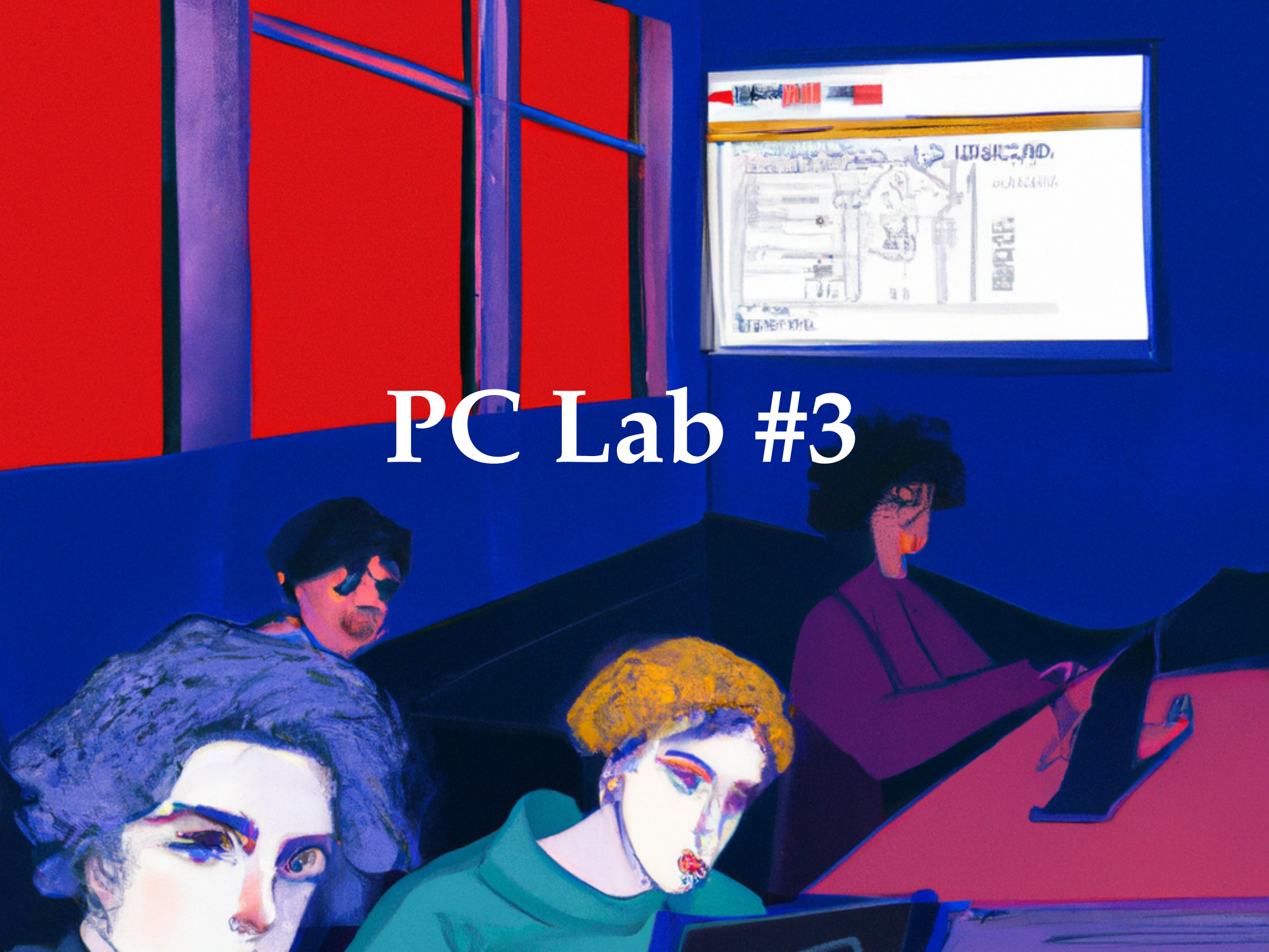
20598 – Finance with Big Data

PC Lab #3: Creating a Factor from Text Data (Week 4)

Clément Mazet-Sonilhac
clement.mazetsonilhac@unibocconi.it

Department of Finance, Bocconi University


PC Lab #3

PC Labs Grading

- PC Labs solutions are submitted as Jupyter Notebooks, via email
  - Email title : PCLab#3 - Group X - Name1 Name2 Name3
  - Your Jupyter Notebook starts in the same way (same .ipynb name)
  - Tell me (again) how long did it take

