Use Machine Learning to Forecast Future Earnings

Edward, XU Zhaoyu

Clair, CUI Xinyue

Ashlley, ZHOU Yue

The Hong Kong Polytechnic University

Contents

I. Project Overview

Project Overview

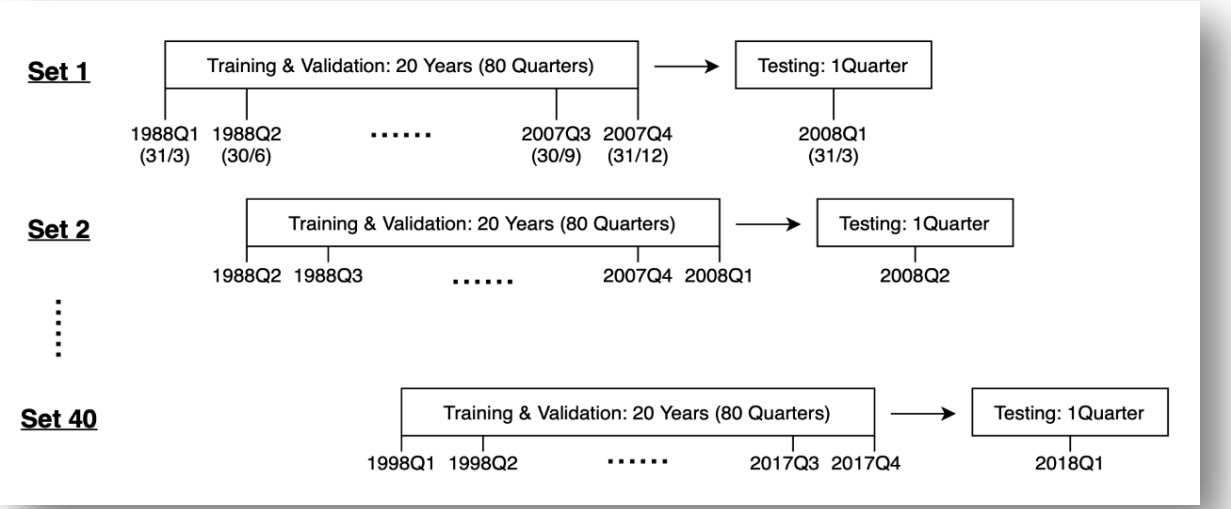
Overview

Objective

- We are trying to select, adjust and integrate a series of machine learning or deep learning models to comprehensively assess their feasibility and suitability on **predicting the company fundamentals** (i.e. earnings).

Highlights

- **Large Samples**: select from Top 3000 Market Capitalization Companies in US
- **Multi-Class Prediction**: 2 to 9 groups compared to peers
- **Rolling Window validation**: 40 overlapping sets
- **Relative Earnings Change**: Year-over-Year (YoY) or Quarter-over-Quarter (QoQ)

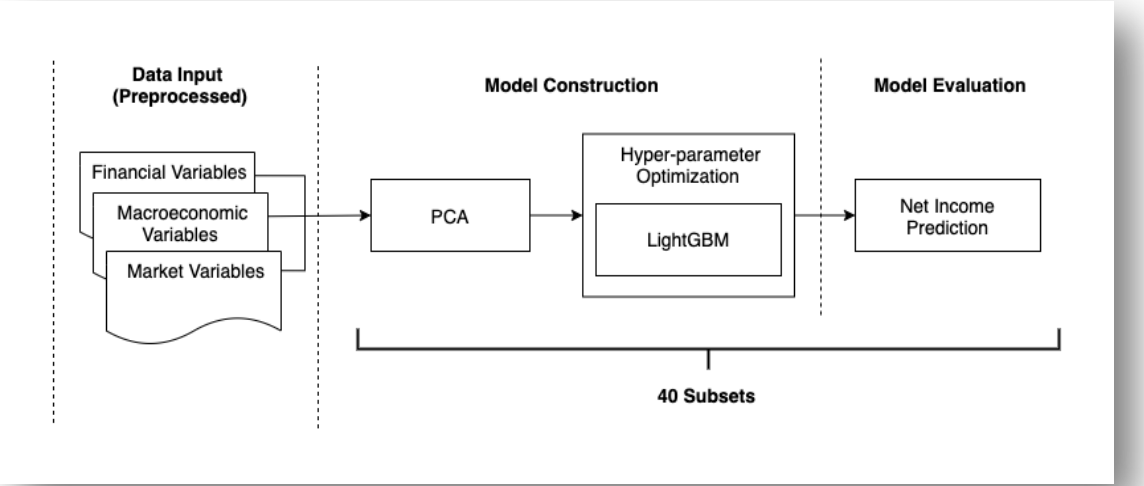


$$E_t = Classification\left(\frac{Net\ Income_{t+1} - Net\ Income_t}{Net\ Income_t}\right)$$

Project Overview

Techniques

- **Principal Component Analysis (PCA)**
  - Dimension reduction of input variables for easier training
- **LightGBM**
  - Gradient Boost Decision Tree (GBDT) Main Classifier
- **Hyperopt**
  - Tree of Parzen Estimators (TPE) Hyperparameter Optimization



Project Overview

Variables

Financial Variables

- Fundamental Data on Company Level
- **73** most relevant variables based on past literature

WRDS

Macroeconomic Variables

- **Gross Domestic Product (GDP)**
- **Inflation:**
  - Personal Consumption Expenditure Price Index (PCE)
- **S&P 500 Index**
- **Interest Rate:**
  - Central bank market rate
  - Short-term market rate (rM3)
  - Long-term interest rate (rY10)

Market Variables

- Abnormal returns of each company against the S&P500 benchmark

Time-series Variables

- 19 Quarterly Lagging variables over the past 5 years

Project Overview

Literature Review

- **Fundamental Data** is highly supportive.

  -----by Kothari (2001)

- Most frequently cited research papers of **earnings forecast**

  - Ordinary least squares (OSL) regression (Lewellen, 2004)
  - Time-series model (Myers, Myers, & Skinner, 2007)
  - Random walk model (Watts & Leftwich, 1977)
  - Statistical distribution of earnings

- Highest cited papers regarding **machine learning models** in Accounting & Finance

  - Support vector machines (SVMs) (Kim, 2003; Pai and Lin, 2005)
  - Neural networks (Campbell, 1987; Chang, Liu, Lin, Fan, & Ng, 2009; Chen, Leung, & Daouk, 2003)
  - Autoregressive conditional heteroskedasticity (ARCH/GARCH) (Engle, 1982; Bollerslev, 1986)
  - **Gradient Boosting Decision Tree (GBDT)** (Jones, Johnstone, & Wilson, 2015)

II. Model Construction

Model Construction

Dimension Reduction – Principal Components Analysis

Why bother to reduce the dimensionality of our features?

