

kaggle

# Using News to Predict Stock Movements

## 结题报告

**Team 5**

徐哲 张凌霄 许卓群 路万通 于苗苗

# 目录

CONTENT

课题背景及意义

PART ONE



探索性数据分析(EDA)

PART TWO



模型

PART THREE



团队分工及计划

PART FOUR



# 课题背景及意义

# 课题背景及意义



TWO SIGMA

## Using News to Predict Stock Movements

### 利用新闻预测股票趋势

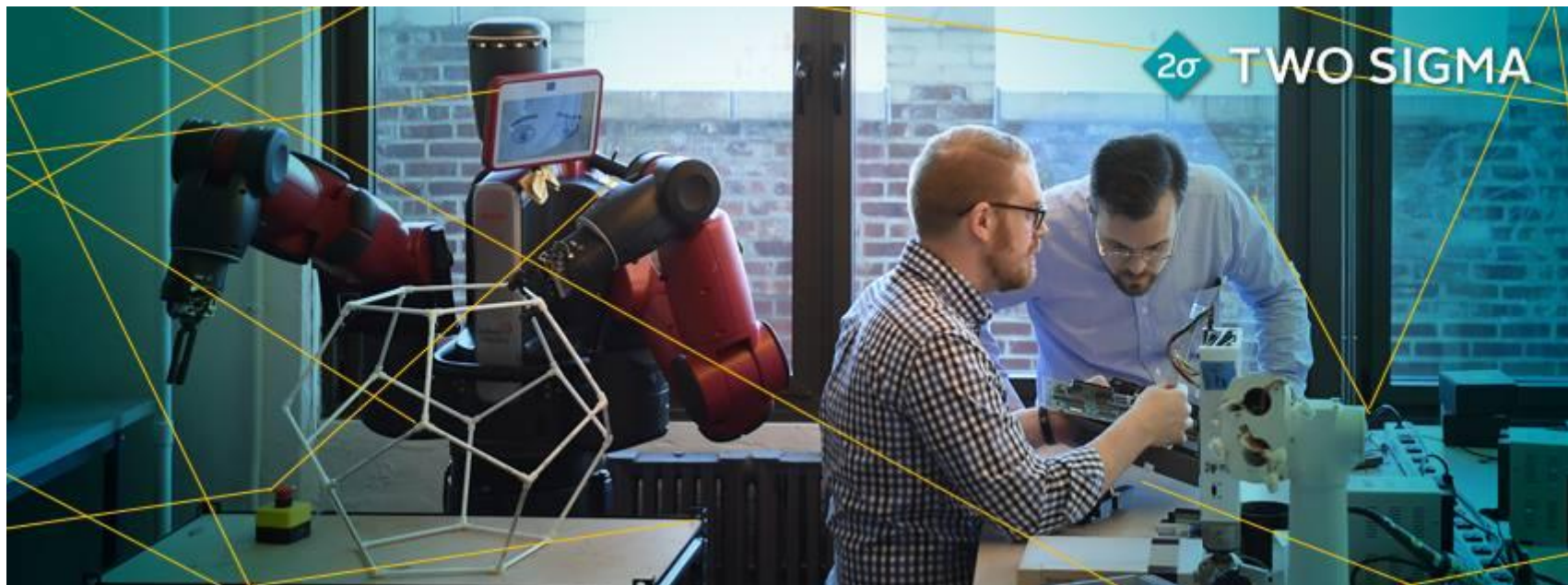
Can we use **the content of news** analytics to **predict stock price performance**? The ubiquity of data today enables investors at any scale to make better investment decisions. The challenge is ingesting and interpreting the data to determine which data is useful, finding the signal in this sea of information. Two Sigma is passionate about this challenge and is excited to share it with the Kaggle community.

As a scientifically driven investment manager, Two Sigma has been applying technology and data science to financial forecasts for over 17 years. Their pioneering advances in big data, AI, and machine learning have pushed the investment industry forward. Now, they're eager to engage with Kagglers in this continuing pursuit of innovation.

By analyzing news data to predict stock prices, Kagglers have a unique opportunity to advance the state of research in understanding the predictive power of the news. This power, if harnessed, could help predict financial outcomes and generate significant economic impact all over the world.

---

# 课题背景及意义



 TWO SIGMA

一家位于纽约的对冲基金公司，使用各种技术方法，包括人工智能，机器学习和分布式计算，用于其交易策略

# 课题背景及意义

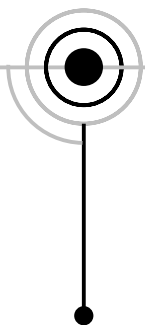
Wuthrich

收集财经新闻作为语料库，用机器学习的算法决策树和KNN对股票进行预测

Boolen, Mao, Zeng等

抓取Twitter上与道琼斯工业指数相关的重点词并构建公众情绪指数来表征公众情绪，使用公众情绪指数来预测道琼斯工业指数。

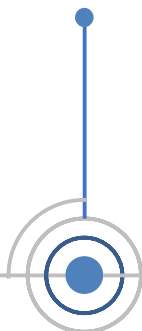
1971



采用简单统计方法研究《纽约时报》发布重大新闻后股票市场的表现

Niederhoffer

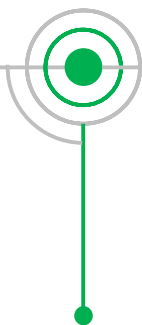
1998



用SVM来预测股票价格指数变动，特征取自财经新闻语料库，正确率达到58.9%

Mahajan

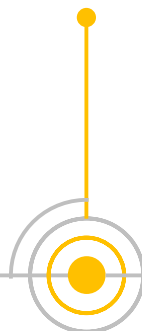
2008



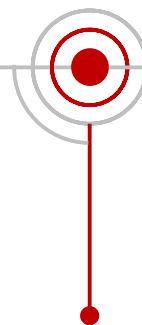
研究发现在预测标普500指数时，使用财经新闻标题比新闻内容更有效

Manuel R.Vargas

2011



2017



# 探索性数据分析(EDA)

# 探索性数据分析(EDA)

本次比赛的数据来源主要来自于两个方面:

- 1.由Intrinio提供的2007年至今市场数据,其中包含金融市场信息,如开盘价,收盘价,交易量,计算回报等。
2. 2007年至今新闻数据(资料来源:汤森路透),其中包含有关资产的新闻文章/警报信息,如文章详情,情绪和其他评论。

```
market_train_df.describe()
```

	volume	close	open	returnsClosePrevRaw1	returnsOpenPrevRaw1
count	4.072956e+06	4.072956e+06	4.072956e+06	4.072956e+06	4.072956e+06
mean	2.665312e+06	3.971241e+01	3.971233e+01	5.473026e-04	9.569113e-03
std	7.687606e+06	4.228822e+01	4.261116e+01	3.697774e-02	7.084388e+00
min	0.000000e+00	7.000000e-02	1.000000e-02	-9.776464e-01	-9.998881e-01
25%	4.657968e+05	1.725000e+01	1.725000e+01	-1.089241e-02	-1.108987e-02
50%	9.821000e+05	3.030000e+01	3.029000e+01	3.373819e-04	3.824092e-04
75%	2.403165e+06	4.986000e+01	4.985000e+01	1.165695e-02	1.183612e-02
max	1.226791e+09	1.578130e+03	9.998990e+03	4.559245e+01	9.209000e+03

市场数据的分布

```
news_train_df.describe()
```

	urgency	takeSequence	bodySize	companyCount	sentenceCount	wordCount
count	9.328750e+06	9.328750e+06	9.328750e+06	9.328750e+06	9.328750e+06	9.328750e+06
mean	2.321202e+00	2.122825e+00	3.768918e+03	5.027720e+00	2.250942e+01	5.800000e+01
std	9.470095e-01	2.944505e+00	7.475653e+03	8.787980e+00	3.601975e+01	9.500000e+01
min	1.000000e+00	1.000000e+00	0.000000e+00	1.000000e+00	1.000000e+00	1.000000e+00
25%	1.000000e+00	1.000000e+00	0.000000e+00	1.000000e+00	1.000000e+00	2.000000e+00
50%	3.000000e+00	1.000000e+00	1.571000e+03	1.000000e+00	1.000000e+01	2.500000e+01
75%	3.000000e+00	2.000000e+00	4.504000e+03	5.000000e+00	3.000000e+01	7.600000e+01
max	3.000000e+00	9.700000e+01	1.227700e+05	4.300000e+01	1.205000e+03	2.000000e+02

新闻数据的分布



# 探索性数据分析(EDA)

## 市场数据:

- 回报总是计算为未平仓（从一个交易日的开盘时间到另一个交易日的开盘时间）或接近收盘价（从一个交易日的收盘时间到另一个交易日的开盘时间）。
  - 回报是原始的，意味着数据不是根据任何基准进行调整，也不是市场残差。
  - 所提供的数据包括，当前时间，资产的唯一ID，与资产ID对应的名称，一个表示当天的工具是否包含在评分中的布尔值，成交量，当天的开盘价和收盘价。
-

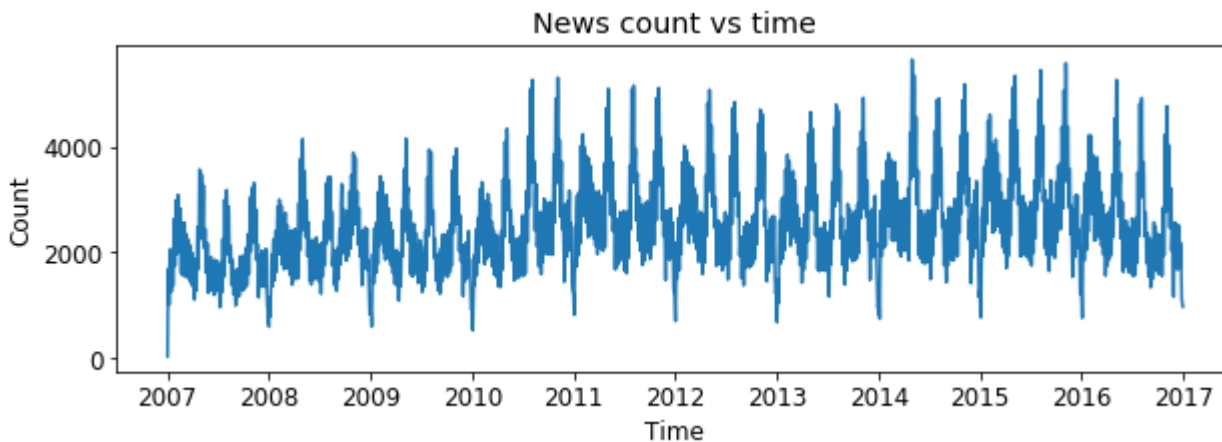
# 探索性数据分析(EDA)

新闻数据:

新闻数据给出的数据有, **Feed上数据可用时间的UTC时间戳**, **新闻项创建时的UTC时间戳**, **项目第一个版本的UTC时间戳**, **每个新闻项的ID**, **项目的题目**, **新闻的紧迫性** (1代表警报, 3代表文章), **新闻项的获取序列号**, **新闻提供机构名称**, **新闻项目相关的主题代码和公司标识符**, **新闻的受众**, **新闻的字符数**, **新闻中列出的公司数**, **汤森路透的新闻标题**, **市场条件的布尔指标**, **新闻中句子总数**, **新闻中词汇标记总数**, **新闻中所提到的资产列表**, **资产的名称**, **从标题开始的第一个句子**, **其中提到了评分资产**, **新闻项与资产的相关性**, **新闻项相对于资产的主要情绪**, **新闻项目的情绪对资产为负的概率**, **新闻项目的情绪对资产是中性的概率**, **新闻项目的情绪对资产有利的概率**, **项目文本中与资产相关的部分中的词法标记数**, **特定资产的新闻项目内容的12小时新颖性**, **每个资产的12小时新闻量**。

