CONNECTION INSTRUCTIONS

- Wi-Fi SSID: GTC_Hands_On
  Password: HandsOnGpu
- 登入 nvlabs.qwiklab.com
  尋找classroom:
  Deep Learning for Finance Trading Strategy
- 請先不要點選任何課程
- 需要任何協助，請詢問助教
- 講義下載點: https://goo.gl/zKuHbr

Trading Strategy for Finance using LSTMs

Andrew Liu, Ph.D.

Certified Instructor, NVIDIA Deep Learning Institute
Solution Architect
NVIDIA Corporation

DEEP LEARNING INSTITUTE

DLI Mission

Helping people solve challenging problems using AI and deep learning.

- Developers, data scientists and engineers
- Self-driving cars, healthcare and finance
- Training, optimizing, and deploying deep neural networks

TOPICS

- Lab Perspective
- Lab
  - LSTMs
  - Two Sigma Investment Dataset in Kaggle
  - Step by Step Implementation
  - How to Use the Predictions for Trading Strategy
  - Next Steps
  - LSTM for Predictive Maintenance

LAB PERSPECTIVE

WHAT THIS LAB IS

- An introduction to:
  - Financial Terminology
  - Financial Time Series Data
  - Tensorflow
  - Data organization using TensorFlow, Pandas, and Numpy
  - TensorBoard

- Hands-on exercises using TensorFlow for trading strategy

WHAT THIS LAB IS NOT

- Intro to machine learning from first principles
- Rigorous mathematical formalism of neural networks
- Survey of all the features and options of TensorFlow, Pandas or other tools
- Complete trading strategy that generates profit and loss curve (P&L)

ASSUMPTIONS

- You are familiar with Long Short Term Memory (LSTM) Networks
- Helpful to have:
  - TensorFlow experience
  - Python experience

TAKE AWAYS

- Understanding the methods for trading strategies
- Ability to setup and train a LSTM network
- Enough info to start using TensorFlow to learn from your own data

Financial Terminologies

- Trading Strategy
- Instrument
- Security
- Stock
- Return

Financial Terminologies

- Long Position
- Short Position
- Fundamental Analysis
- Technical Analysis

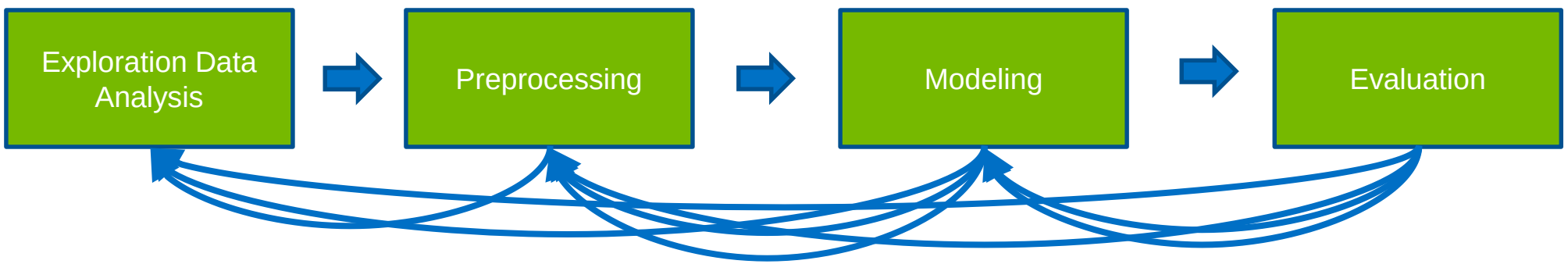
LAB

Trading Strategy for Finance using LSTMs

- Two Sigma Investment's data on Kaggle
- Recurrent Neural Networks
    - Long Short Term Memory Networks
- Data preparation
- Training and Testing
- Evaluating the predicted signal for trading strategy