- In our project, initially there are **3091** different features (dimensions) after adding the artificial lagging. Thus, the **degree of freedom** would be so high that no way we could find sufficient number of samples to ascertain every aspect of them.

- Moreover, there are also a lot of machine learning algorithm that involves **the sequencing of "distance"** (norm). When #dimension goes up, the comparison of distance will be much more difficult.
- The concept of "far" and "near" would become ambiguous in high-dimensional spaces.

- The problems caused by **sparse data samples** and **difficult distance ordering** in high-dimensional situations are serious obstacles common to all machine learning methods, which are called the *curse of dimensionality* (Bellman, 1957).

Special Cases of Minkowski Distance

- $h = 1$: Manhattan (city block, L_1 norm) distance
  - E.g., the Hamming distance: the number of bits that are different between two binary vectors

$$d(i,j)=|x_{i_1}-x_{j_1}|+|x_{i_2}-x_{j_2}|+...+|x_{i_p}-x_{j_p}|$$

- $h = 2$: (L_2 norm) Euclidean distance

$$d(i,j)=\sqrt{(|x_{i_1}-x_{j_1}|^2+|x_{i_2}-x_{j_2}|^2+...+|x_{i_p}-x_{j_p}|^2)}$$

- $h \to \infty$. "supremum" (L_{max} norm, L_∞ norm and Chebyshev) distance.
  - This is the maximum difference between any component (attribute) of the vectors

$$d(i,j)=\lim_{h\to\infty}\left(\sum_{f=1}^{p}|x_{if}-x_{jf}|^h\right)^{\frac{1}{h}}=\max_{f}^{p}|x_{if}-x_{jf}|$$

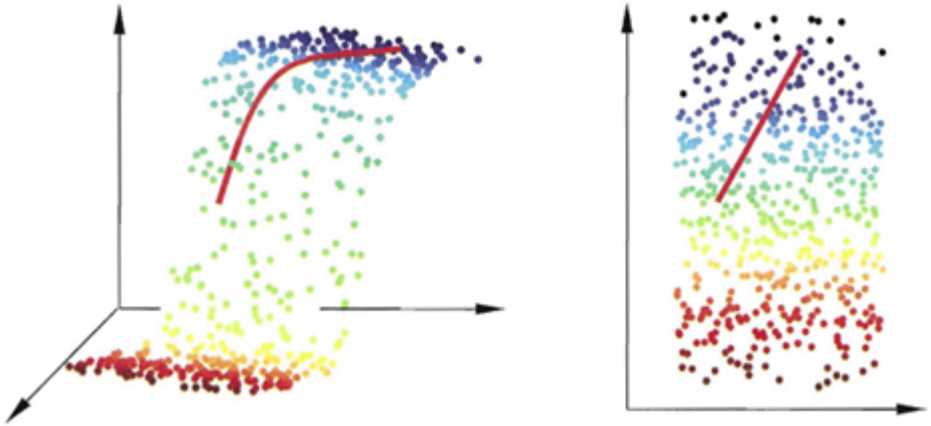
Model Construction

Dimension Reduction– Principal Components Analysis

Then, how do we deal with such issue?

Naturally, we want to reduce the number of dimensions.

Or more rigorously, that is to **find a low-dimensional subspace** (like a “simplified version”) of the original high-dimensional dataset, in which the sample density will be greatly increased, and the distance ranking will also become much easier.



Well, however, though such thought is quite simple and plain, but one has to realize that a “subspace” would certainly mean that, **it only contains “part” of the original information**, while some of the other information are correspondingly lost in the “reduced dimensions”.

Model Construction

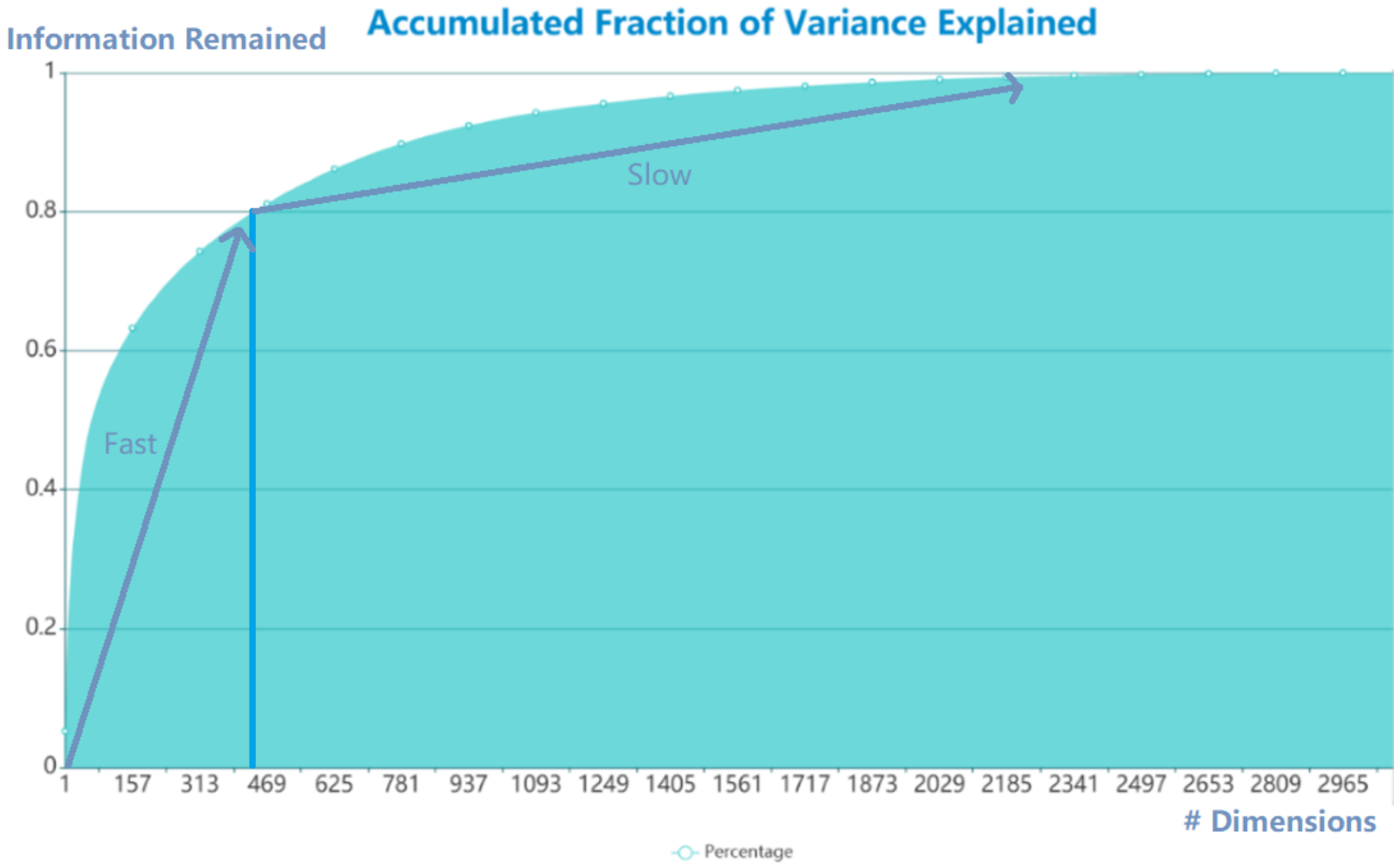
Dimension Reduction – Principal Components Analysis