---

# 探索性数据分析(EDA)

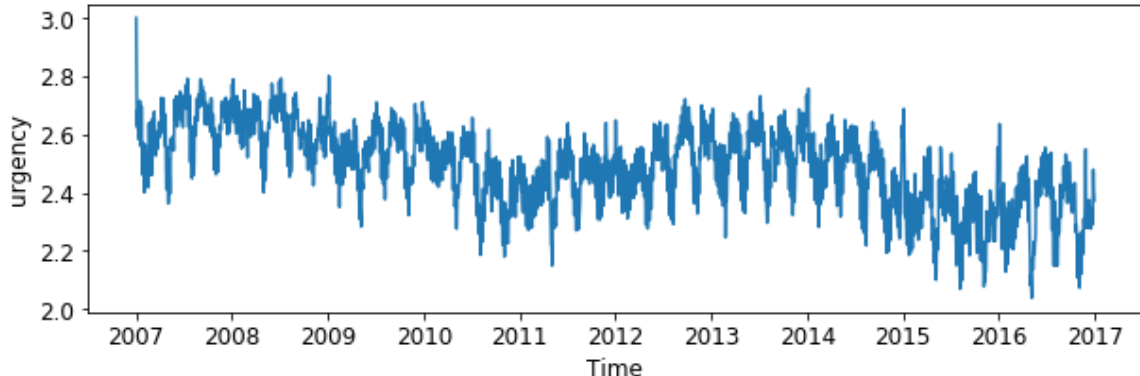


新闻数据随时间变化图

由以上图表可知，新闻数据量在每个季度呈周期性变化趋势，在每年年末（即圣诞节期间），新闻量达到最小值，且新闻数据量逐年上涨。

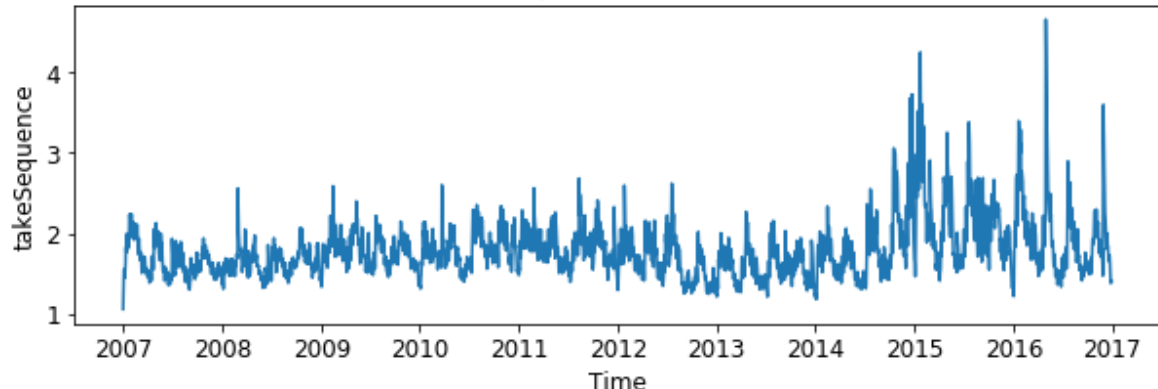
# 探索性数据分析(EDA)

urgency versus time



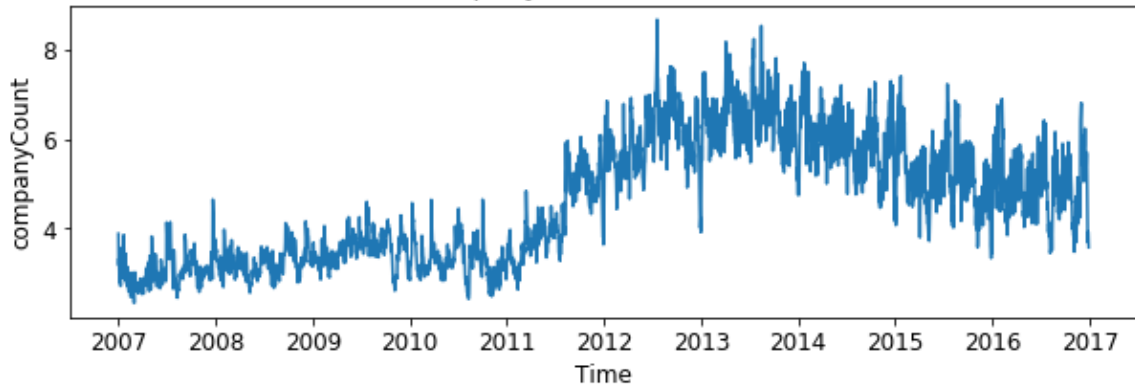
新闻紧迫性随时间变化图表

takeSequence versus time



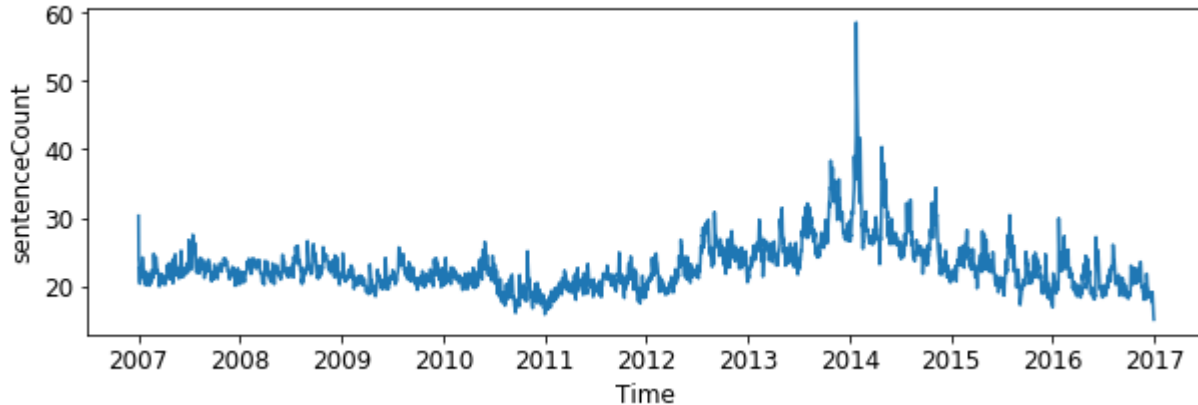
新闻项的获取序列号随时间变化图表

companyCount versus time



subject字段的新闻项中明确列出的公司数随时间变化图表

sentenceCount versus time



新闻条目中的句子总数随时间变化关系

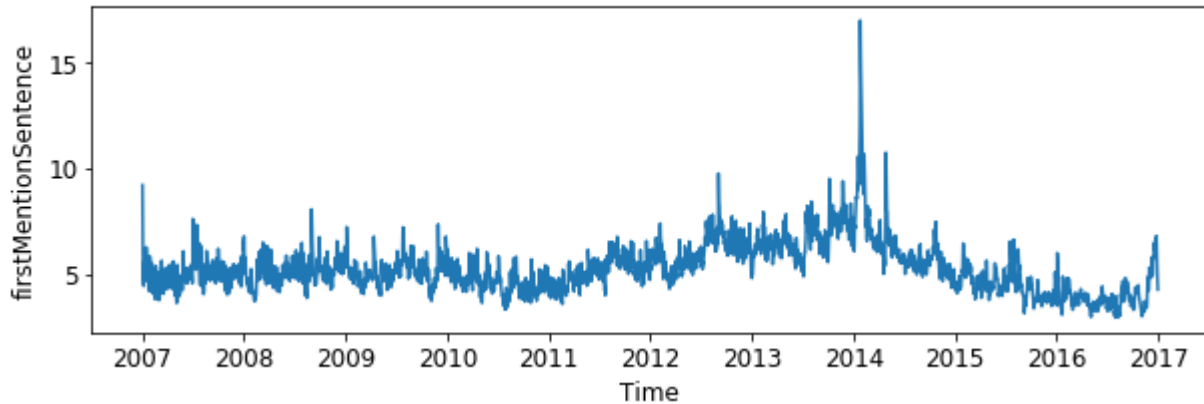
# 探索性数据分析(EDA)

由以上图表可知，**新闻的紧迫性**随时间变化**降低**，意味着新闻趋近于文章的趋势降低，趋近于警报的趋势升高。新闻中列出的**公司数目**和新闻中出现的**句子总数**总体呈上升趋势，这两者均在2014年达到峰值。

---

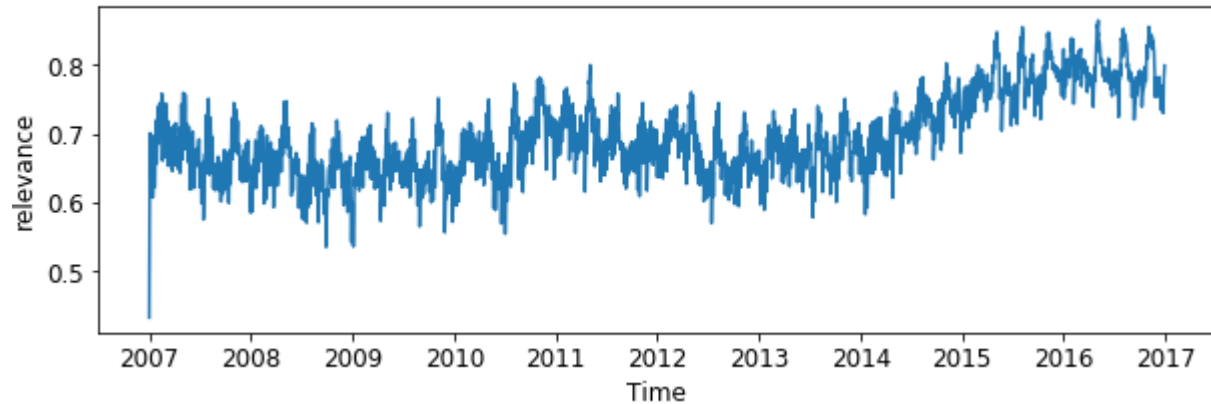
# 探索性数据分析(EDA)

