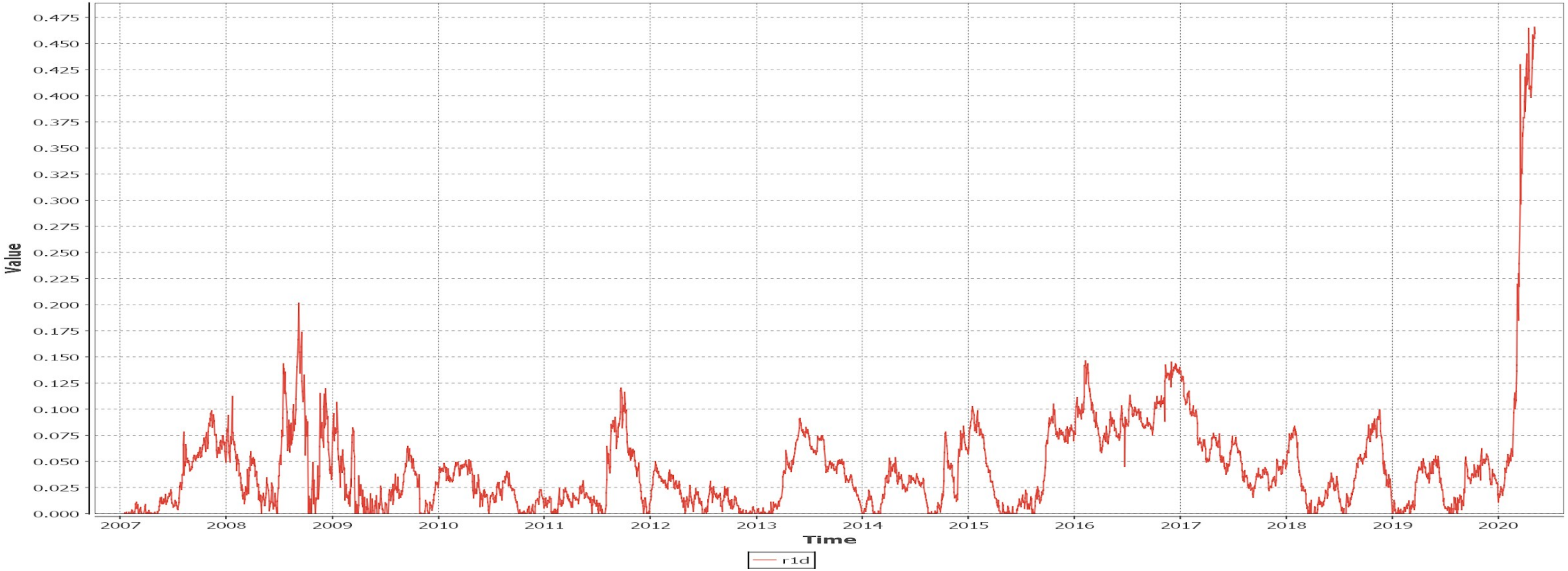
Evaluate efficacy of alpha factor using 1 day forward returns

Alpha Factors in Real World (4)

Factor Performance and drawdowns

Alpha Factors in Real World (5)

Drawdowns of 1 day mean reversion



Alpha Factors in Real World (6)

Q: Can we improve the factor shown before?