Trade-off between

Number of Dimensions

and

Number of Information



Model Construction

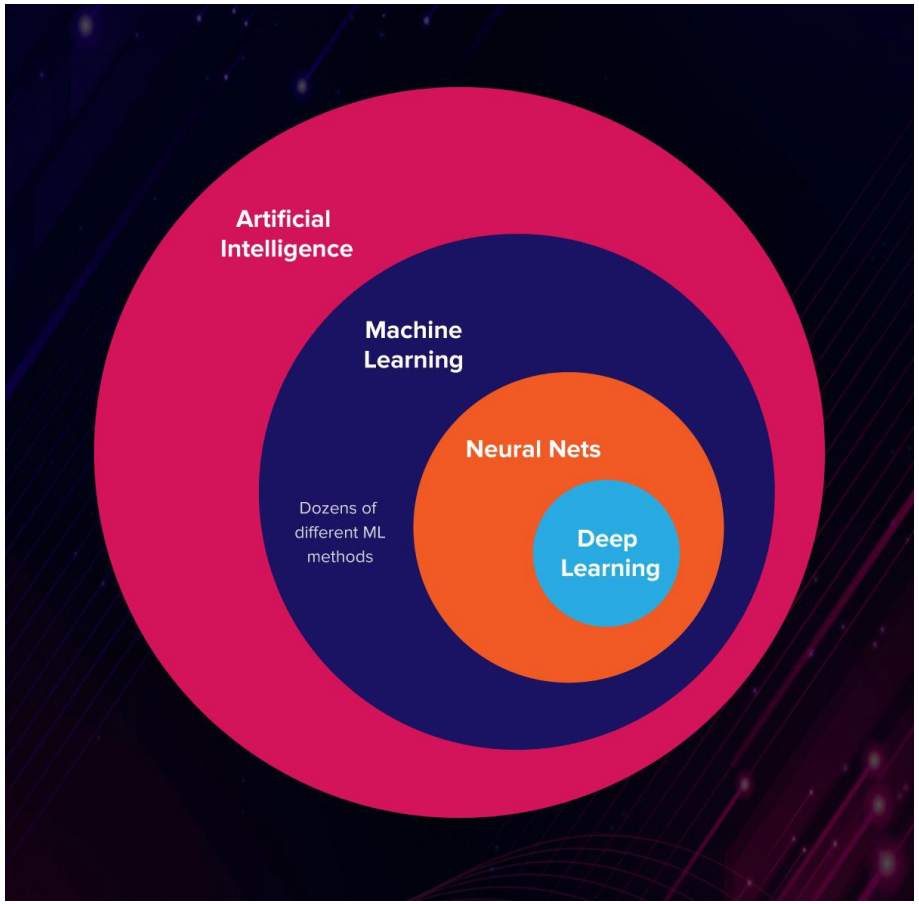
Gradient Boost Decision Tree - LightGBM

What is Machine Learning?

- **Artificial intelligence** is a science like mathematics or biology. It studies ways to build intelligent programs and machines that can creatively solve problems (which has always been considered a human prerogative).

- **Machine learning** is a subset of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed.

- **Deep learning** (or deep neural learning), is a subset of machine learning, which uses the neural networks to analyze different factors with a structure that is similar to the human neural system.