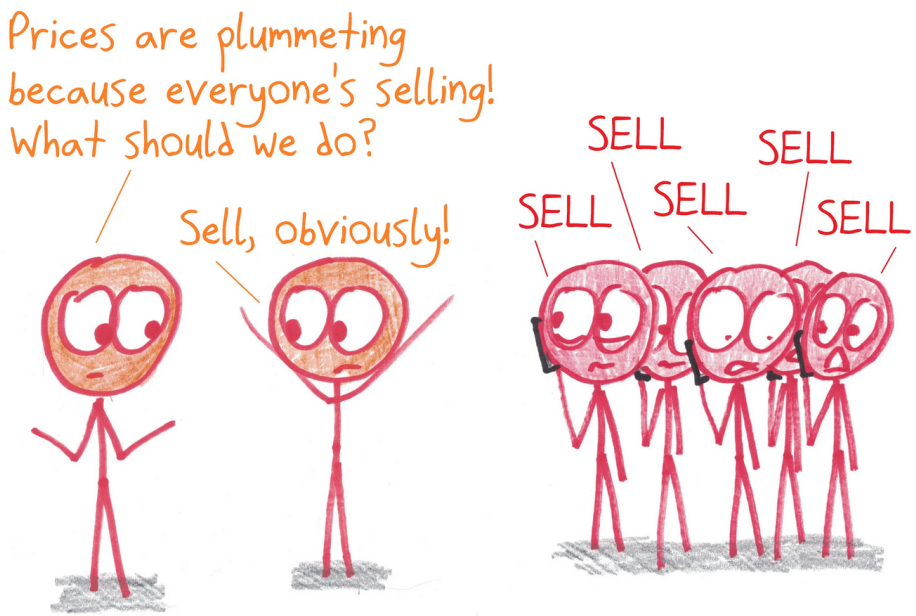
- PC Labs grade will depend on :
  - Your ability to submit it before the deadline **(Friday, midnight)**
  - The quality of your code (comments, readability, use of functions, etc.)
  - The structure of the Jupyter Notebook: well organized, explain what you are doing and why
  - Your ability to complete the tasks and innovate
  - You should maybe produce less, but more *useful* outputs

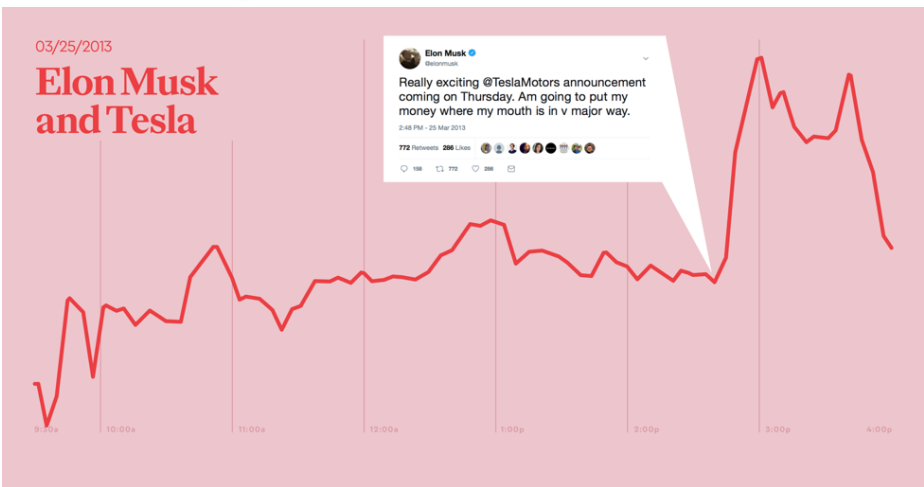

StockCats
@StockCats
Follow
EMERGENCY ALERTS
now
Emergency Alert
THE S&P 500 HAS GONE NEGATIVE ON THE DAY.
SEEK IMMEDIATE SHELTER. THIS IS NOT A DRILL.
Slide for more
1:52 PM - 16 Jan 2018
Michael B
The ma
compa
$4,095
The ma
500 co
$4,092
Downtown Josh Brown
@ReformedBroker
Follow
this person knocked $3 billion off the mar
value of Snapchat with a tweet, just in cas
you're curious about what inning humanit
in.
Kylie Jenner @KylieJenner
last night i had cereal with milk for the first time. life changing.
9:50 PM - 18 Sep 2018
The_Real_Fly
@The_Real_Fly
Follow
TRADERS AT OPEN VS CLOSE TODAY
4:56 PM - 29 Oct 2018

Goals

- Manipulate and visualize financial Tweets
- Clean text data
- Train a model to predict Tweets' sentiment
- Compute a measure of media attention