Data Science Process

High overview



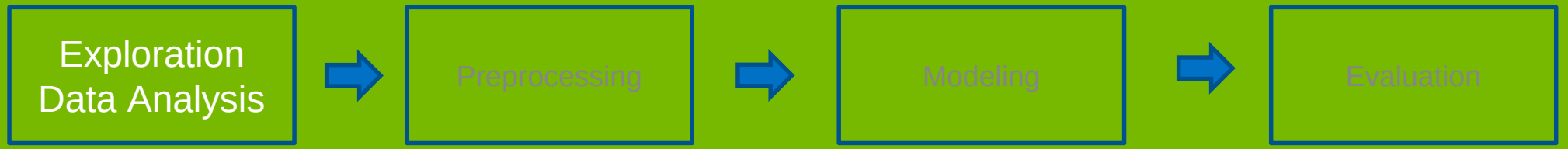
- Statistical Test
- Descriptive analysis
- Correlation matrix
- …

- Feature selection
- Filter / Wrapper
- Normalization
- Missing values processing
- …

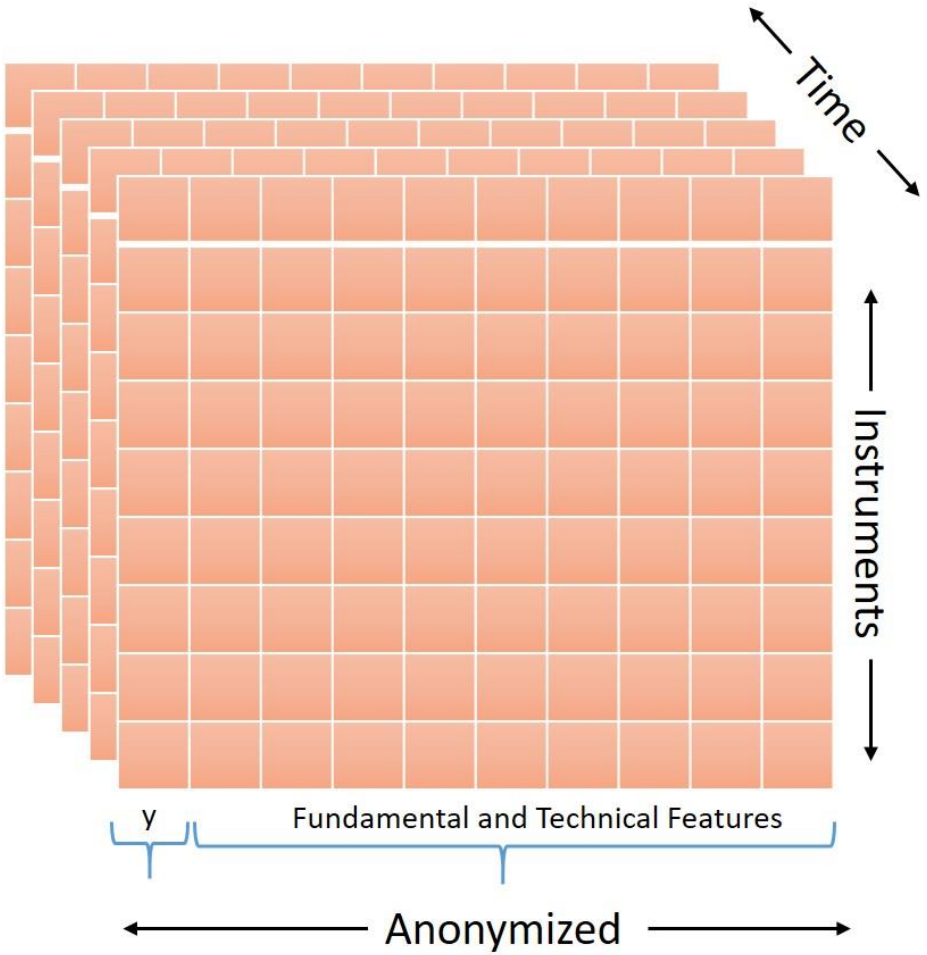
- Logistic Regression
- Classification/Regression?
- Ensemble?
- LSTM / Autoencoder / CNN?
- …

- Accuracy
- Recall
- Precision
- Pearson Correlation
- …

EDA


Exploration
Data Analysis
Preprocessing
Modeling
Evaluation

Two Sigma Investment's dataset on Kaggle



Fundamental Features: Macroeconomic factors (overall economy, industry conditions, financial conditions), revenues, earnings, future growth, profit margins, etc.