Model Construction

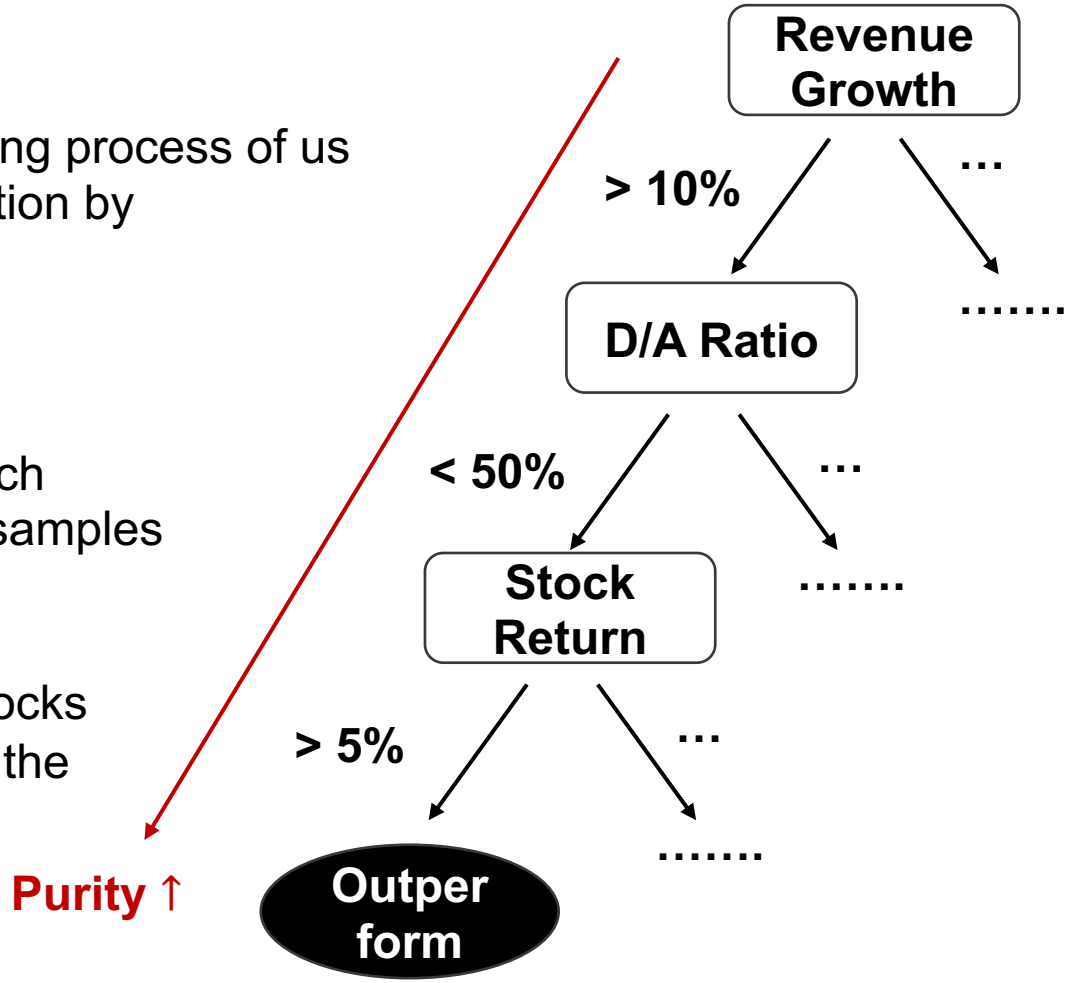
Gradient Boost Decision Tree - LightGBM

What is Decision Tree?

A decision tree is actually an imitation of the decision-making process of us human-beings – You make a decision to an **ultimate** question by progressively considering several '**sub-questions**'.

Intuitively, we shall expect that with the development of such division (i.e. with the tree growing deeper), the **'purity'** of samples contained in each node will also increase gradually.

Ultimately, in the **perfect** case, only the real outperform stocks can be classified into the final category – "Outperform(2)", the purity of this class will then be 100% (i.e. 100% correct).



Model Construction

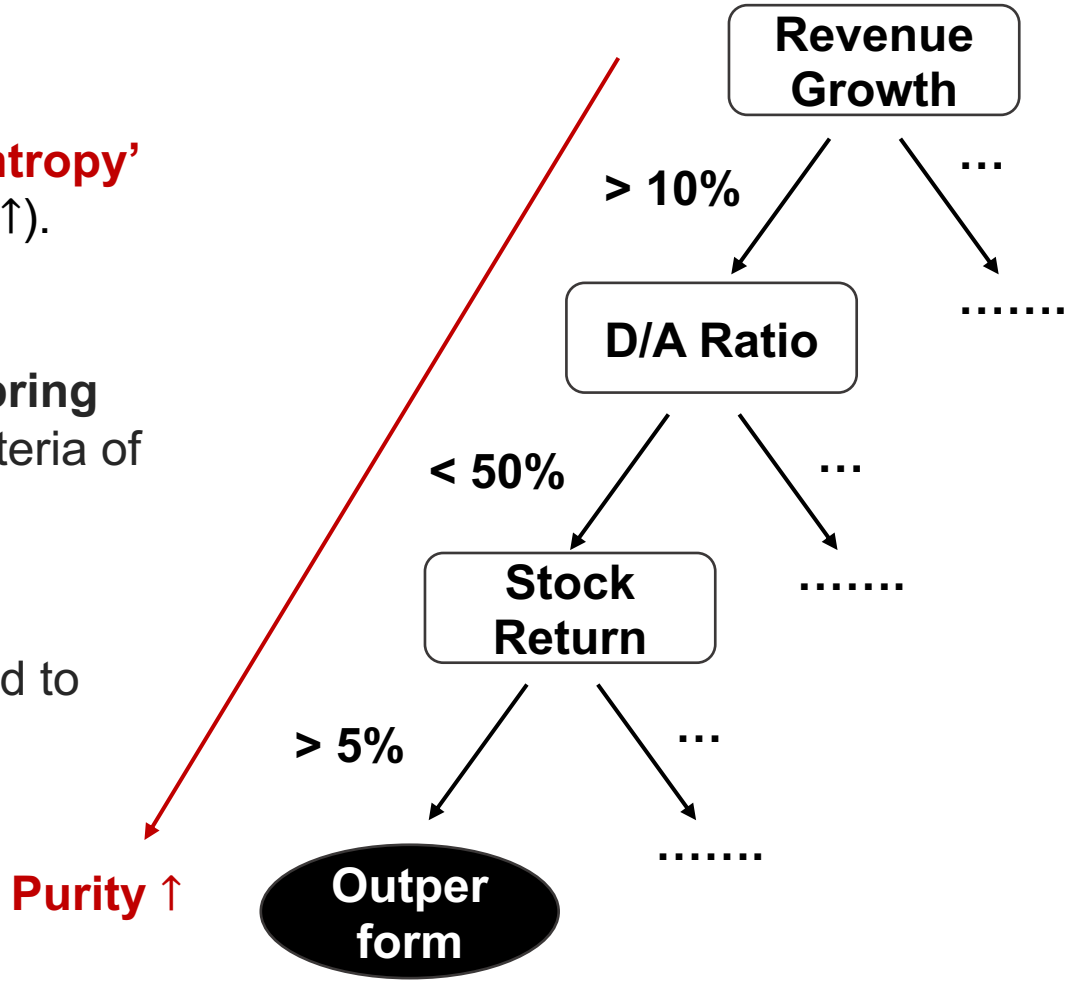
Gradient Boost Decision Tree - LightGBM

What is Decision Tree?

After each split, we shall assume that the **'Information Entropy'** for the subsets obtained would gradually decrease (purity ↑).

For each split, we will then choose the feature that could **bring the largest improvements** ("Information Gain") as the criteria of classification.