A: Yes, use algorithms to improve signal to noise ratio

CAPM/APT/Risk Model

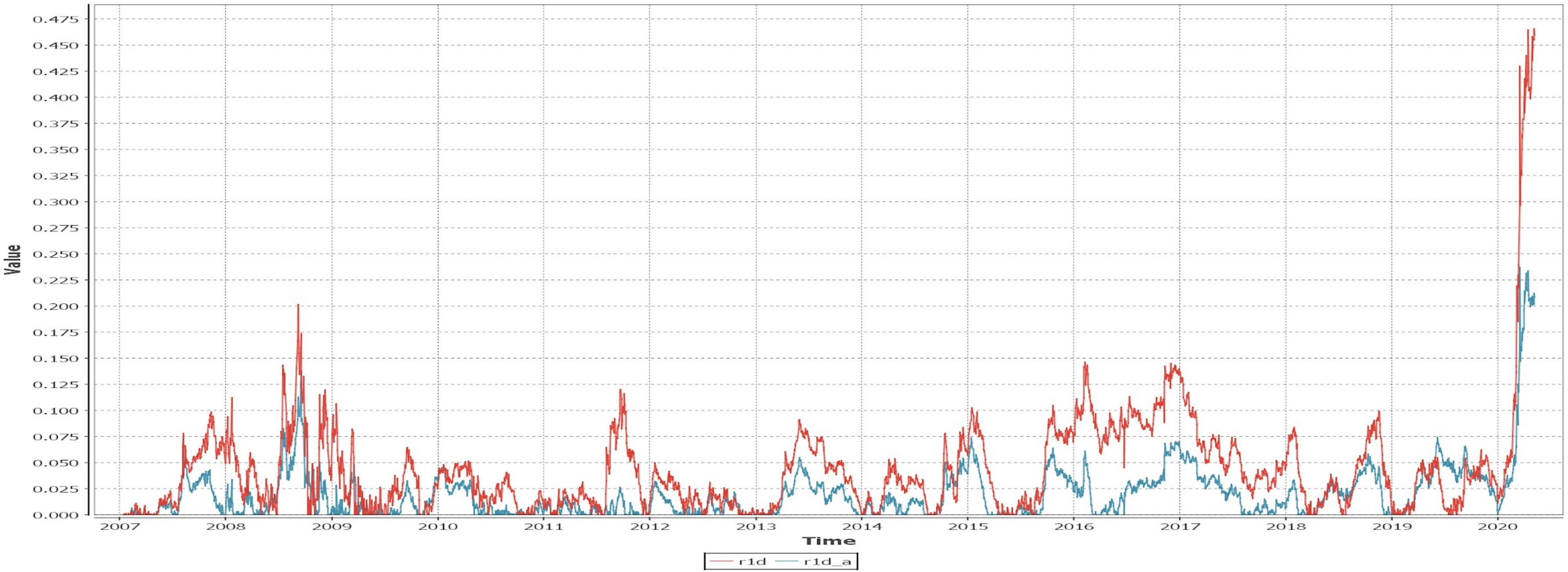
Orthogonalize factor defined before using risk model.

Alpha Factors in Real World (7)

R1d_a - alpha factor orthogonal to risk model

Alpha Factors in Real World (8)

Drawdowns:



Alpha Factors in Real World (9) Momentum

1 year momentum = rank[log[P(i,t)/P(i,t-252)]] L1norm = 2

Alpha Factors in Real World (9)

Realistic Alpha Models:

“ k “ - alpha factor index, k = 1…. NAlphaFactors

Linear Models

CombinedAlpha[i,t] = Sum[w[k]*Alpha[i,t, k], k = 1… NAlphaFactors]

Linear Models: FutureReturn(i,t+TimeHorizon) = CombinedAlpha[i,t]

Nonlinear Models (Machine Learning/Deep Learning) CombinedAlpha=f(Alpha)

Questions:

1) Time horizons (alpha realization profiles)
2) Alpha Factor’s turnovers

Alpha Factor Turnover/Effective Time Horizons

Definitions:

A[i,t] -> 2*A[i,t] / sum[Abs(A[i,t]), i=1...N]]

Turnover(t) = Sum[Abs[A[i,t]-A[i,t-1]], i=1….N]

- Alpha Factors with higher turnover hard to capture during actual trading
- Consistent Alpha Factors with low turnover are hard to find

More examples of Alpha Factors

Attribution Analysis

Alpha Model

Alpha Model (aka Forecast Model) attempts to forecast returns for all the instruments in your trading universe.

The common approach to the problem is to identify reasons for possible inefficiency and come up with a set of features that allow to forecast residual/excess returns.