Technical Features: Price movements, analytical and statistical tools like mean, standard deviation, moving averages, etc.

"y" scalar variable: Return of the instrument

EDA 1

There are ? columns, ? rows present in the dataset.

- ? id column
- ? timestamp column
- ? columns with name prefix 'derived’
- ? columns with name prefix 'fundamental’
- ? columns with name prefix 'technical’
- ? target variable named ‘y’
- ? unique ids

EDA 1

There are 111 columns, 1710756 rows present in the dataset.

- 1 id column
- 1 timestamp column
- 5 columns with name prefix 'derived’
- 63 columns with name prefix 'fundamental’
- 40 columns with name prefix 'technical’
- 1 target variable named ‘y’
- 1424 unique ids

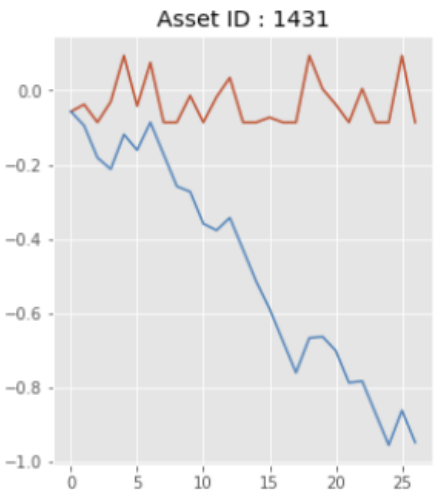
EDA 2

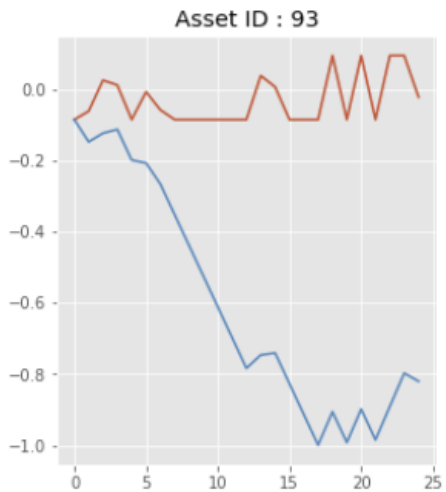
Is there something wrong?

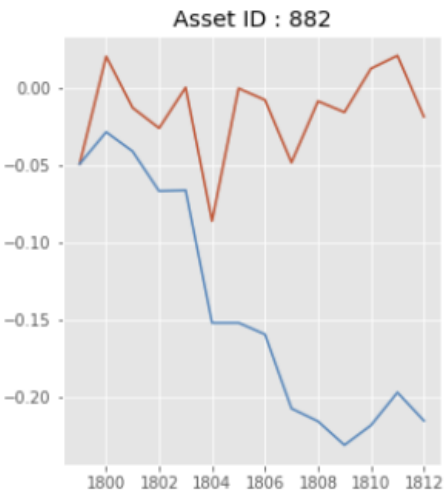
	id	timestamp	derived_0	derived_1	derived_2	derived_3	derived_4	fundamental_0
count	1.710756e+06	1.710756e+06	1.637797e+06	1.629727e+06	1.312105e+06	1.561285e+06	1.304298e+06	1.686809e+06
mean	1.093858e+03	9.456257e+02	-4.537569e+00	7.729437e+11	-3.321289e-01	-5.047151e-01	1.803233e+01	-2.041142e-02
std	6.308563e+02	5.195685e+02	2.497790e+02	7.620848e+13	6.521051e+01	1.020845e+02	9.260062e+02	2.496619e-01
min	0.000000e+00	0.000000e+00	-2.017497e+04	-7.375435e-02	-9.848880e+03	-3.434176e+04	-8.551914e+03	-2.344957e+00
25%	5.500000e+02	5.040000e+02	-1.449710e-01	-2.956479e-02	-5.967524e-02	-1.655826e-01	-1.057050e-01	-1.996543e-01
50%	1.098000e+03	9.560000e+02	-8.368272e-04	5.523058e-03	2.109505e-02	2.475614e-03	1.175234e-02	-4.064488e-02
75%	1.657000e+03	1.401000e+03	1.199108e-01	1.078554e-01	1.952209e-01	3.037236e-01	1.556464e-01	1.303819e-01
max	2.158000e+03	1.812000e+03	3.252527e+03	1.068448e+16	3.823001e+03	1.239737e+03	6.785965e+04	1.378195e+00