And similar procedures will go on and on, until you reached to the pre-set limits (If the **trees goes too deep**, i.e. the **classification is too specified**, the result would become useless -- **Overfitted**).



Model Construction

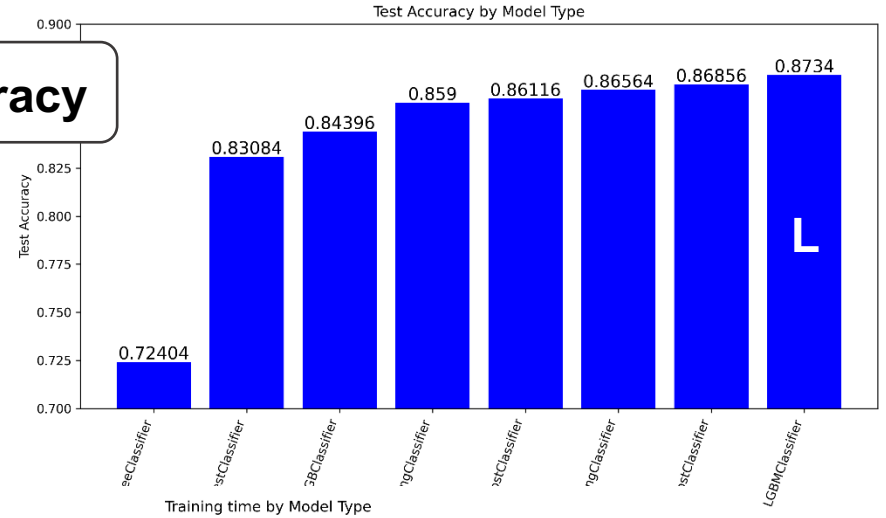
Gradient Boost Decision Tree - LightGBM

What is LightGBM?

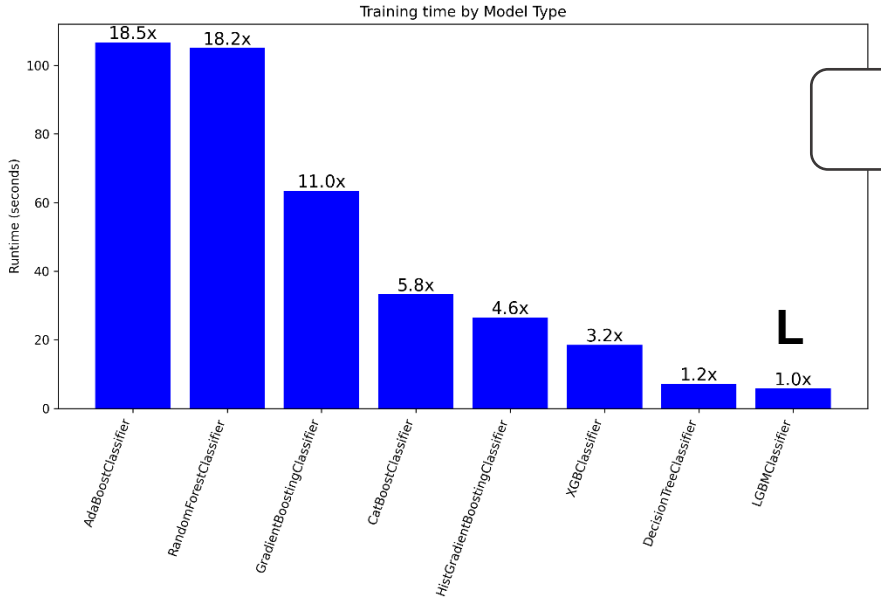
Introduced by **Microsoft** in 2017, LightGBM is a **“ridiculously” fast** toolkit specifically designed for modeling **extremely large data sets of high dimensionality**, often being more accurate and many times faster than XGBoost or several other previous tree-based algorithms.

In virtue of its satisfactory and consistent performance on multiple tasks, more and more competitors on Kaggle (one of the largest data science contest platform) started to integrate such model into their tasks.

Accuracy



Time

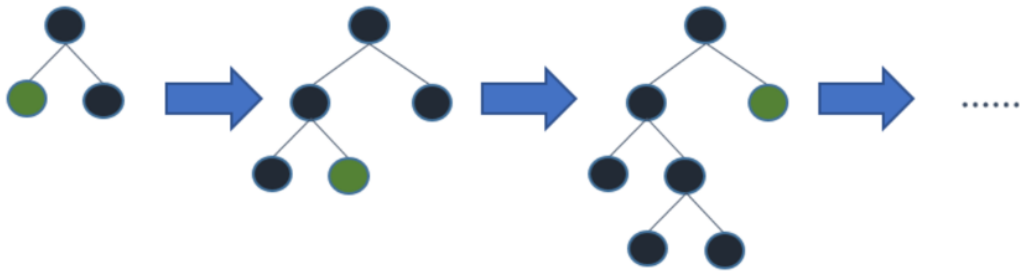


Model Construction

Gradient Boost Decision Tree - LightGBM

What is LightGBM?

LightGBM grows tree **vertically** (Leaf-wise) while other algorithms grow trees **horizontally** (Level-wise).



Leaf-wise tree growth

Level-wise tree growth

LightGBM will only choose the leaf with max **delta loss** (another measurement of 'purity') to grow.

When growing the same leaf, **Leaf-wise** algorithm can **reduce more loss** (increase more **purity**) than a level-wise algorithm.

LightGBM is prefixed as 'Light' because of its high speed in virtue of this ground-breaking algorithm. Therefore, LightGBM can **handle the large size of data and takes lower memory to run**.

III. Data and Experiment

Data and Experiment

Sample Selection

How to Select Samples?

Universe

- Quarterly Data in US Equity Market
- 3000 companies with the highest market capitalization
- 30-year period from 1988 to 2017

Exclusion

- Remove ETF (i.e. without Total Asset)
- Remove samples with abnormal trading histories (i.e. share price below $1)
- Remove Utility (GICS:55) and Finance (GICS:40) companies
- Remove Fiscal year-end not on Mar, Jun, Sep, or Dec

Final

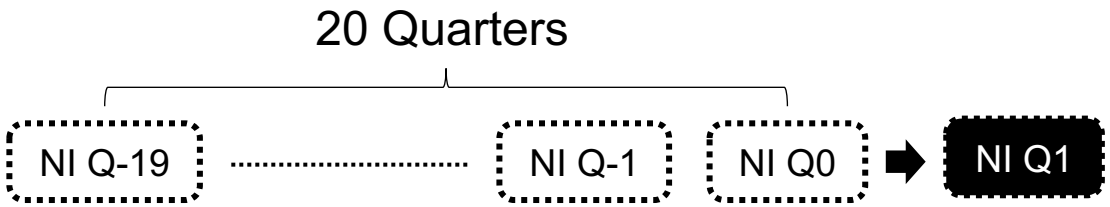
- 333325 samples from 4592 companies

Data and Experiment

Preprocessing

How to Preprocess Samples?