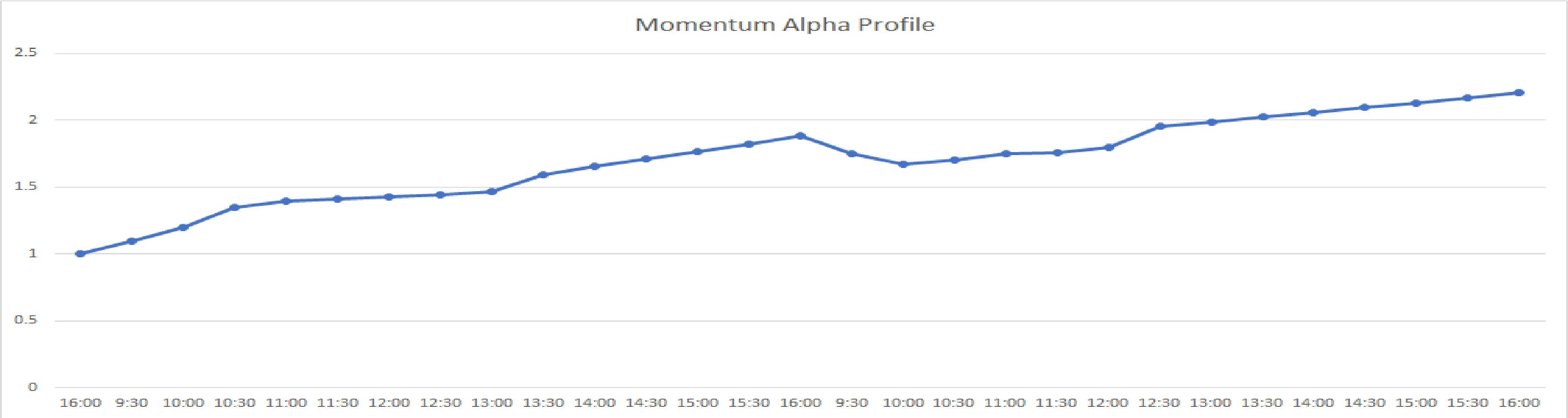
Philosophically there are few reasons for market inefficiencies:

1. Cost of processing new information. Different market participants respond to the new information with different rate. ---> “Momentum” factors
2. Liquidity events/overreaction (Providing service) ---> “Reversion” factors

Alpha Factors and Alpha Profiles (Momentum)

Classification of Alpha Factors:

- “Momentum” factors. The positive changes in price are followed by positive changes and vice versa.
- Examples: a) 1 Year Momentum
- b) Changes in Analyst recommendations / fundamentals



Risk Models

- Statistical Approach (PCA)

Decompose Covariance Matrix into eigenvectors

Cov[i,j] = Sum[F[i,k]*F[j,k]*Eigenvalue[k], k = 1 …. NriskFactors]

- Fundamental Approach BARRA

Use intuition about market structure to construct factor loadings

R[i,t] = Sum(FL[i,k]*L[k,t], k = 1… NriskFactors) + …. (ALpha(i,t))

Market Impact Model

Parameters:

- ADV - projected average volume during trading interval
- Relative Spread = (Ask - Bid) / Mid
- Volatility = (Max(P)-Min(P))/CloseP

TradingSlippage (relative to Entry Price) = SpreadCrossing + MarketImpact

SpreadCrossing = SpreadCrossCoefficient*Relative Spread * dollarsTraded

MarketImpact = MarketImpactCoef*Volatility*f(dollarsTraded/ADV)*dollarsTraded

f(x) ~ x, for small x (x<0.05)

f(x) ~ sqrt(x), for large x (x>=0.05)

Optimization (Portfolio Construction)

All the ingredients are used as inputs:

- Forecast (Alpha Model)
- Risk Model
- Market Impact Model

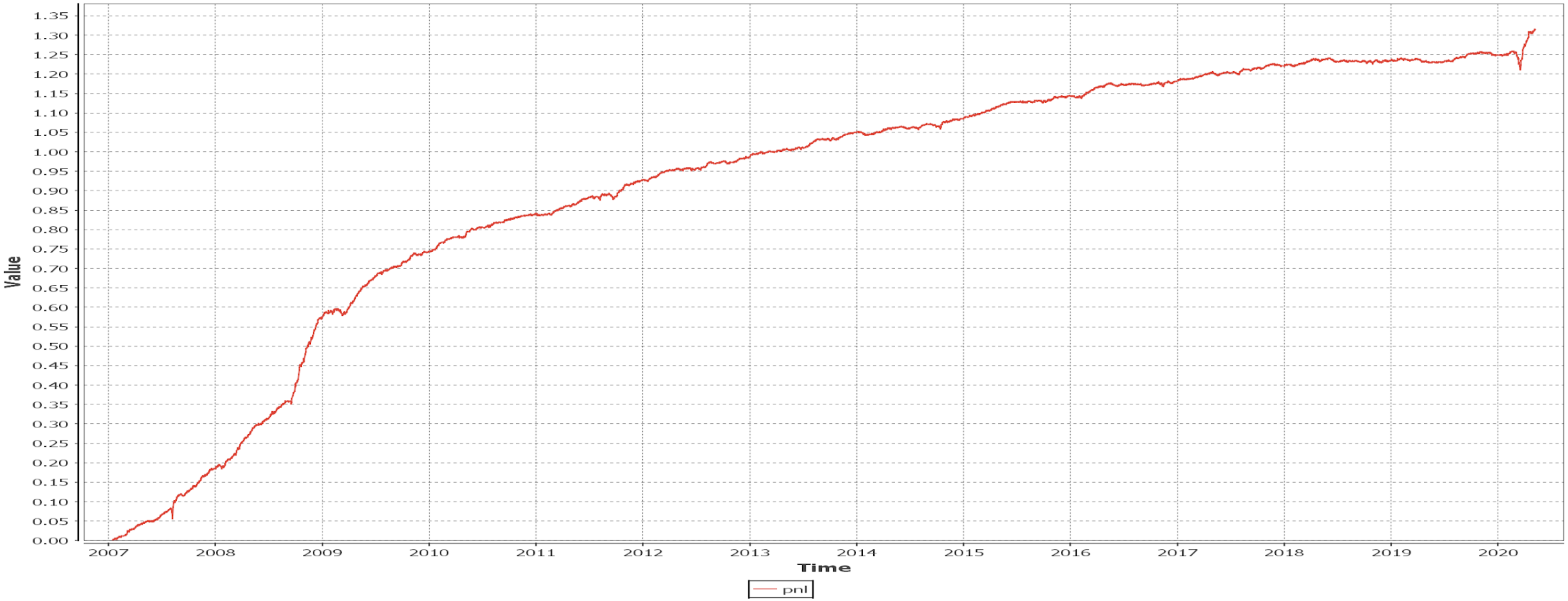
Outputs: desired positions (weights at next time interval “t+1”)

Max(Sum[Alpha[i,t]*w(i,t+1), i=1...N] - TradingCost)