Big picture context



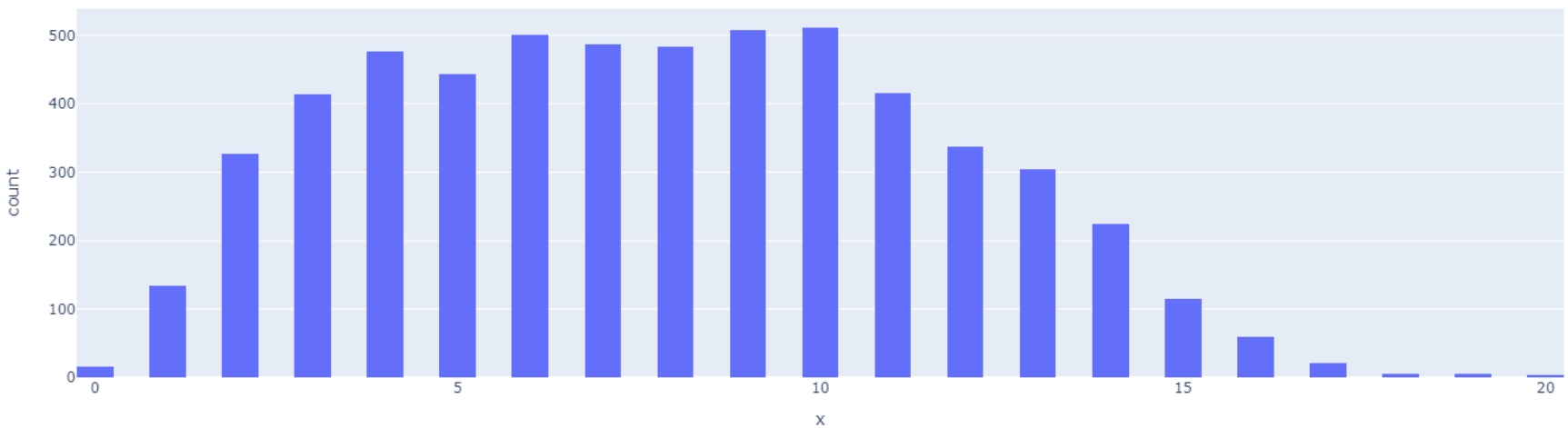


- You've just been hired by a sophisticated hedge-fund
- The hedge-fund manager is interested in Twitter's predictive power
- He asks you to perform sentiment analysis on a sample of recent financial tweets...
- ... and to build a firm-level measure of media attention: that may be a great factor idea !

Task #1: Basic manipulation and descriptive statistics

- Import the `Data_PCLab3_Twitter_Stock_Sentiment.csv` data and describe the sample (data available on BBoard)

- How many tweets, how many words per tweets, distribution of number of words per tweets, average sentiment, etc.



Task #2 : Cleaning and visualization

- Usual cleaning steps: remove punctuation, stopwords, short words, etc.
- Try your cleaning on this sentence: `$I love AI & Machine learning applied to Finance...!! ;)`
- Plot a word cloud for text with positive and negative sentiment separately
- What is the number of unique words ?

Task #3 : Sentment analysis

If any of those steps are not crystal clear, please tell me now (or shoot me an email soon after the class)

- More usual steps: Tokenizing the text and padding
  - Tokenize: vectorize text corpus, transform text into numbers
  - Padding: make all sentences the same lenght (fill with 0 short sentences)
- Split the sample in a train / test dataset (test = 10% of the total sample)
- Train model **of your choice** (RNN, LTSM, etc.) to predict the sentiment (1 or 0) on the test sample (you could use embedding layer to reduce the dimension of the problem)
- Plot the confusion matrix and compute the accuracy score (e.g., with sklearn function `accuracy`)

Task #4 : Sentment analysis - **Optional**

- Use transformers (BERT from Huggingface) to perform another sentiment analysis and compare to the sentiment value in the data
  - Use pipeline: `from transformers import pipeline`
  - Use the already created library: text-classification or sentiment-analysis
- What is the performance of the algorithm on financial tweets ? I.e., how does BERT classifies the Tweets compared to the original classification you have ?
- Hint: check the Huggingface website
- To go further: check FinBERT

Task #5 : Measuring media attention

- Use the list of tickers gathered during last PC Lab (see the web-scrapping part) to compute the number of tweets about each stock
  - e.g., `AAPL: 36 tweets, 12 negative, 24 positive`
- Rank the stocks by their amount of total media attention, positive and negative media attention
- **Optional:** Using the stock prices data on our 8 stocks (or more from the web-scrapping task), do you see a correlation between media attention and excess return ?
  - If yes, could Twitter attention is likely to be a good factor ?

Packages you may need

- Among others: `wordcloud`, `nltk.stem`, `nltk.corpus`, `nltk.tokenize`, `gensim`, `tensorflow`, `string.punctuation`, `sklearn`, `etc.`