- Add **time-series** with lagging data point



- **Standardize** samples for time-series / cross-section analysis
  - *Year-over-year growth rate* (YoY): Balance Sheet Items
  - *Quarter-over-quarter growth rate* (QoQ): Income Statement / Cash Flow Statement Items
  - *Percentage of total asset* (%Asset): Balance Sheet Items
  - *Percentage of total revenue* (%Revenue): Income Statement / Cash Flow Statement Items
  - *Nominal value*: Total Asset and Total Revenue as Company Size Indicator

Data and Experiment

Preprocessing

How to Preprocess Samples?

Outliers

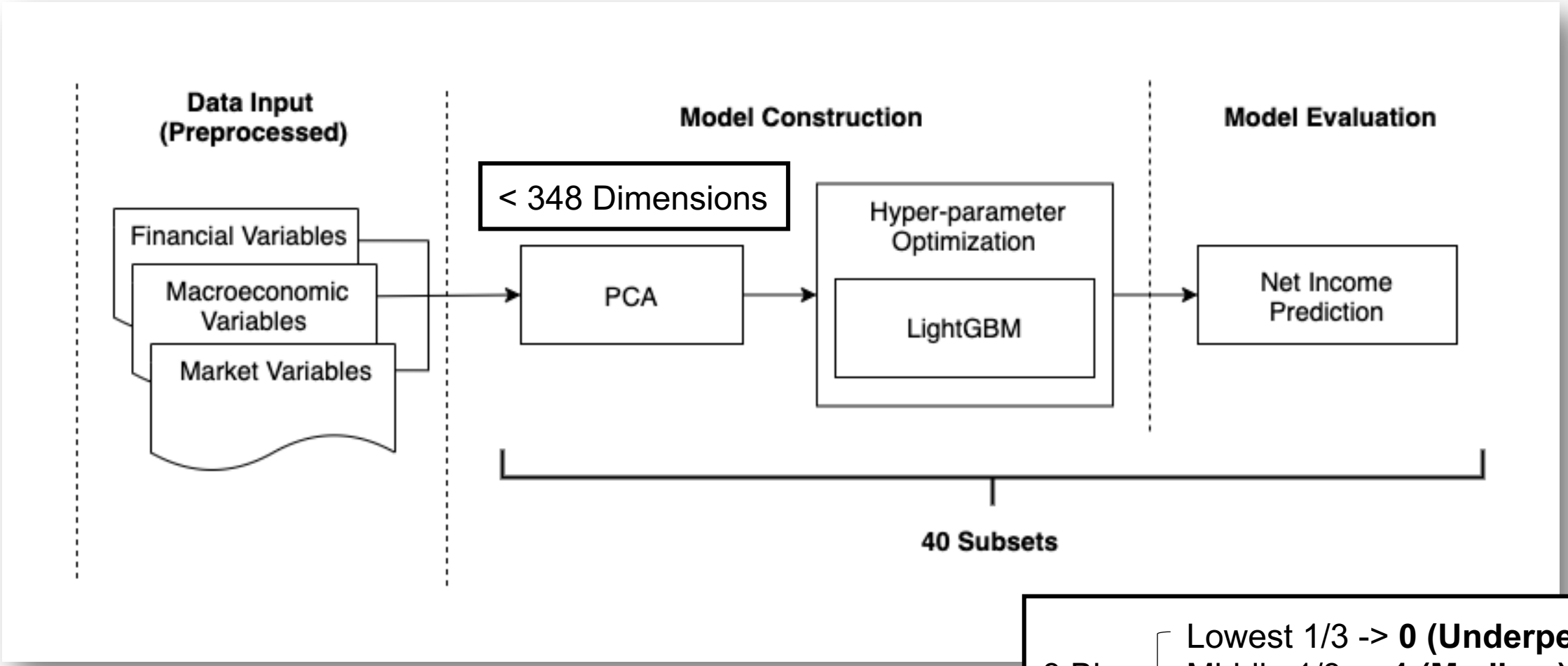
- (Before conversion) Replace all negative value to zero to avoid misleading conversion;
- Use 95% Percentile as maximum value to avoid infinity values

Missing Values

- *Sample deletion*: delete samples with missing on Total Assets, Total Liabilities, Total Equity, Revenue, Net Incomes, and Cash & Cash Equivalent;
- *Variable deletion*: remove variables with over 70% missing rate;
- *Relevant* fill-in: fill in with the rolling average over the optimal fill-in period
- *Constant fill-in*: fill-in remaining missing with constant value -1 to imply the missing