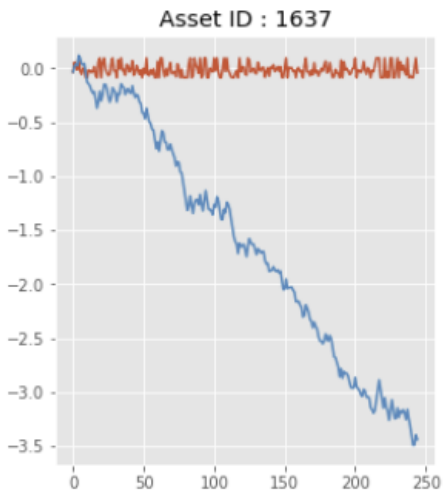
EDA 3

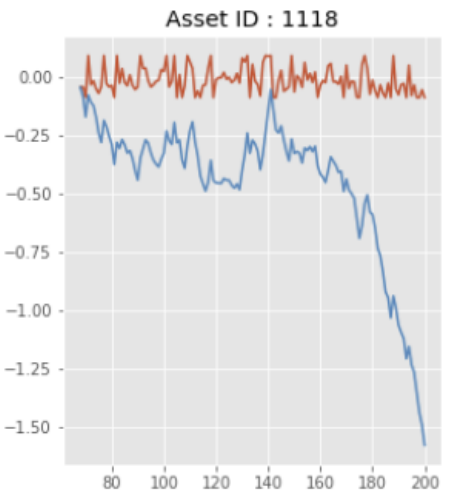
Last 5 Assets







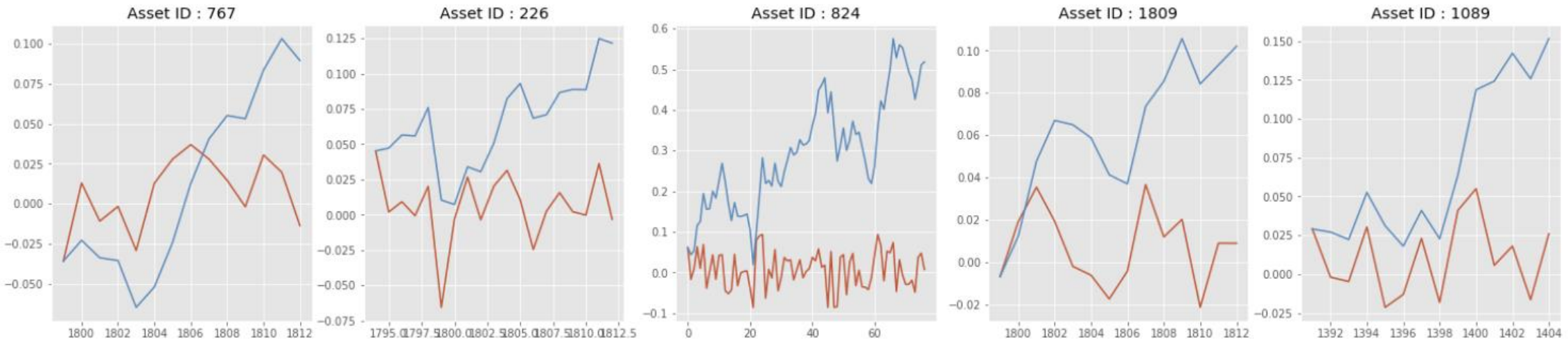





```
temp_df = df.groupby('id')['y'].agg('mean').reset_index().sort_values(by='y')
temp_df.head()
```

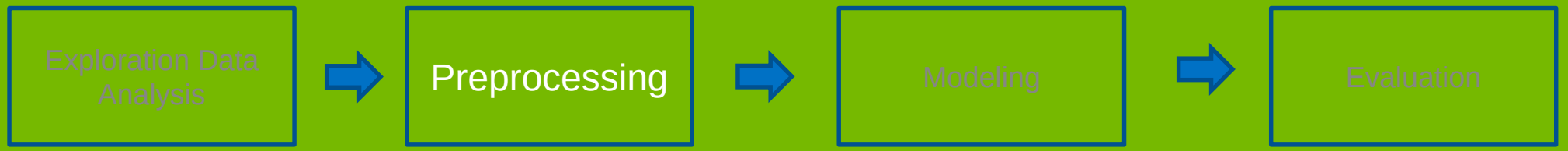
EDA 4

First 5 assets




```
temp_df = df.groupby('id')['y'].agg('mean').reset_index().sort_values(by='y')
temp_df.tail()
```

Preprocessing



Preprocess

What preprocess that you think is necessary for this dataset?