firstMentionSentence versus time



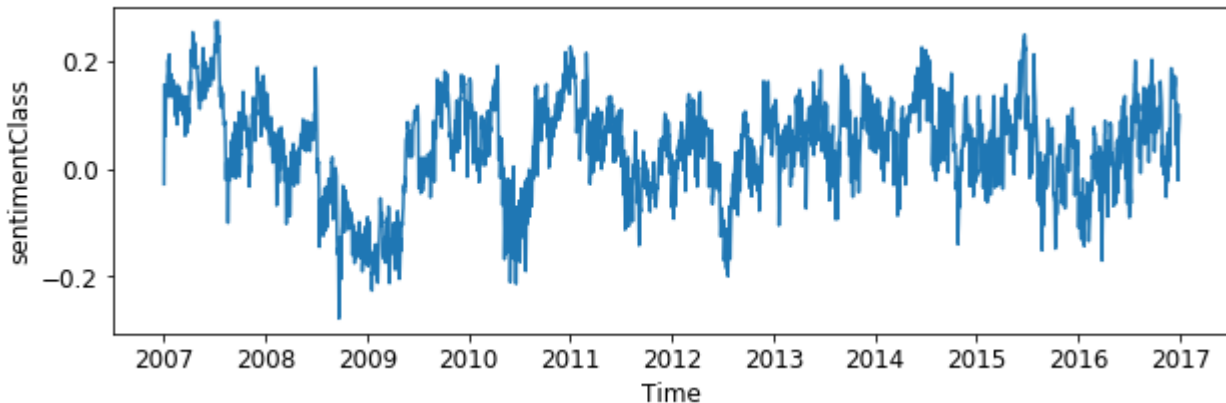
提到的评分资产的次序随时间变化图

relevance versus time



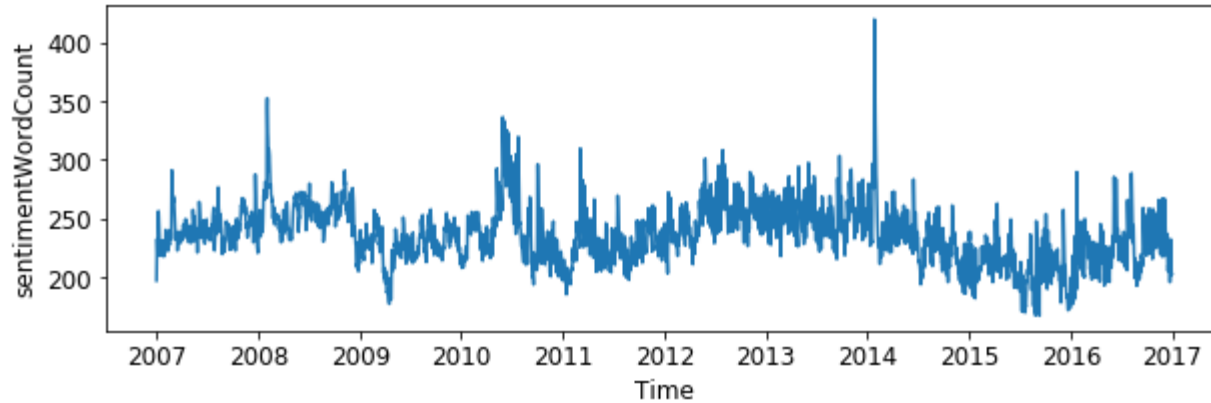
新闻项与资产的相关性随时间变化关系

sentimentClass versus time



新闻项相对于资产的主要情绪类随时间变化图

sentimentWordCount versus time



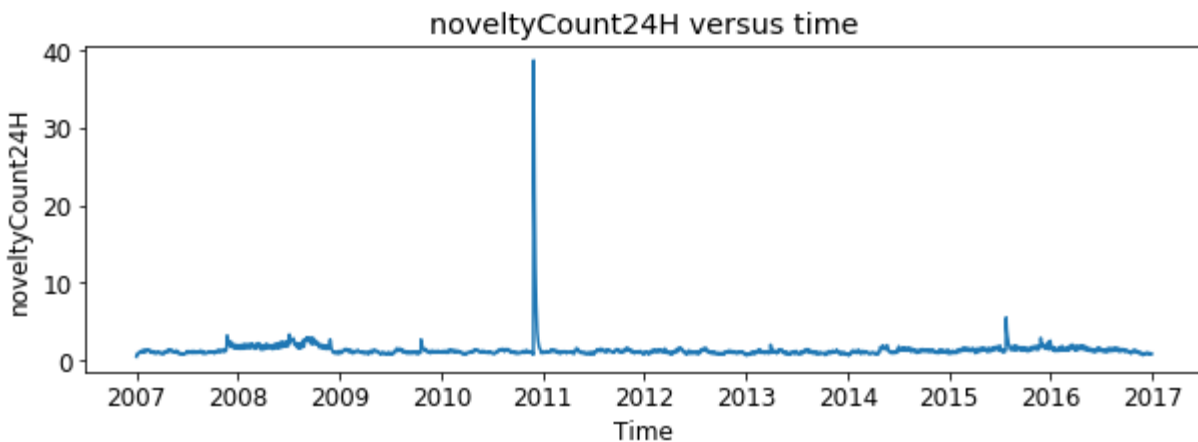
词法标记数随时间变化关系

# 探索性数据分析(EDA)

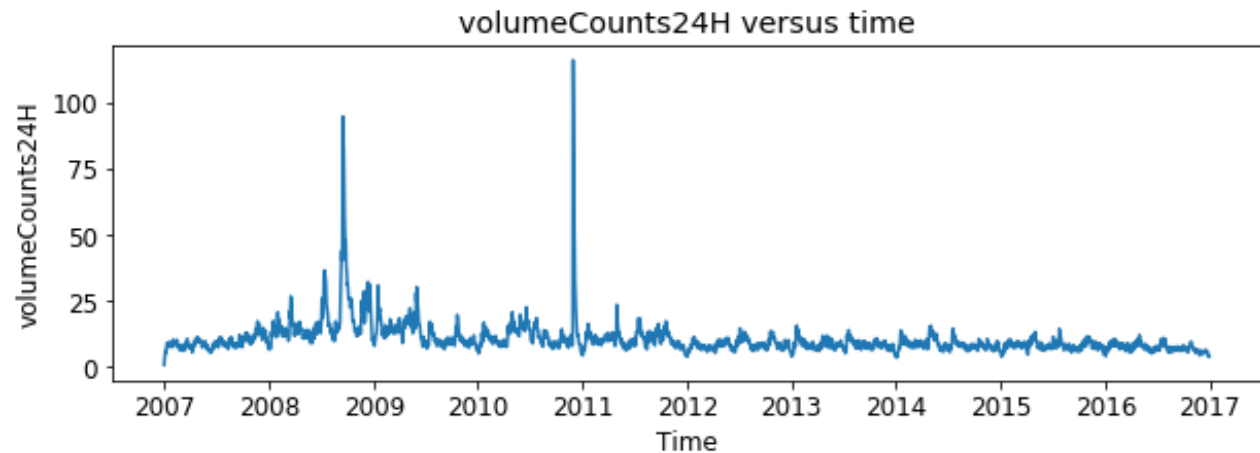
由以上图表可知，新闻中提到的**评分资产的次序**整体呈**下降趋势**，在**2014年**达到**最高峰值**，说明在2014年的新闻内容中常出现与股票有关信息。**新闻项与资产的相关性**随着时间的变化**增强**，说明新闻与股票的关系越来越紧密。新闻项相对于资产的主要情绪呈周期性波动，其中在2009年明显呈消极趋势，这与2009年的经济危机相吻合。**词法标记数**随时间变化呈**波动趋势**，其中在2014年明显增多，这说明在2014年底的新闻中明显增多了对资产预测的信息。

---

# 探索性数据分析(EDA)



特定资产的新闻项目内容的24小时新颖性  
随时间变化关系

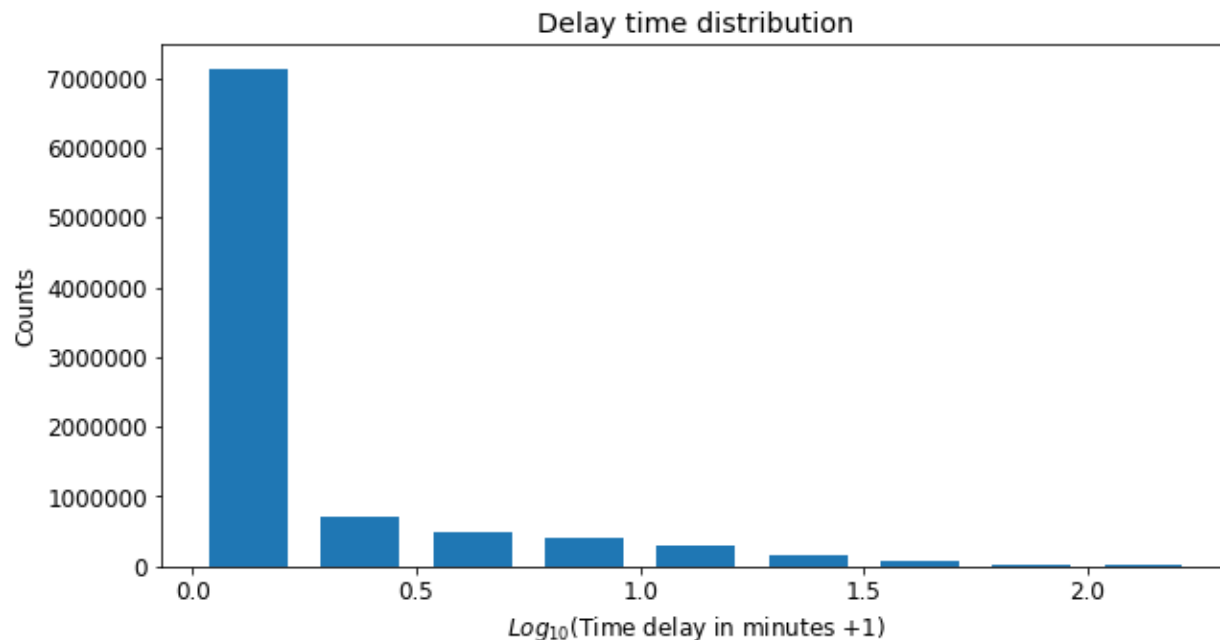


每个资产的24小时新闻量随时间变化关系

由以上图表可知，在2011年底，特定资产的新闻项目的24小时新颖性和每个资产的24小时新闻量明显增多，说明新闻的新颖性与新闻量成正比。



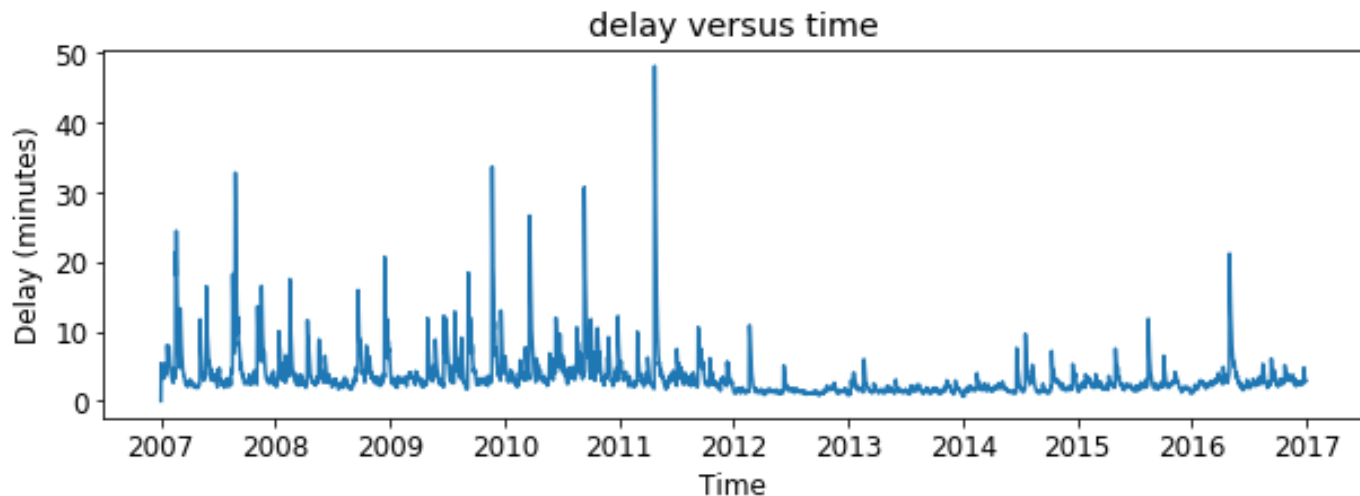
# 探索性数据分析(EDA)



延迟时间分布随时间变化关系表

由以上图表可知，延迟时间分布的时间大部分都较短，说明新闻反映股票预测趋势具有延迟性。

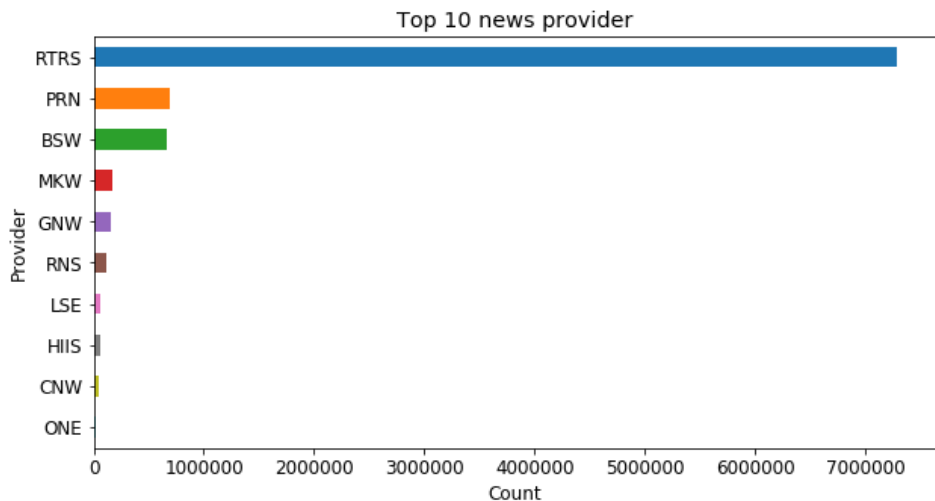
# 探索性数据分析(EDA)