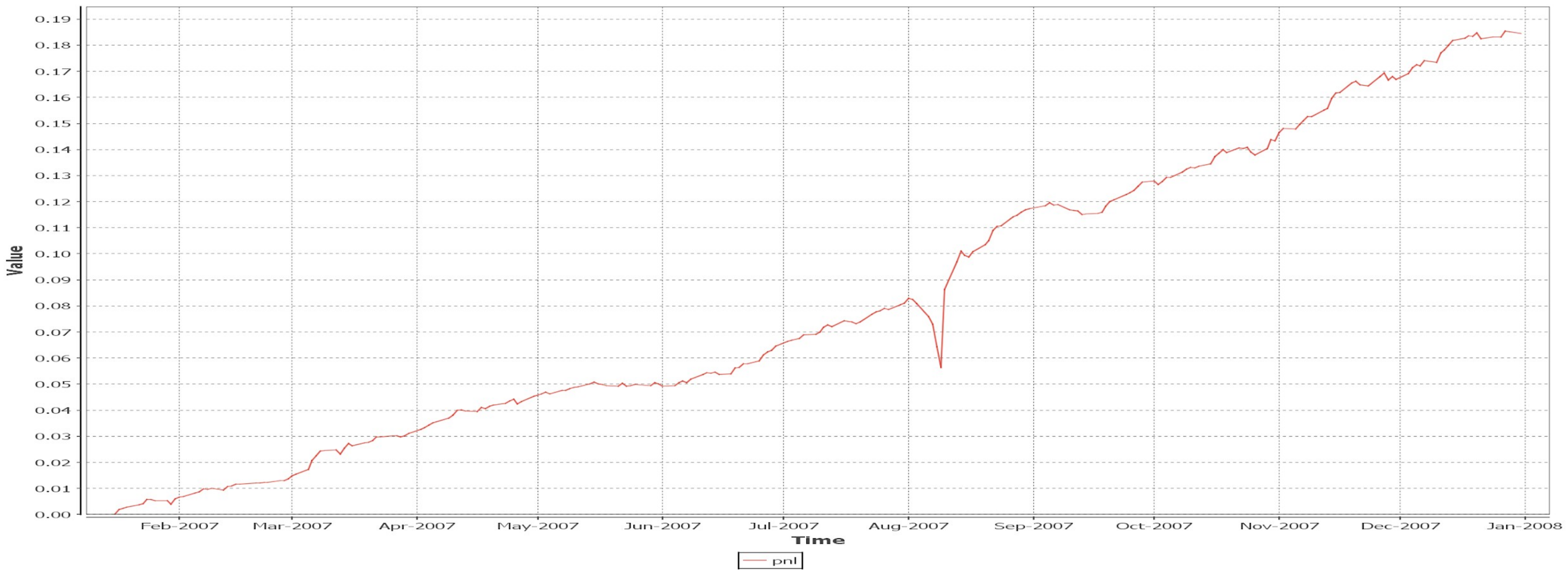
Subject to risk constraints: |w(i,t+1)| < PosLimitWeight

Sum[FL[i,k]*w(i,t+1), i=1...N] < C[k]