Preprocess

./prepareData.py

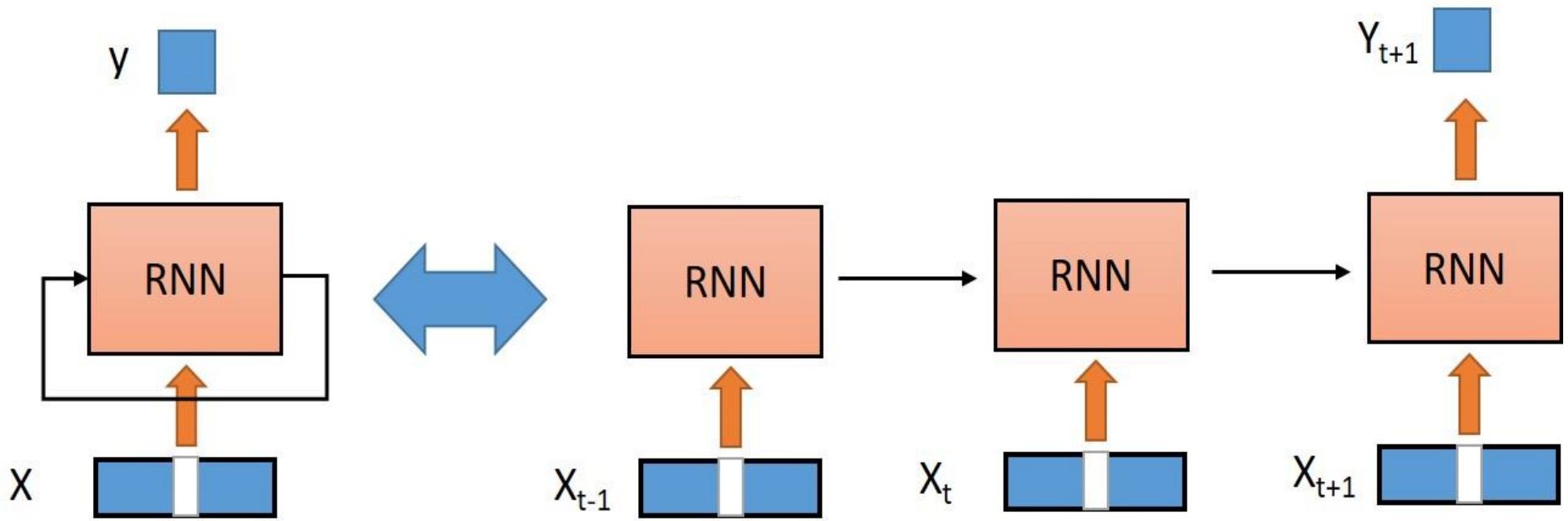
What preprocess that you think is necessary for this dataset?