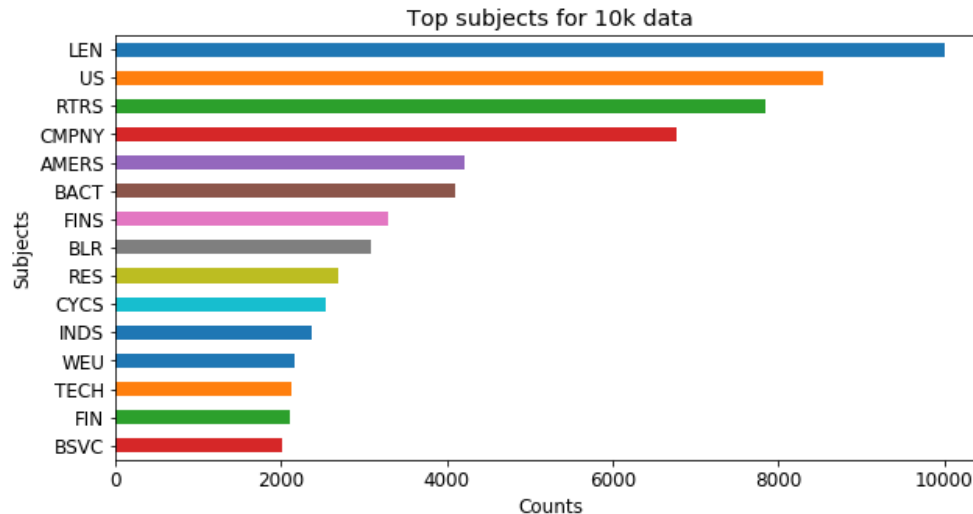
延迟时间随时间变化关系表

由上表可知，延迟时间随时间变化呈逐渐降低的趋势，说明新闻的时效性越来越强。

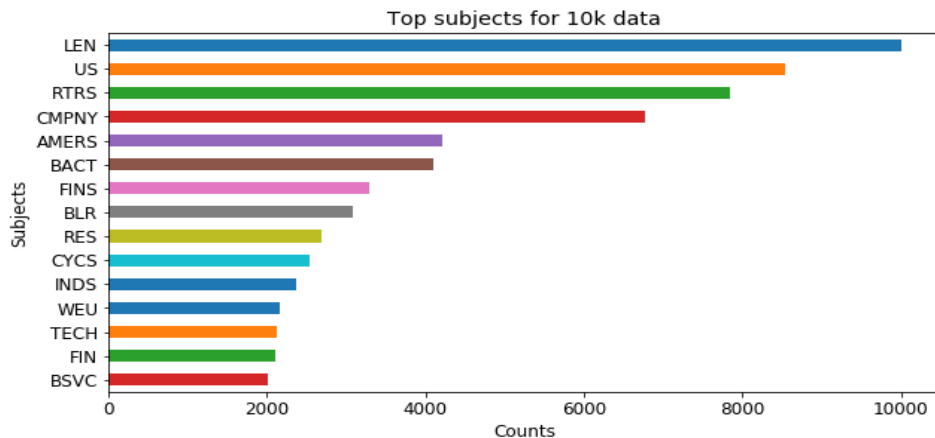
# 探索性数据分析(EDA)



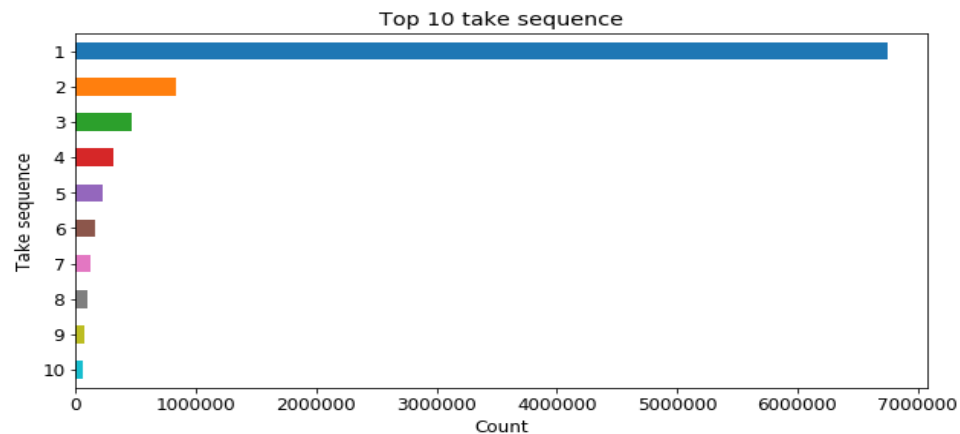
10大新闻提供商



10大新闻主题

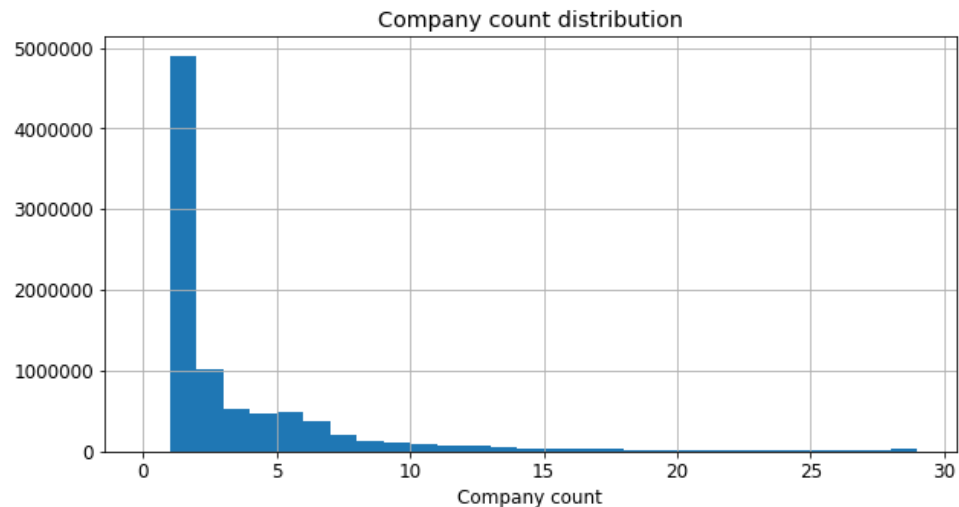


10大受众

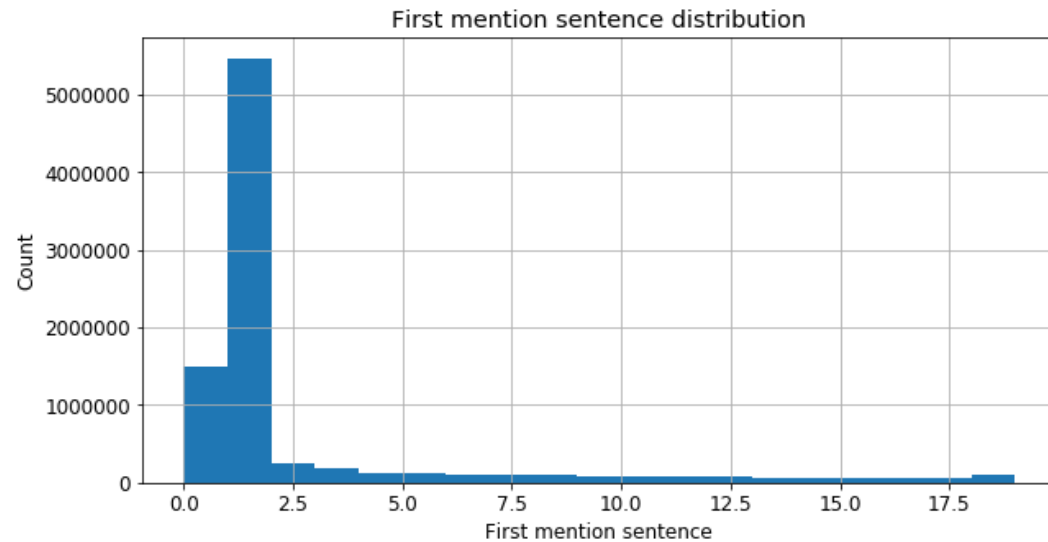


10大把序列

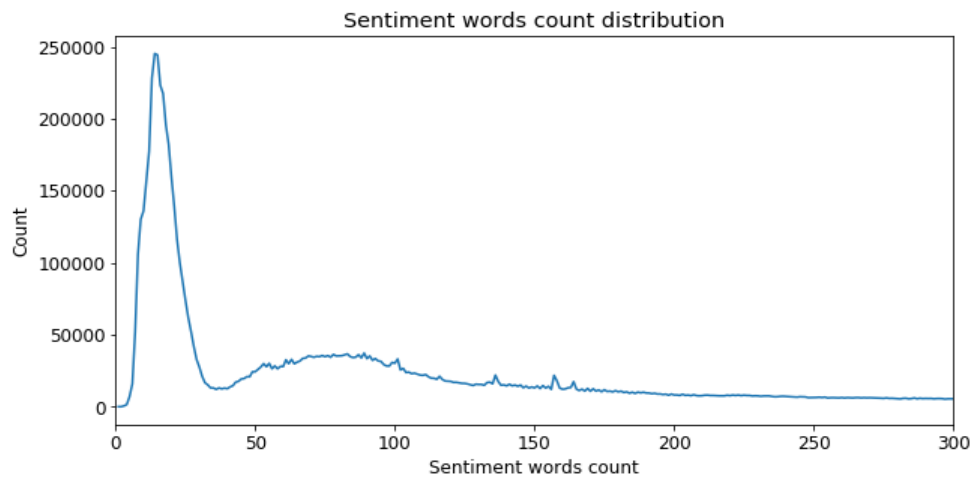
# 探索性数据分析(EDA)



