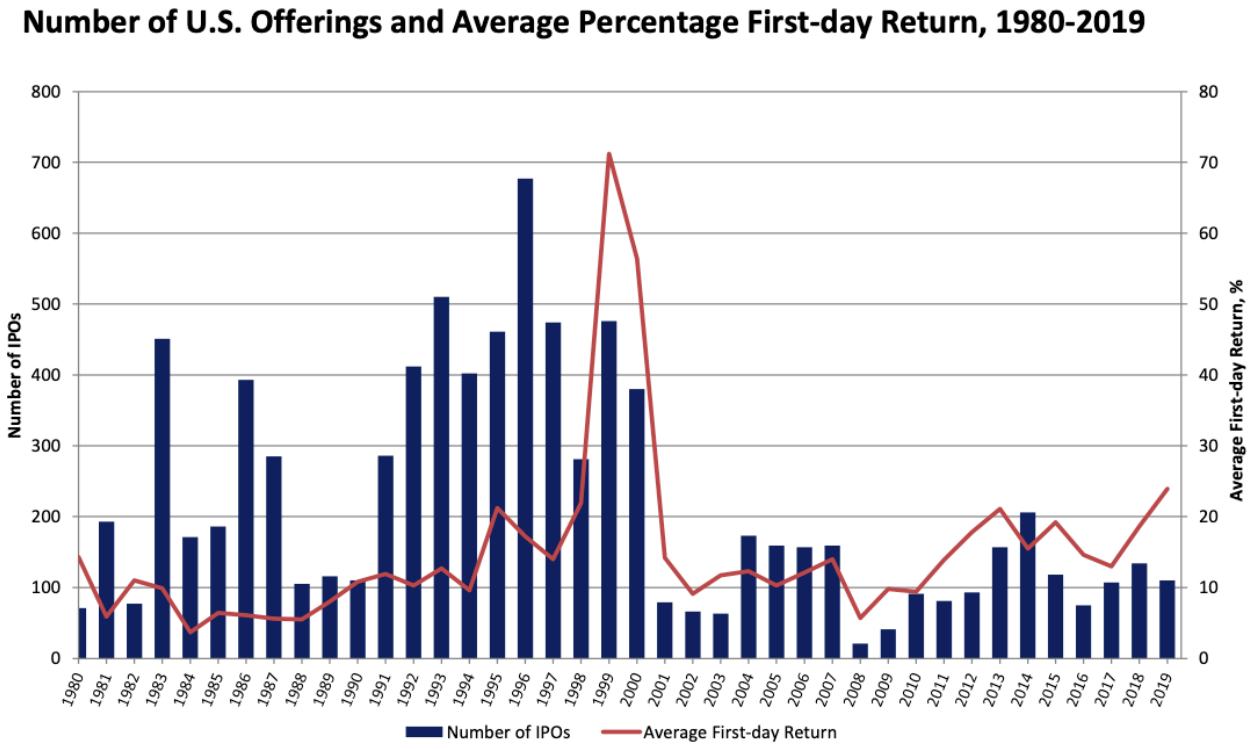
Investor sentiment and the prediction of the IPO Underpricing Level-Based on the Random Forest-LSTM Model