Data and Experiment

Process Illustration



3 Bins
- Lowest 1/3 -> **0 (Underperform)**
- Middle 1/3 -> **1 (Medium)**
- Top 1/3 -> **2 (Outperform)**

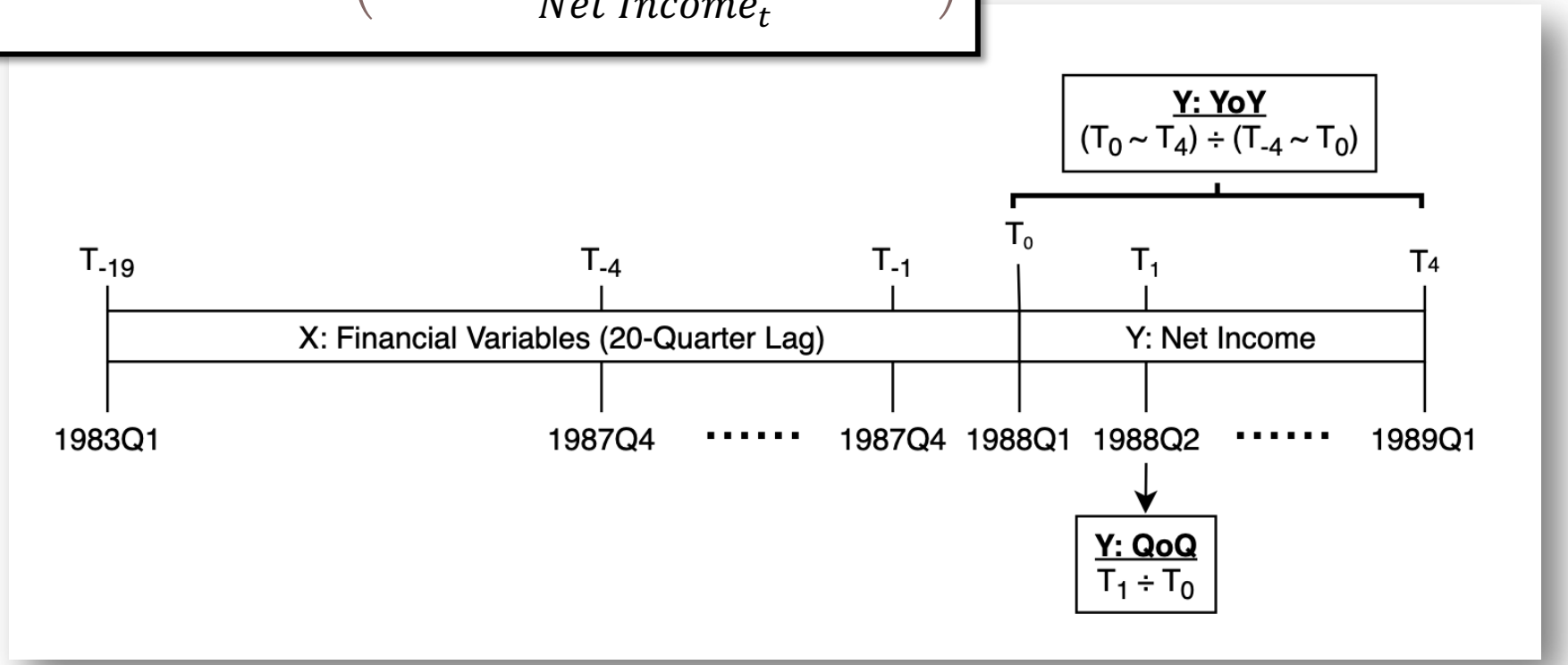
Data and Experiment

Dependent Variables

$$E_t = \ Classification\left|\left(\frac{Net\ Income_{t+1} - Net\ Income_t}{Net\ Income_t}\right)\right.$$

YoY Prediction:
Growth of Annual Net Income

QoQ Prediction:
Growth of Quarterly Net Income



IV. Results and Analysis

Results and Analysis

Benchmark Comparison – Logistic Regression

LightGBM vs Logistic Regression

Comparison of Prediction Performance of Sign of Earnings Changes

Methods	Our Paper	Ou and Penman (1989)*	Hunt, Myers & Myers (2019)**
LightGBM	64.2%		
Stepwise Logistic Regression		63.1%	62.1%
Elastic Net Logistic Regression			62.0%

* This paper predicts probablity of earnings increase in the subsequent year. Here shows the average accuracy.

** Here presents the maximum accuracy in both case.

- Our model outperform the accuracy of Both Logistic Regression Model proposed by Ou and Penman (1989) and Hunt Myers & Myers (2019).
- All models here deal with **Sign of Earnings Change** problem (increase / decrease).

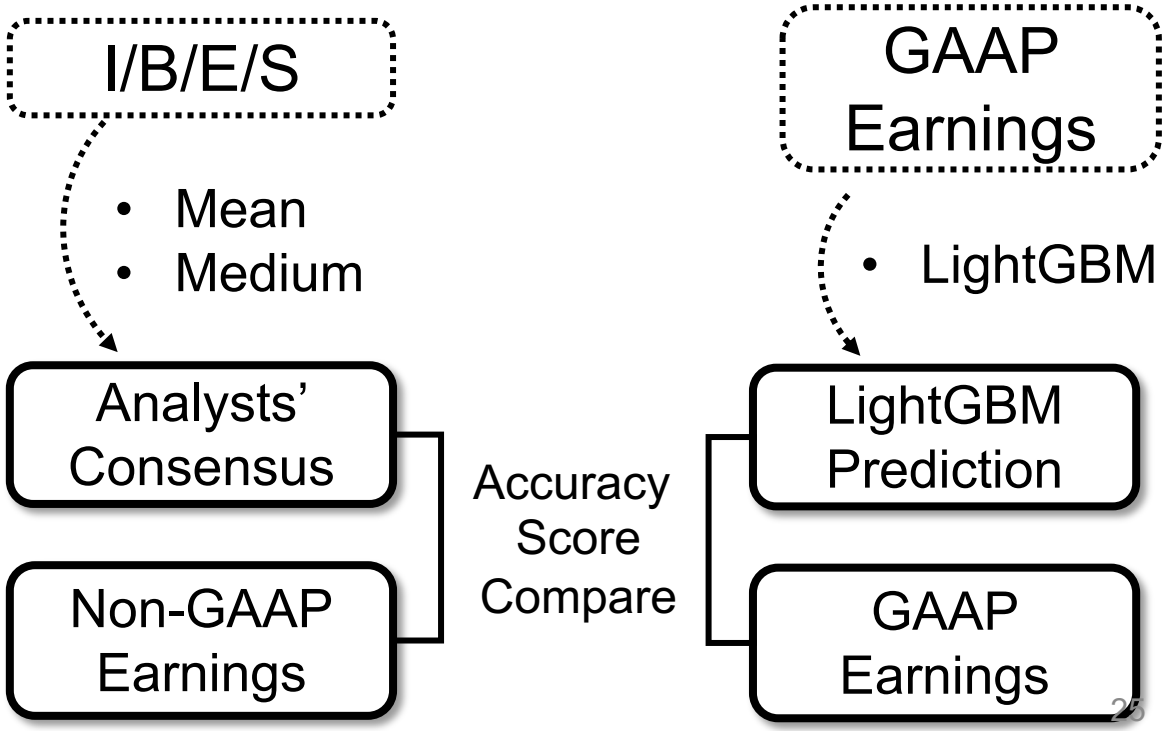
Results and Analysis

Benchmark Comparison – Consensus

LightGBM vs I/B/E/S Analysts' Consensus Prediction

- Because I/B/E/S predict **Non-GAAP Earnings**, we compared the classified I/B/E/S prediction with the classified Street Earnings.
- For equivalent comparison, we cut I/B/E/S consensus with the **same threshold** for GAAP earnings conversion as the dependent variable of our model.