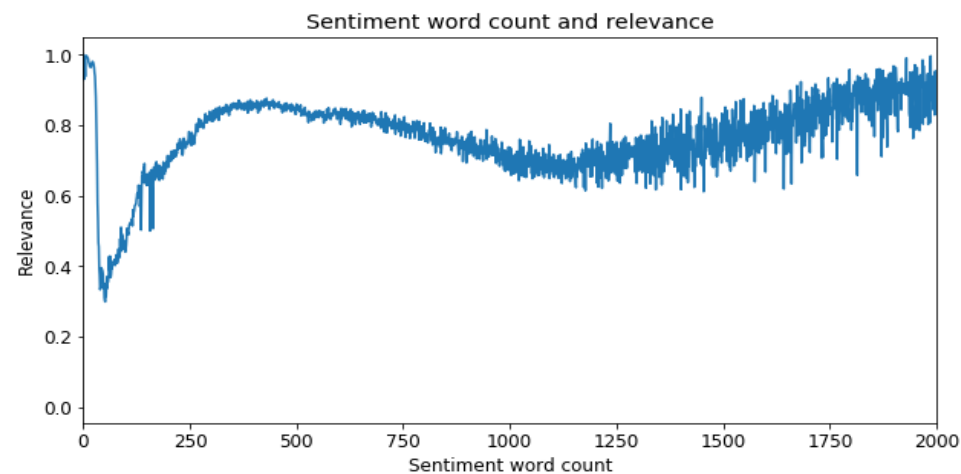
新闻中所提到公司数目分布



首次提到资产的次序分布表



情感单词的数目分布表



情感单词数目及其相关性

# 探索性数据分析(EDA)

urgency	1	-0.57	0.2	0.15	0.51	0.3	-0.45	0.044	0.46	0.072	0.091	0.1	0.11	0.098	0.13	0.14	0.15
takeSequence	-0.57	1	0.0089	-0.077	-0.28	-0.16	0.22	-0.053	-0.27	0.045	0.017	-0.00079	-0.016	0.035	-0.025	-0.044	-0.055
companyCount	0.2	0.0089	1	0.25	0.53	0.58	-0.53	-0.046	-0.049	0.12	0.12	0.12	0.12	0.066	0.033	0.017	0.0023
marketCommentary	0.15	-0.077	0.25	1	0.28	0.29	-0.28	-0.059	-0.035	0.045	0.04	0.037	0.035	0.16	0.11	0.089	0.076
sentenceCount	0.51	-0.28	0.53	0.28	1	0.52	-0.43	-0.0071	0.49	0.13	0.14	0.14	0.15	0.062	0.059	0.054	0.047
firstMentionSentence	0.3	-0.16	0.58	0.29	0.52	1	-0.64	-0.047	-0.14	0.11	0.11	0.11	0.11	0.054	0.05	0.045	0.038
relevance	-0.45	0.22	-0.53	-0.28	-0.43	-0.64	1	0.1	0.013	0.014	0.0074	0.0041	0.0014	-0.14	-0.16	-0.17	-0.17
sentimentClass	0.044	-0.053	-0.046	-0.059	-0.0071	-0.047	0.1	1	0.058	0.011	0.018	0.023	0.028	-0.09	-0.088	-0.086	-0.082
sentimentWordCount	0.46	-0.27	-0.049	-0.035	0.49	-0.14	0.013	0.058	1	-0.063	-0.055	-0.051	-0.046	-0.025	-0.0055	0.0024	0.006
noveltyCount24H	0.072	0.045	0.12	0.045	0.13	0.11	0.014	0.011	-0.063	1	0.95	0.91	0.87	0.33	0.27	0.24	0.23
noveltyCount3D	0.091	0.017	0.12	0.04	0.14	0.11	0.0074	0.018	-0.055	0.95	1	0.97	0.94	0.34	0.31	0.29	0.27
noveltyCount5D	0.1	-0.00079	0.12	0.037	0.14	0.11	0.0041	0.023	-0.051	0.91	0.97	1	0.98	0.33	0.31	0.31	0.29
noveltyCount7D	0.11	-0.016	0.12	0.035	0.15	0.11	0.0014	0.028	-0.046	0.87	0.94	0.98	1	0.32	0.31	0.31	0.31
volumeCounts24H	0.098	0.035	0.066	0.16	0.062	0.054	-0.14	-0.09	-0.025	0.33	0.34	0.33	0.32	1	0.88	0.82	0.78
volumeCounts3D	0.13	-0.025	0.033	0.11	0.059	0.05	-0.16	-0.088	-0.0055	0.27	0.31	0.31	0.31	0.88	1	0.94	0.89
volumeCounts5D	0.14	-0.044	0.017	0.089	0.054	0.045	-0.17	-0.086	0.0024	0.24	0.29	0.31	0.31	0.82	0.94	1	0.97
volumeCounts7D	0.15	-0.055	0.0023	0.076	0.047	0.038	-0.17	-0.082	0.006	0.23	0.27	0.29	0.31	0.78	0.89	0.97	1
urgency																	
takeSequence																	
companyCount																	
marketCommentary																	
sentenceCount																	
firstMentionSentence																	
relevance																	
sentimentClass																	
sentimentWordCount																	
noveltyCount24H																	
noveltyCount3D																	
noveltyCount5D																	
noveltyCount7D																	
volumeCounts24H																	
volumeCounts3D																	
volumeCounts5D																	
volumeCounts7D																	

不同特征之间的关联度分布表

# 探索性数据分析(EDA)

## 评价指标:

在这次比赛中，我们需要预测一个有符号的置信度值  $\hat{y}_{ti} \in [-1, 1]$ ，如果认为股票在未来十天内具有较大的正回报，则有正的置信度值（接近1.0）。如果认为股票具有负回报，则可以为其指定一个较大的负置信度值（接近-1.0）。如果不确定，则可以为其指定接近零的值。对于评估时间段内的每一天，我们计算：

$$x_t = \sum_i \hat{y}_{ti} r_{ti} u_{ti},$$

其中  $r_{ti}$  是工具*i*的第*t*天市场调整后的领先回报，而  $u_{ti}$  是0/1布尔变量，控制特定资产是否包含在特定日期的评分中。然后，提交分数计算为平均值除以每日  $x_t$  值的标准差：

$$\text{score} = \frac{\bar{x}_t}{\sigma(x_t)}$$



模型





# 难点

- a) 只能在Kaggle的Kernels上编写程序及运行，公司不支持原始数据下载
- b) 网络：Kaggle整个网站不稳定，经常在跑程序时出现disconnect的问题，国外服务器，网站不稳定

```
LGBMClassifier(boosting_type='dart', class_weight=None, colsample_bytree=1.0,
                importance_type='split', learning_rate=0.1, max_depth=-1,
                min_child_samples=212, min_child_weight=0.001, min_split_gain=0.0,
                n_estimators=500, n_jobs=4, num_leaves=2452, objective='binary',
                random_state=100, reg_alpha=0.0, reg_lambda=0.01, silent=True,
                subsample=1.0, subsample_for_bin=200000, subsample_freq=0)
```

[Reconnecting]

[Running]

Your kernel is now running in the cloud. Here are some things you can do with it:

- \* Use the Play button or [SHIFT]+[ENTER] to execute the current line of your script (or whatever's highlighted).
- \* Enter some code at the bottom of this Console tab and press [ENTER].

[Reconnecting]





# 难点

- a) 只能在Kaggle的Kernels上编写程序及运行，公司不支持原始数据下载
  - b) 网络：Kaggle整个网站不稳定，经常在跑程序时出现disconnect的问题，网站有些卡
  - c) 组委会特殊要求——不支持GPU计算的结果：尽管可以用Kaggle上的GPU跑程序，但是用GPU计算的结果不能提交评分（故过于复杂的模型不宜使用，先考虑快的方法如一些boosting模型、传统的机器学习方法及采用简单单元的神经网络，尝试过诸如CNN,RNN,LSTM等方法，效率非常低)
  - d) 组委会的限制给我们在**数据预处理、特征工程**提出了**更高的要求**，如outliers的处理、特征选取、是否自行基于现有特征计算出更好的新特征等
  - e) market和news两者数据不平衡，若两者都想用上，需要做好特征选取及数据融合工作
-



# Model 1 - voting LightGBM

## a) Preprocessing for Market data

- **Fill nulls - Market values:** All null data comes from market adjusted columns. We fill them up with the raw values in the same row
- **Outliers - Open to close:** the difference between open price and close price cannot be too much difference (market would corrupt otherwise). We treat these outliers by clipping the close-to-open ratio (and add as a new feature)
- **Outliers>Returns:** Return should not exceed 50% or falls below 50%. If it does, it is either noise, or extreme data that will confuse our prediction later on. We remove these extreme data.
- **Remove strange data:** Here we remove data with unknown asset name or asset codes with strange behavior.

## b) Preprocessing for News data

- **Remove outliers:** apply a clip filter to reduce too extreme data

## c) Then, process both market and news data, then merge them

---



# Model 1 - voting LightGBM

d) Data selection (Origin: 2007–2016)

- Looking at the statistics, most data behave homogeneously after 2009 (volume increase, price increase, etc.). However, **before 2009**, due to the burst of the **housing bubble** that leads to the **financial crisis** in 2008, the data behaves differently.

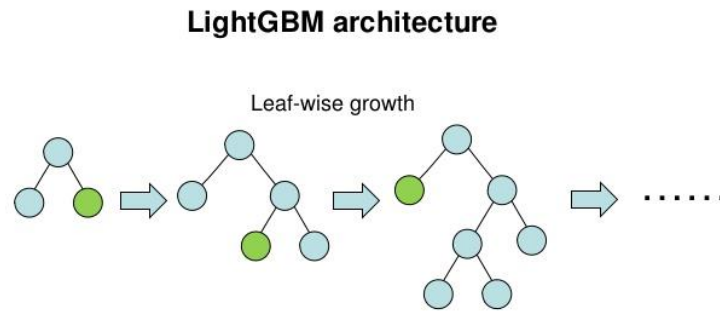
So the question to make the right prediction for this problem is: Will there be a financial crisis in the next 6 months?

If the answer is **Yes**, then we include data before 2009. If the answer is **No** then we exclude them.  
**##** we used data from data from 2009.1.1 last time **but this time we used data from 2010.1.1.**

- Random train-test split
  - Extract data: X, Y, r, u, d (which are used to calculate final score)
-

# Model 1 - voting LightGBM

e) Model building - choose LightGBM classifier



- A tree-based model, which do not require standardization
- Have tried a regression model, but a problem is that it gives close-to-0 values for most of prediction, which leads to bad result.
- Thus, I convert this problem into a classification problem: **0 for negative return and 1 for positive return**. And the target variable-return can be represented by `predicted_return = y_predict_proba[:,1] - y_predict_proba[:,0]`.



# Model 1 - voting LightGBM

## f) opt\_params参数优化：利用GridSearchCV和RandomizedSearchCV方法各寻了几次最优参数

- n\_estimators':500 - 这是生成的最大树的数目，也是最大的迭代次数
- boosting\_type ': ' dart' - LGB里面的boosting参数有gbdt, rf, dart, dross
- num\_leaves ':2452 - 一棵决策树上的叶子数
- objective: 'binary' - binary logloss(用于寻参)
- min\_child\_samples':212 - 一个叶子上数据的最小数量。可以用来处理过拟合。
- reg\_lambda ':0.01 -权重的L2正则化项。这个参数是用来控制XGBoost的正则化部分的，用于减少过拟合

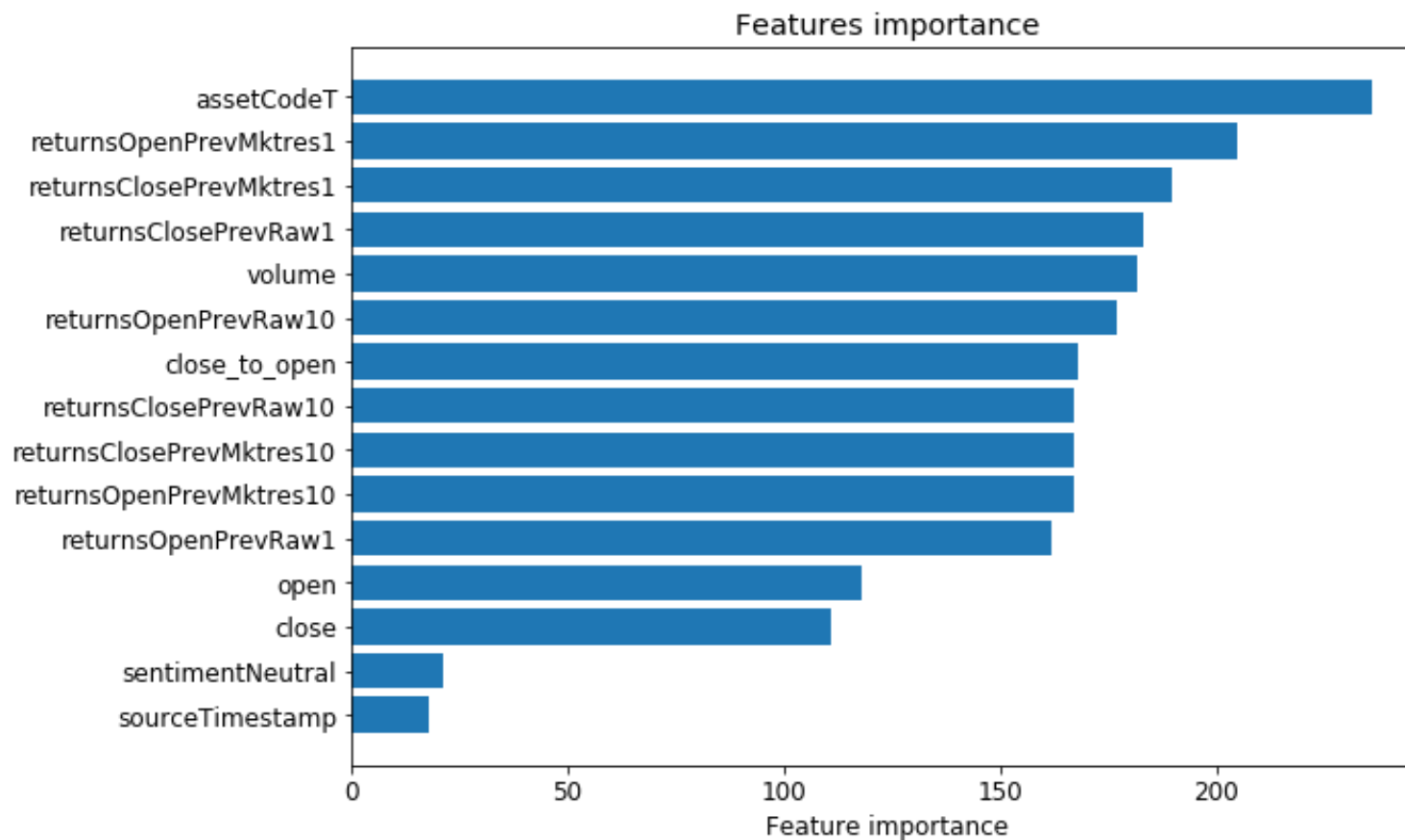
```
Training until validation scores don't improve for 40 rounds.
[20]  valid_0's binary_logloss: 0.689611      valid_1's binary_logloss: 0.690806
[40]  valid_0's binary_logloss: 0.689767      valid_1's binary_logloss: 0.690905
Early stopping, best iteration is:
[1]   valid_0's binary_logloss: 0.689386      valid_1's binary_logloss: 0.690634

LGBMClassifier(boosting_type='dart', class_weight=None, colsample_bytree=1.0,
                importance_type='split', learning_rate=0.1, max_depth=-1,
                min_child_samples=212, min_child_weight=0.001, min_split_gain=0.0,
                n_estimators=500, n_jobs=4, num_leaves=2452, objective='binary',
                random_state=100, reg_alpha=0.0, reg_lambda=0.01, silent=True,
                subsample=1.0, subsample_for_bin=200000, subsample_freq=0)
```