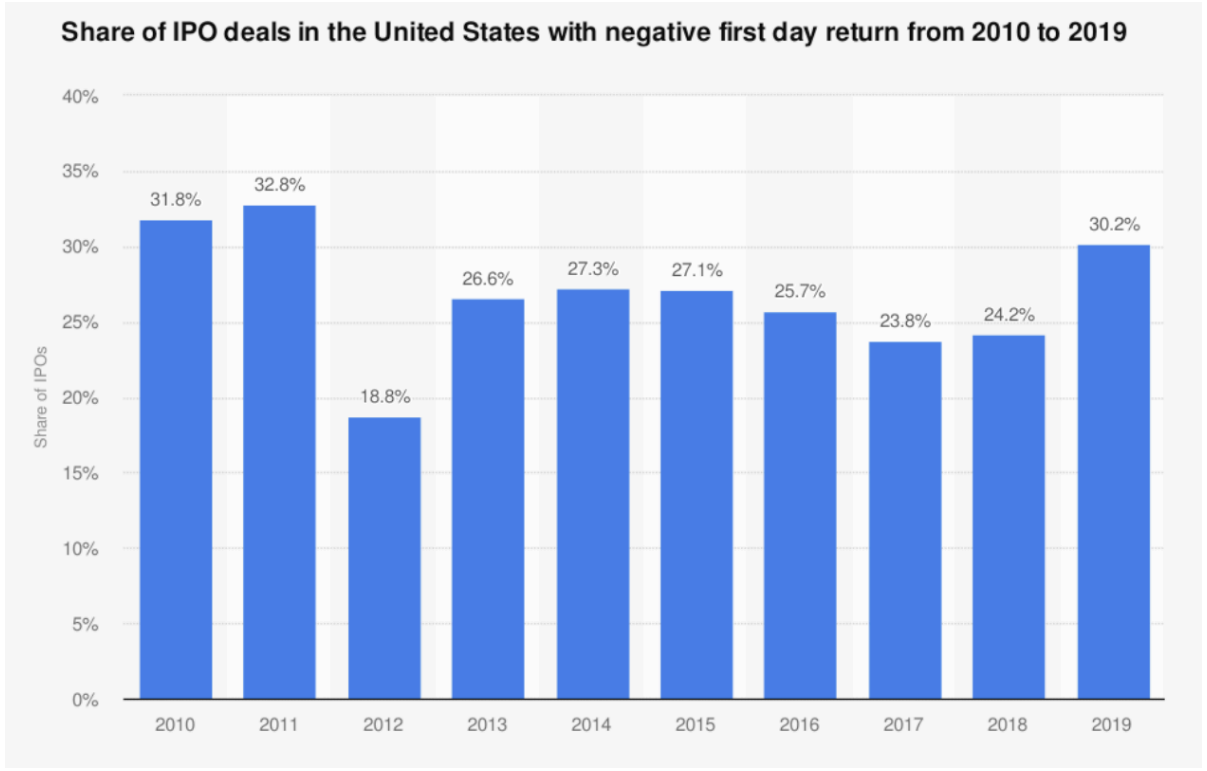
Yanwei Pan

Perspectives on Computational Research for Economics
Spring 2020

- Inspirations



- Jay R. Ritter



- Statista

"New issues Puzzle"

20 Century: Theoretical Debate

- Logue (1973) and Ibbotson (1975)
- Winner's curse (Rock, 1986)
- Signaling model (Welch, 1989)

21 Century: Empirical Analysis

- Tian (2011)
- Butler and Keefe (2014)

New Insight: Behaviral Economics

- Sanjiv and Thosar (2004)
- Ljungqvist, Nanda and Singh (2006)