$$E_t = \ Classification\left|\left(\frac{Net\ Income_{t+1} - Net\ Income_t}{Net\ Income_t}\right)\right.$$



Results and Analysis

Benchmark Comparison – Consensus

LightGBM vs I/B/E/S Analysts' Consensus Prediction

Comparison of Multi-Class Prediction Performance on Average

Types of Dependent Variables	Number of Class	LightGBM	Conesnsus (Mean)	Conesnsus (Medium)
QoQ	3	*52.7%*	*74.3%*	*74.2%*
	6	32.3%	56.3%	56.3%
	9	23.4%	46.2%	46.3%
YoY	3	52.1%	71.6%	72.0%
	6	32.4%	50.8%	51.4%
	9	23.5%	40.4%	40.7%

- Our model underperform I/B/E/S Analysts' Consensus Prediction.
- LightGBM performs relatively better for Year-over-Year (YoY) prediction and when the number of class is less (i.e. 3-Class) .

Results and Analysis

Benchmark Comparison – Consensus

LightGBM vs I/B/E/S Analysts' Consensus Prediction

Comparison of Converge Accuracy and Total Accuracy

Types of Dependent Variables	Number of Class	Converge		Total	
		LightGBM	Conesnsus (Mean)	LightGBM	Conesnsus (Mean)
QoQ	3	68.4%	*81.3%*	52.7%	*74.3%*
	6	52.0%	67.3%	32.3%	56.5%
	9	43.3%	58.4%	23.4%	46.4%
YoY	3	63.0%	78.4%	52.1%	71.6%
	6	43.7%	61.2%	32.4%	50.8%
	9	33.5%	52.1%	23.5%	40.4%

- Our model improves the accuracy of I/B/E/S Analysts' Consensus Prediction by evaluating the converging case of LightGBM models and Consensus Prediction (i.e. same class).
- 3-class Quarter-over-Quarter problem achieved highest accuracy to 81.3%. However, comparatively 9-class YoY problem improves the most by 29%.

Results and Analysis

Benchmark Comparison – Consensus

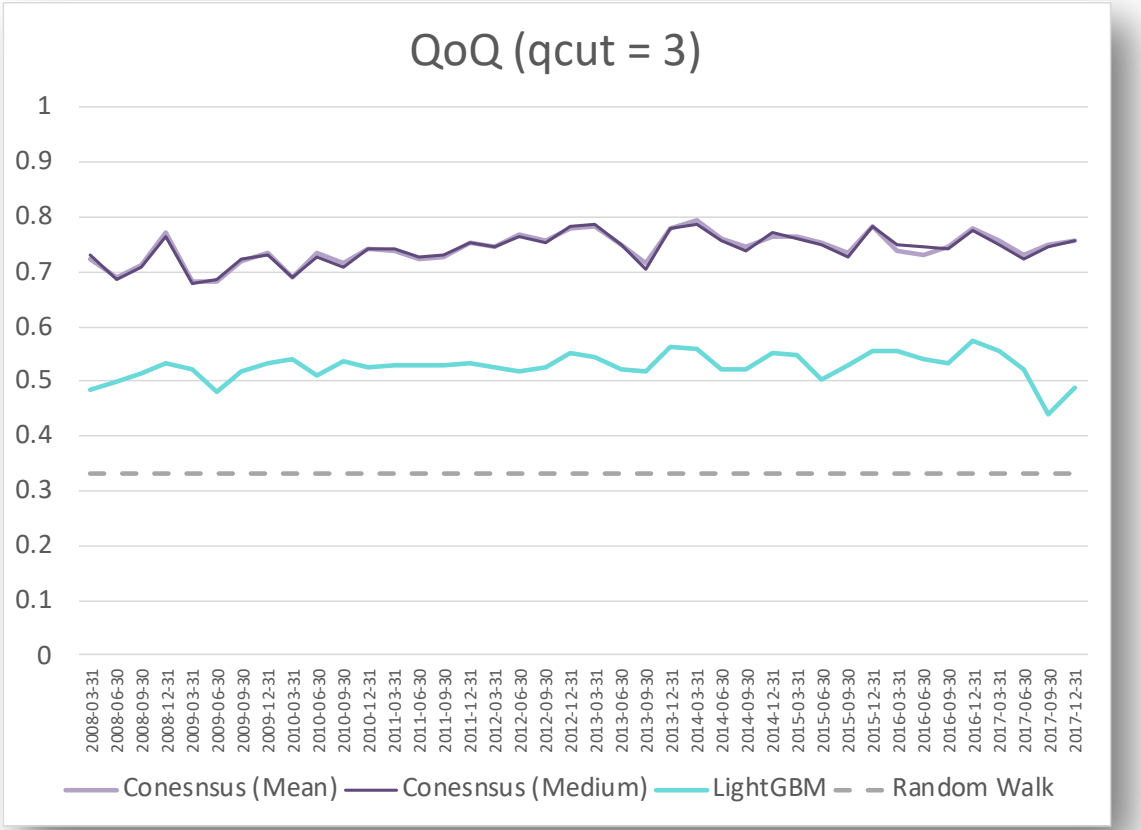
Why LightGBM fails Consensus?

Analyst's Superiority[1]

- Non-accounting information
  - News
  - Language & sentiment data
- Information released after the prior fiscal year/quarter

Correlation

- Our results has 0.44 correlation with the consensus prediction.



[1] Fried & Givoly 1982, Das, Levine, & Sivaramakrishnan 1998

V. Conclusion

Conclusion

Summary and Suggestions

Summary

- LightGBM is an innovative machine learning techniques that has great potential in accounting and finance research.
- Our paper proves its ability to predict future earnings and generate relative accurate results compared to many other statistical models.

Suggestions

- Unfortunately, due to the constrains of time and data access, our paper failed to outperform consensus.
- Future research may implement Natural Language Processing (NPL) to include non-financial data into the analysis and improves the results.

Thanks for Listening!

Edward, XU Zhaoyu
zhaoyu.xu@connect.polyu.hk

Clair, CUI Xinyue
xinyue.cui@connect.polyu.hk

Ashlley, ZHOU Yue
yue.zhou@connect.polyu.hk