# Model 1 - voting LightGBM

## g) 基于树的模型可以用来评估特征重要性



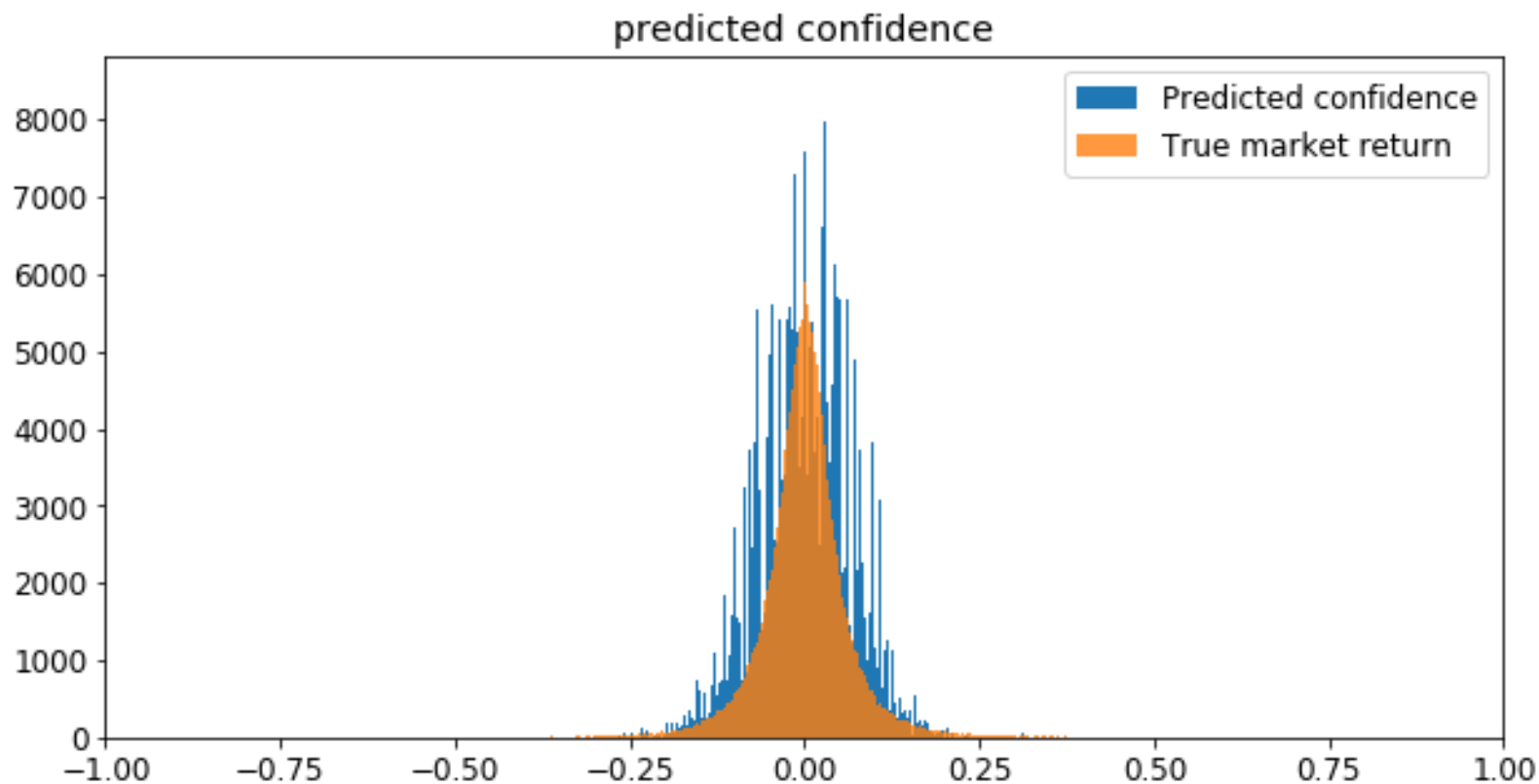


# Model 1 - voting LightGBM

h) 计算final score中的 $\hat{y}_{ti}$ , 即我们的target variable并与ground truth即returnsOpenNextMktres10对比

$$x_t = \sum_i \hat{y}_{ti} r_{ti} u_{ti},$$

$$\text{score} = \frac{\bar{x}_t}{\sigma(x_t)}$$



好像效果....



# Model 1 - voting LightGBM

## i) Inspired by <https://www.kaggle.com/skooch/lgbm-w-random-split-2>



Eric Antoine Scuccimarra

### LGBM w Random Split 2

last run 3 months ago · IPython Notebook HTML · 855 views  
using data from [Costa Rican Household Poverty Level Prediction](#) · Pub

[Notebook](#) Code Data (1) Output Comments (0) Log Versions (8) Forks (62)

Notebook

## LGMB with random split for early stopping

Edits by Eric Antoine Scuccimarra - This is a fork of <https://www.kaggle.com/mlisovyi/feature-eng-with-f1-macro>, by Misha Losvyo, with a few changes:

- Some additional features have been added.
- Some features which were previously dropped have been retained.
- Some of the code has been reorganized.
- Rather than splitting the data once and using the validation data for the LGBM early stopping, I during the training so the entire training set can be trained on. I found that this works better than

### Fit a voting classifier

Define a derived VotingClassifier class to be able to pass `fit_params` for early stopping. Vote based on LGBM models with early stopping based on macro F1 and decaying learning rate.

The parameters are optimised with a random search in this kernel: <https://www.kaggle.com/mlisovyi/lighgbm-hyperoptimisation-with-f1-macro>

```

# these parameters have not been altered from when they were originally tuned
# opt_parameters = {'colsample_bytree': 0.93, 'min_child_samples': 56, 'num_leaves': 19, 'subsampling': 0.84, 'reg_lambda': 0.5, }
opt_parameters = {'colsample_bytree': 0.88, 'min_child_samples': 90, 'num_leaves': 16, 'subsampling': 0.94, 'reg_lambda': 0.5, }
opt_parameters = {'colsample_bytree': 0.88, 'min_child_samples': 95, 'num_leaves': 25, 'subsampling': 0.94, 'reg_lambda': 0.5, }

def evaluate_macroF1_lgb(truth, predictions):
    # this follows the discussion in https://github.com/Microsoft/LightGBM/issues/1483
    pred_labels = predictions.reshape(len(np.unique(truth)), -1).argmax(axis=0)
    f1 = f1_score(truth, pred_labels, average='macro')
    return ('macroF1', f1, True)

fit_params={"early_stopping_rounds":800,
            "eval_metric" : evaluate_macroF1_lgb,
            "eval_set" : [(X_train,y_train), (X_test,y_test)],
            "eval_names": ['train', 'valid'],
            "verbose": False,
            "categorical_feature": 'auto'}

```





# Model 1 - voting LightGBM

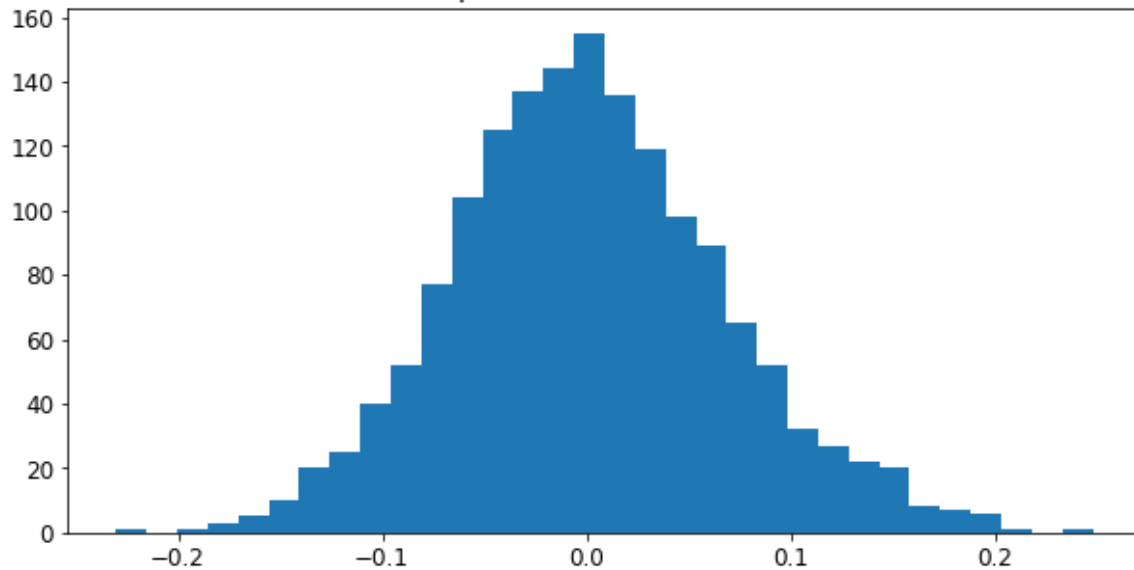
j) Construct an ensemble of multiple classifier and use soft voting to get the final result

Two Sigma: Using News to ...  
2 months to go · Top 12%

166<sup>th</sup>  
of 1508

Last Time

predicted confidence

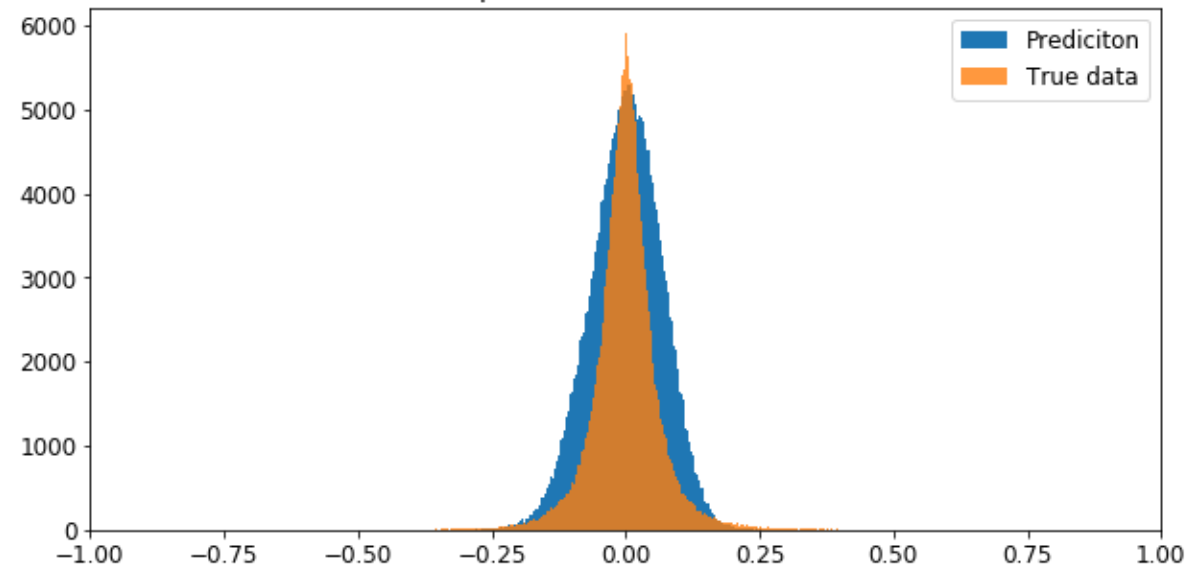


Two Sigma: Using News to ...  
a month to go · Top 5%

98<sup>th</sup>  
of 2036

This Time

predicted confidence





# Model 1 - voting LightGBM

## k) Summary

- 在这个比赛中，到后面想要继续提升分数，模型可能不是最关键的。所以现在重点是数据预处理和特征工程，删去不必要的或是整合特征(上次老师提到positive/neutral/negative sentiment合三为一)，以及时间的选取(之前认为2008金融危机就选从2009开始就好，后面发现2009是过渡期其实数据也不稳定，用2010以后的会更好)
  - 优势在于EDA给我们后续提供了很多特征上的一些参考，仍然有一些优化工作没有完成，所以现在的暂时还不是最优成绩。
  - 老问题：工作强度与效率与网站效率不相匹配.....昨晚找到了新的突破点然而网站崩了连接不上服务器
-