- Missing values
- Outliers
- Create new features
- Normalization

RNN

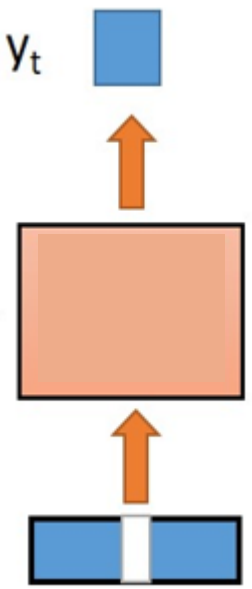
Recurrent Neural Networks

A neural network with memory

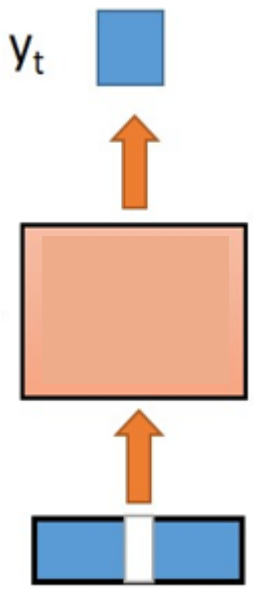


RNN cell types

How you deal with your memories

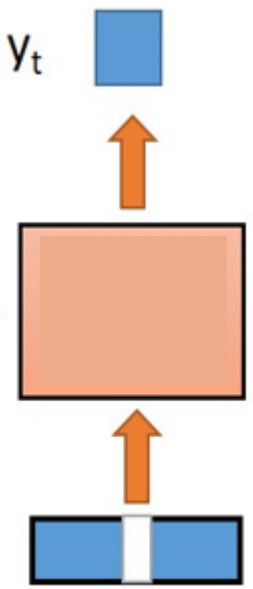


Simple RNN cell



GRU cell

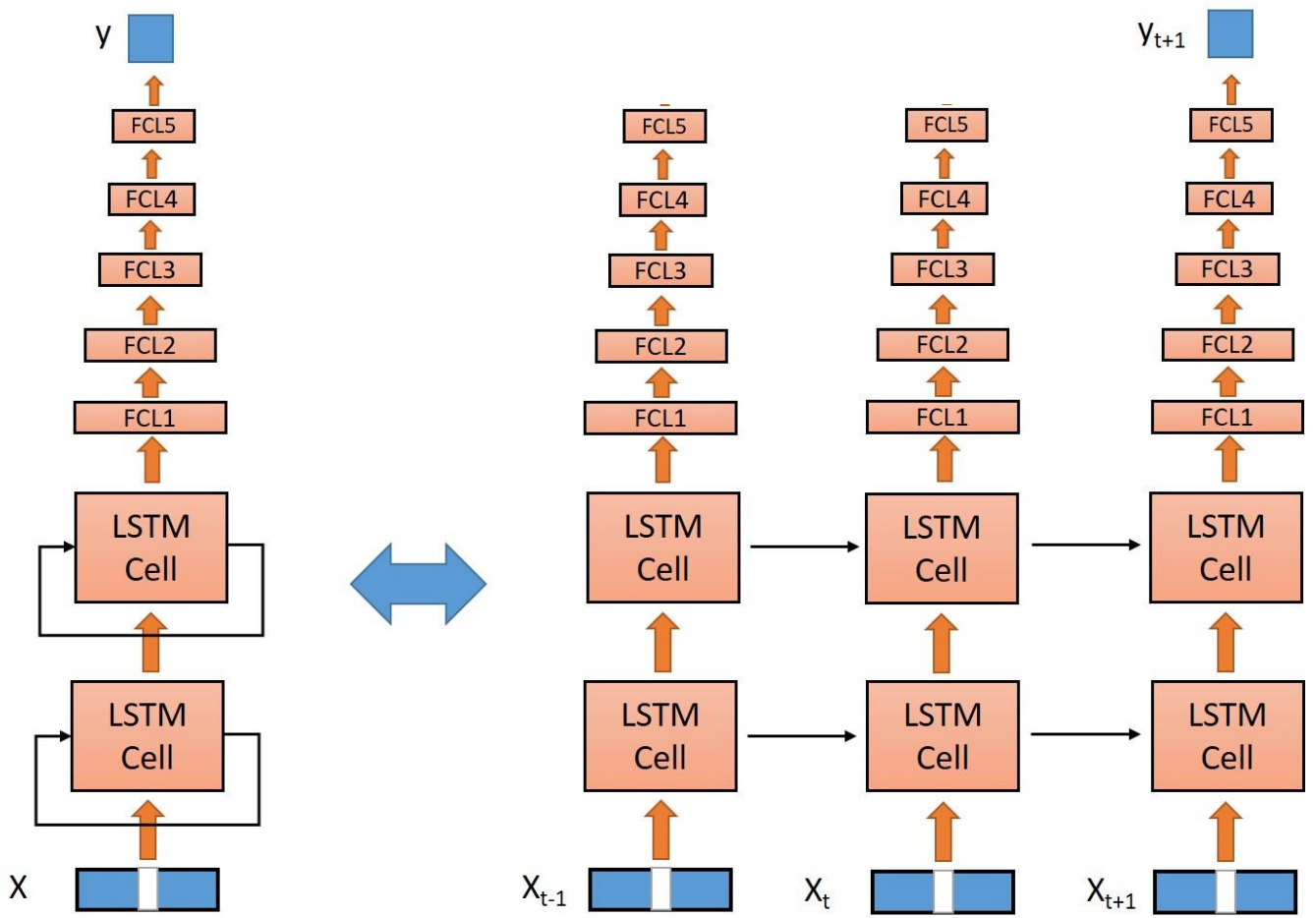
"Generalized Recurrent Unit"