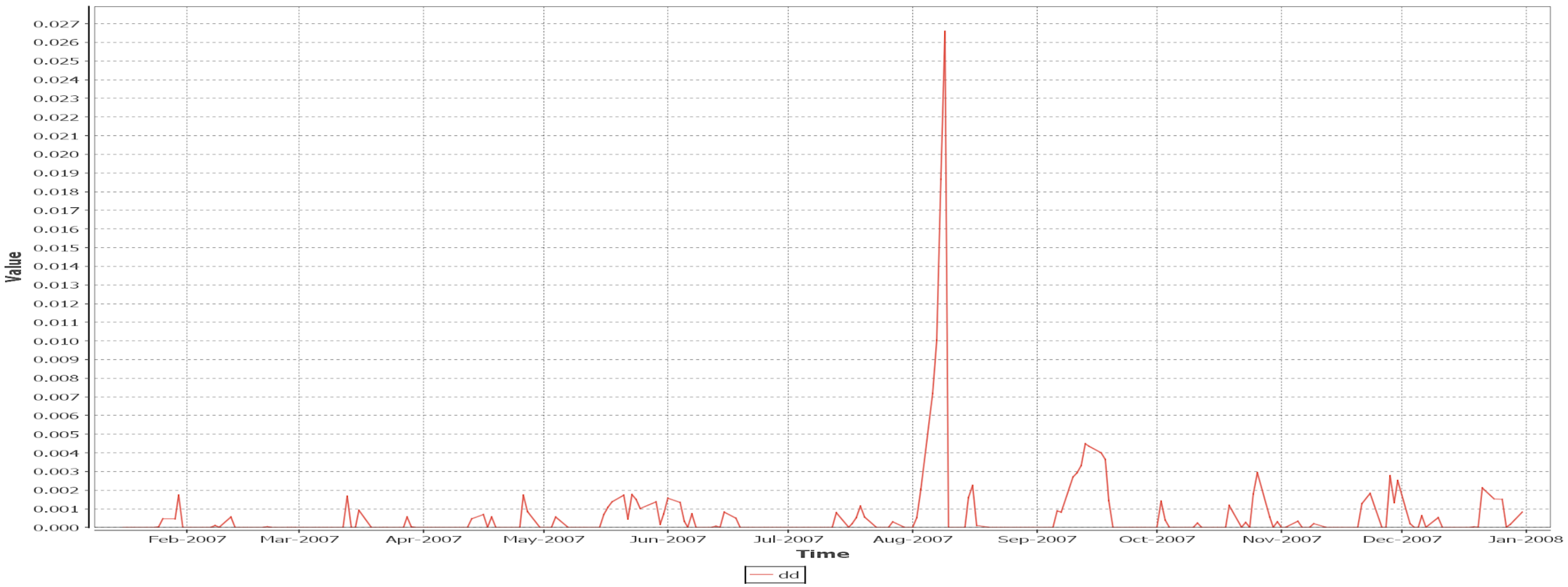
Example of realistic trading strategy



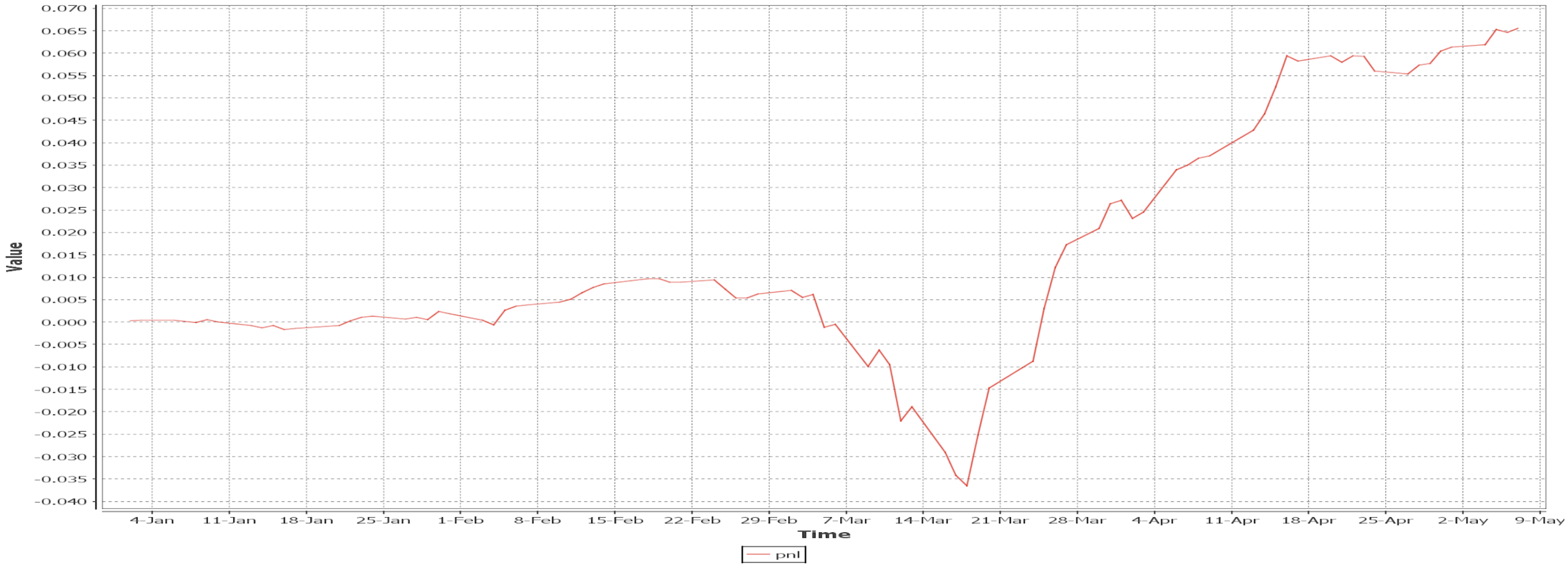
Example of Realistic Trading Strategy (2)



Example of Trading Strategy (3)



Example of Real Trading Strategy (3)



Example of Real Trading Strategy (4)