# Model 2 – NN For Market Data

## a) Preprocessing for **Market** data

```
cat_cols = ['assetCode']
num_cols = ['volume', 'close', 'open', 'returnsClosePrevRaw1', 'returnsOpenPrevRaw1', 'returnsClosePrevMktres1',
            'returnsOpenPrevMktres1', 'returnsClosePrevRaw10', 'returnsOpenPrevRaw10', 'returnsClosePrevMktres10',
            'returnsOpenPrevMktres10']
```

### #1 Handling categorical variables:

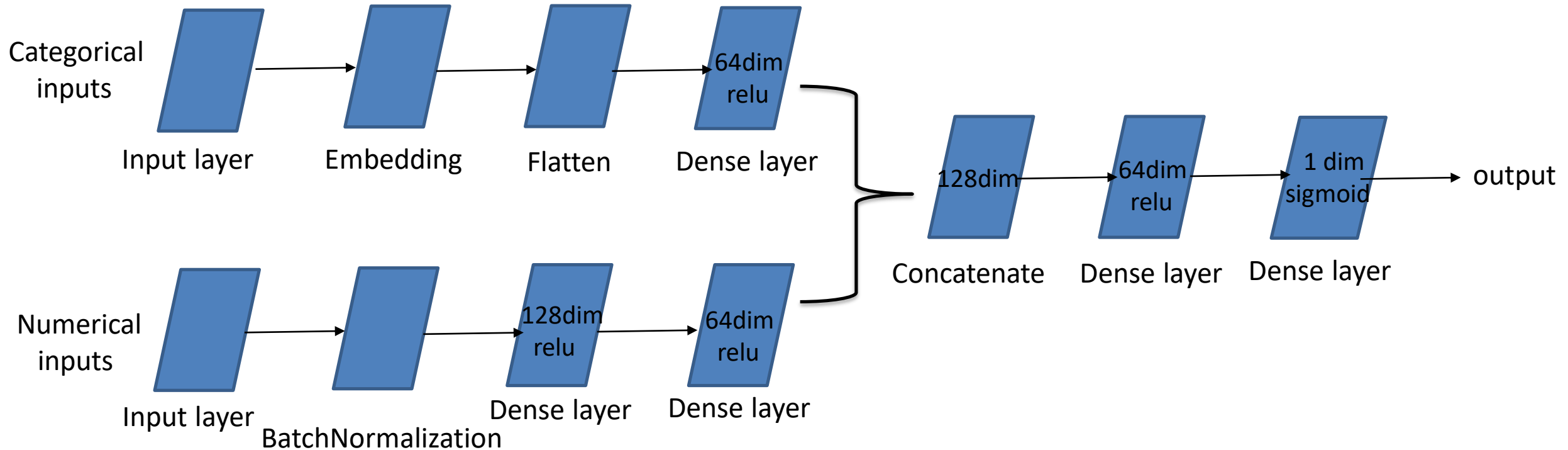
- encoding

### #2 Handling numerical variables:

- Fill NA with zeros
- Standardize features by removing the mean and scaling to unit variance  
(from sklearn.preprocessing import StandardScaler)

# Model 2 – NN For Market Data

## b) Define NN Architecture





# Model 2 – NN For Market Data

## b) Define NN Architecture

```
# Lets print our model  
model.summary()
```

Layer (type)	Output Shape	Param #	Connected to
num (InputLayer)	(None, 11)	0	
assetCode (InputLayer)	(None, 1)	0	
batch_normalization_2 (BatchNor	(None, 11)	44	num[0][0]
embedding_2 (Embedding)	(None, 1, 10)	37810	assetCode[0][0]
dense_7 (Dense)	(None, 128)	1536	batch_normalization_2[0][0]
flatten_2 (Flatten)	(None, 10)	0	embedding_2[0][0]
dense_8 (Dense)	(None, 64)	8256	dense_7[0][0]
dense_6 (Dense)	(None, 64)	704	flatten_2[0][0]
concatenate_2 (Concatenate)	(None, 128)	0	dense_8[0][0] dense_6[0][0]
dense_9 (Dense)	(None, 64)	8256	concatenate_2[0][0]
dense_10 (Dense)	(None, 1)	65	dense_9[0][0]

优化器和loss函数:

```
model = Model(inputs = categorical_inputs + [numerical_inputs], outputs=out)  
model.compile(optimizer='adam', loss=binary_crossentropy)
```

参数:

```
Total params: 56,671  
Trainable params: 56,649  
Non-trainable params: 22
```



# Model 2 – NN For Market Data

c) Train NN model

模型保存到: [model.hdf5](#)

本次训练当中, 耗时3个 epoch, 总时长约13分钟。

```
Train on 3665660 samples, validate on 407296 samples
Epoch 1/3
3665660/3665660 [=====] - 267s 73us/step - loss: 0.6826 - val_loss:
0.6831

Epoch 00001: val_loss improved from inf to 0.68306, saving model to model.hdf5
Epoch 2/3
3665660/3665660 [=====] - 266s 73us/step - loss: 0.6821 - val_loss:
0.6849

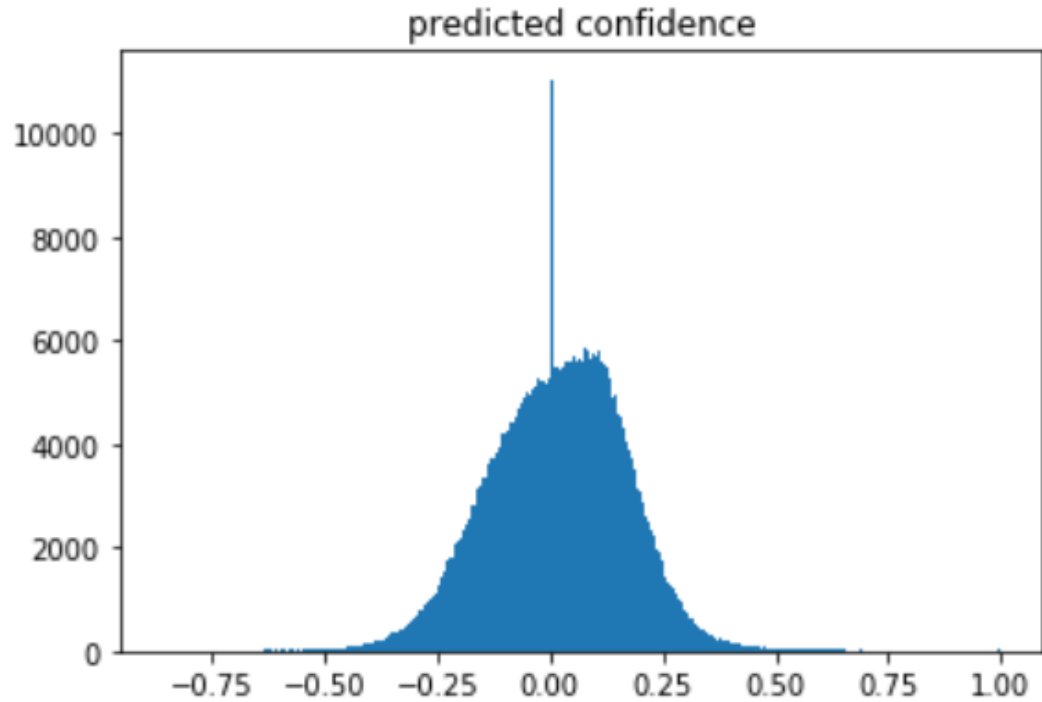
Epoch 00002: val_loss did not improve from 0.68306
Epoch 3/3
3665660/3665660 [=====] - 265s 72us/step - loss: 0.6816 - val_loss:
0.6823

Epoch 00003: val_loss improved from 0.68306 to 0.68228, saving model to model.hdf5
```



# Model 2 – NN For Market Data

## d) Evaluation of Validation Set



Market数据总共有四百多万条,  
我们取出1/10的数据集用来验证,  
计算我们模型的得分为0.6896左右

```
print(score_valid)
```


```
0.6896094517661229
```




# Model 2 – NN For Market Data

e) the final score


得分: **0.67552**, 排名: 前**11%**


158 new LuWantong  0.67552 6 1m


**Your Best Entry** ↑

Your submission scored 0.67552, which is an improvement of your previous score of 0.65269. Great job!  **Tweet this!**

**1 Active Competition**

  
TWO SIGMA

**Two Sigma: Using News to Predict Stock Movements**  
Use news analytics to predict stock price performance  
*Featured* · 2 months to go ·  news agencies, time series, finance, money

 **158/1485**  
Top 11%

No more competitions to show



# Feature Selection

## a) Market data

主要数据说明:

“open” : 股票开盘价

“close” : 股票收盘价

“Prev” : 向后看

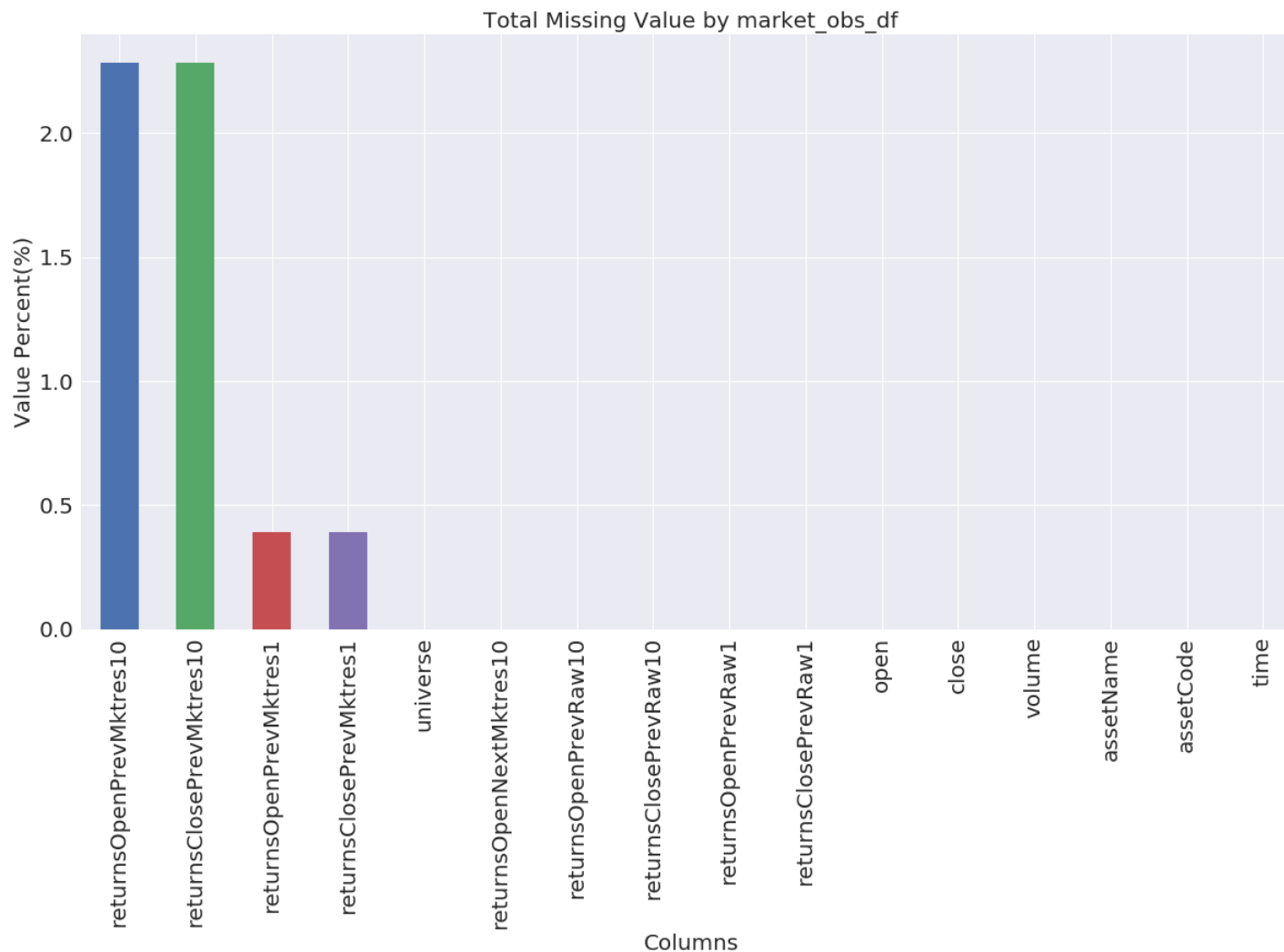
“Next” : 向前看

“Raw” : 原始数据

“Mktres” : 考虑市场残差后的数据

“1” : 数据视野为1天

“10” : 数据视野为10天



# Feature Selection

## b) News data

主要数据说明:

“sentimentNegative” : 新闻情感为负向的概率

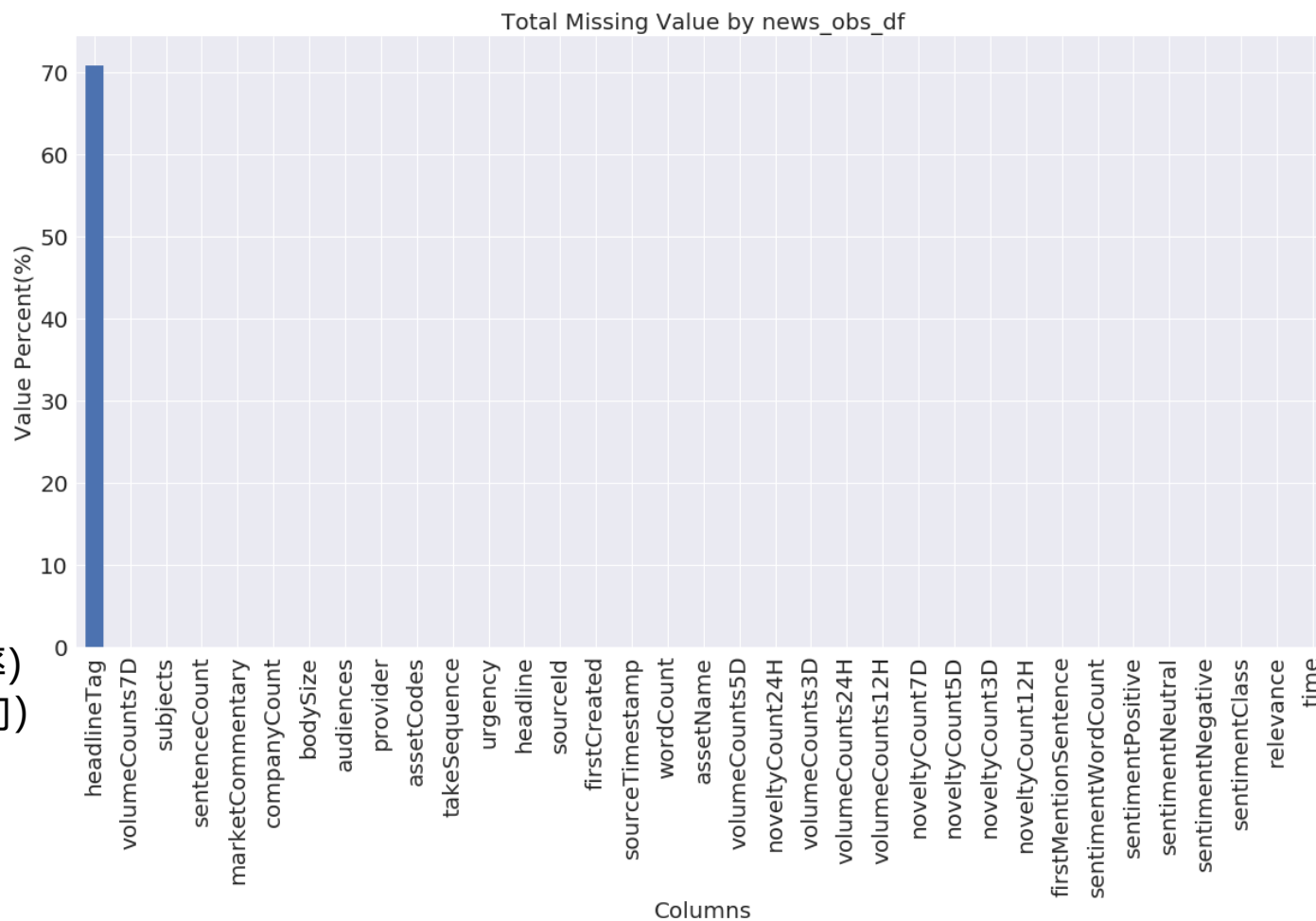
“sentimentNeutral” : 新闻情感为中性的概率

“sentimentPositive” : 新闻情感为正向的概率

“sentimentClass” : 新闻情感类型

(比较上述三者概率, 取概率最大项为该条新闻概率)

(“-1”代表负向, “0”代表中性, “1”代表正向)



# Feature Selection

## a) Market data Selection

**时间选择:** 2010年至今的数据

**主要原因:** 太过久远数据, 由于政策等金融外部环境与当今不符合, 参考意义不大

```
market_train_df['time'] = market_train_df['time'].dt.date
```

```
market_train_df = market_train_df.loc[market_train_df['time'] >= date(2010, 1, 1)]
```

---

# Feature Selection

## a) Market data Selection

### 特征选择:

选择

'returnsClosePrevMktres10', 'returnsClosePrevRaw10', 'open', 'close' 四个特征, 分别扩充平均值、最大值、最小值三个方面的特征, 窗口大小选择为3,7,14三种

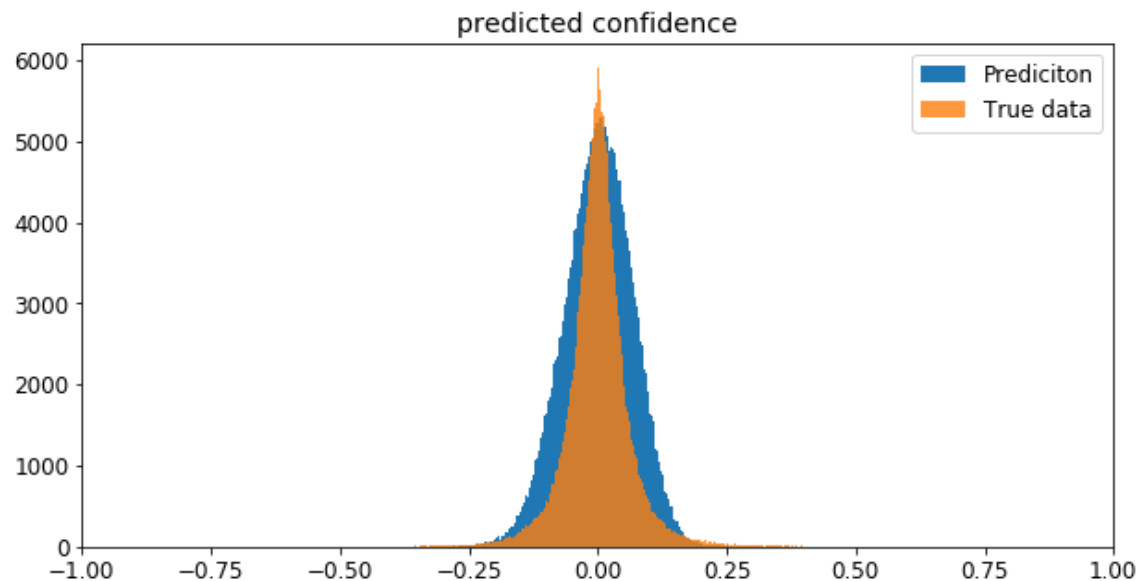
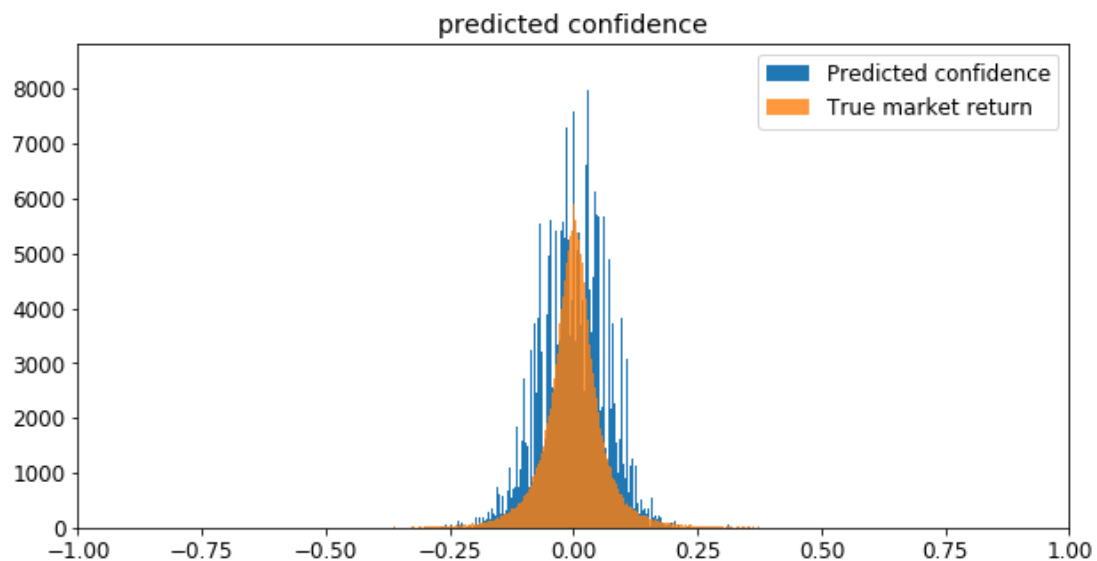
**主要原因:** 由于使用原始特征样本直接预测, 预测结果波动性过大, 加入数据局部特征, 可减小预测结果波动性

```
Index(['time', 'assetCode', 'assetName', 'volume', 'close', 'open',
      'returnsClosePrevRaw1', 'returnsOpenPrevRaw1',
      'returnsClosePrevMktres1', 'returnsOpenPrevMktres1',
      'returnsClosePrevRaw10', 'returnsOpenPrevRaw10',
      'returnsClosePrevMktres10', 'returnsOpenPrevMktres10',
      'returnsOpenNextMktres10', 'universe',
      'returnsClosePrevMktres10_lag_3_mean',
      'returnsClosePrevMktres10_lag_3_max',
      'returnsClosePrevMktres10_lag_3_min',
      'returnsClosePrevMktres10_lag_7_mean',
      'returnsClosePrevMktres10_lag_7_max',
      'returnsClosePrevMktres10_lag_7_min',
      'returnsClosePrevMktres10_lag_14_mean',
      'returnsClosePrevMktres10_lag_14_max',
      'returnsClosePrevMktres10_lag_14_min',
      'returnsClosePrevRaw10_lag_3_mean', 'returnsClosePrevRaw10_lag_3_max',
      'returnsClosePrevRaw10_lag_3_min', 'returnsClosePrevRaw10_lag_7_mean',
      'returnsClosePrevRaw10_lag_7_max', 'returnsClosePrevRaw10_lag_7_min',
      'returnsClosePrevRaw10_lag_14_mean', 'returnsClosePrevRaw10_lag_14_max',
      'returnsClosePrevRaw10_lag_14_min', 'open_lag_3_mean', 'open_lag_3_max',
      'open_lag_3_min', 'open_lag_7_mean', 'open_lag_7_max', 'open_lag_7_min',
      'open_lag_14_mean', 'open_lag_14_max', 'open_lag_14_min',
      'close_lag_3_mean', 'close_lag_3_max', 'close_lag_3_min',
      'close_lag_7_mean', 'close_lag_7_max', 'close_lag_7_min',
      'close_lag_14_mean', 'close_lag_14_max', 'close_lag_14_min'],
      dtype='object')
```

# Feature Selection

模型: LightGBM  
集成学习: 软投票、取平均

c) result



**Two Sigma: Using News to Predict Stock Movements**

Use news analytics to predict stock price performance

Featured · a month to go · news agencies, time series, finance, money



92/2026  
Top 5%

排名: 前**5%**

# 团队分工及计划



# 团队分工

团队成员	负责工作
徐哲、张凌霄、路万通	建立模型、特征工程、编写代码与调参
许卓群、于苗苗	项目背景调研、编程对数据进行EDA分析及总结

下步计划:

- 完成一些因昨晚Kaggle的Kernel崩掉没有完成的额外数据预处理及特征选取工作
- 完成书面报告
- 比赛还有一段时间截止，持续跟进



Thank you

IIP Team 5