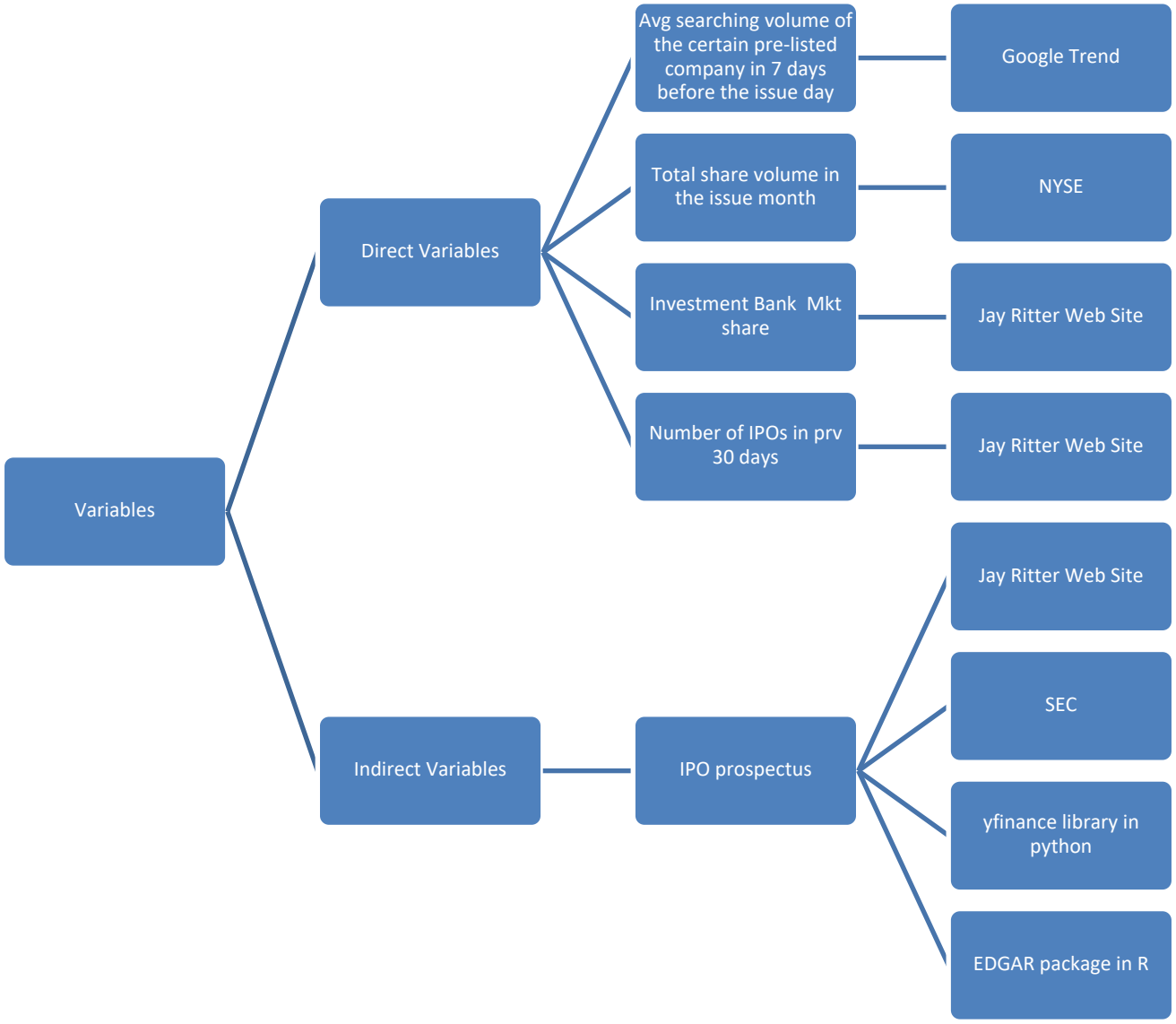
- # Why Interesting?

- Seldom of research focus on predicting the level of underpricing
- Investor Perspectives VS Issuer Perspectives
- Adding the indirect variables that represents the investor sentiment

- # My Contribution

- Using text analysis in behaviral finance area
- IPO pricing process
- Building better strategies
- IPO regulations

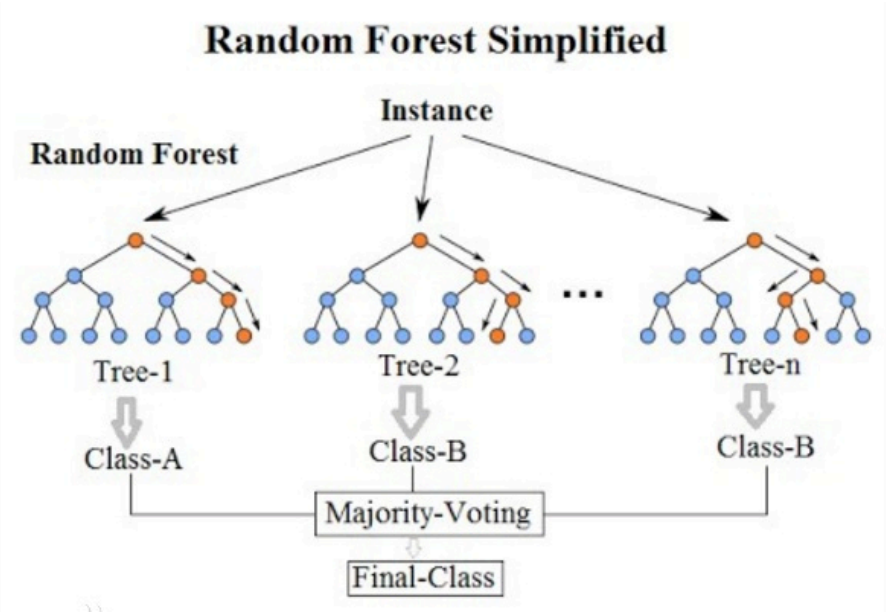
- Data


Variables
Direct Variables
Avg searching volume of the certain pre-listed company in 7 days before the issue day
Google Trend
Total share volume in the issue month
NYSE
Investment Bank Mkt share
Jay Ritter Web Site
Number of IPOs in prv 30 days
Jay Ritter Web Site
Indirect Variables
IPO prospectus
Jay Ritter Web Site
SEC
yfinance library in python
EDGAR package in R

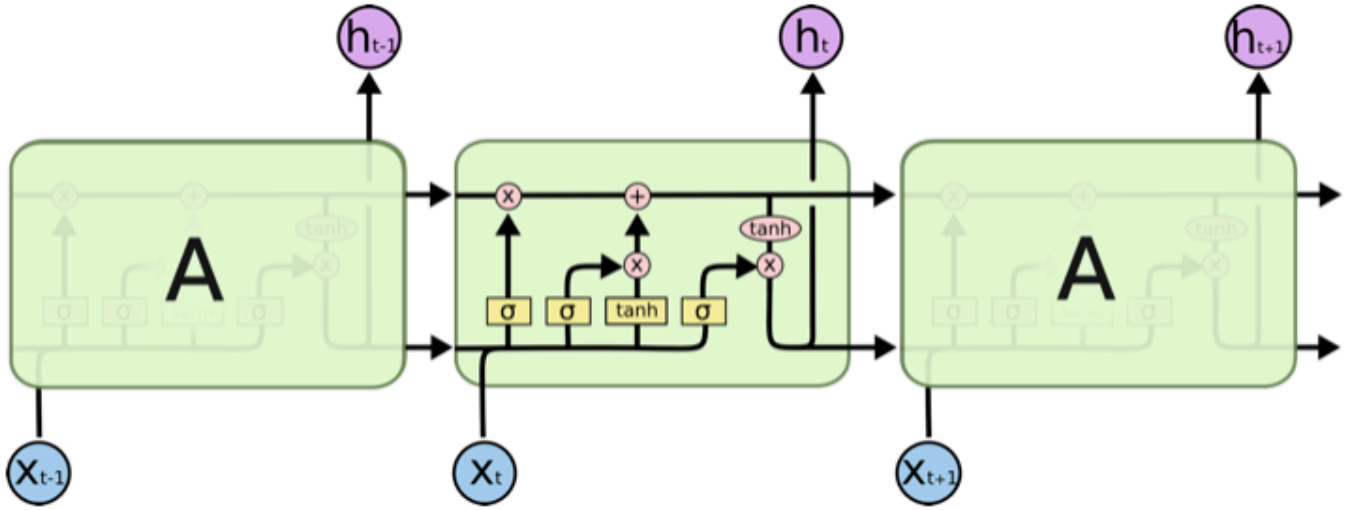
• Thoery & Model

- Investor Sentiment Theory
- Theories about “New Issues Puzzle”

- Random Forest Model:



- LSTM Model:



$$i_t = \sigma(W_i) \cdot [h_{t-1}, x_t] + b_i$$

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

$$C'_t = tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

$$C_t = f_t \times C_{t-1} + i_t \times C'_t$$

$$h_t = o_t \times \tanh(C_t)$$

- # Expected Result

- The accuracy of prediction (measured by RMSE) is much better than

  (1)using pure Random Forest model and direct variables

  (2)using pure LSTM model and indirect variables (text data from IPO prospectus)

- # Potential Channels & Alternatives

- Using Decision Tree instead of Random Forest if there is relatively few variables

Thank you!

Open to Any Suggestions :-P