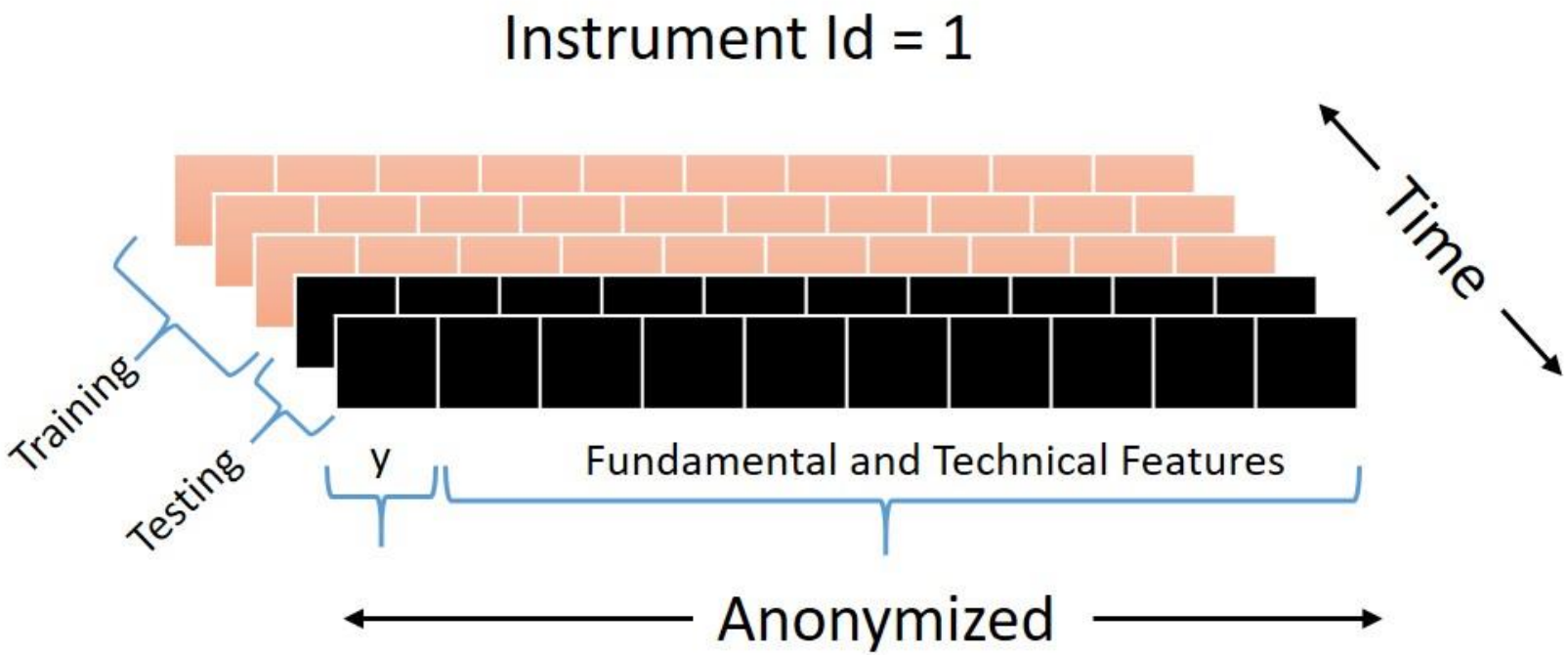
LSTM cell

"Lone Short Term Memory"

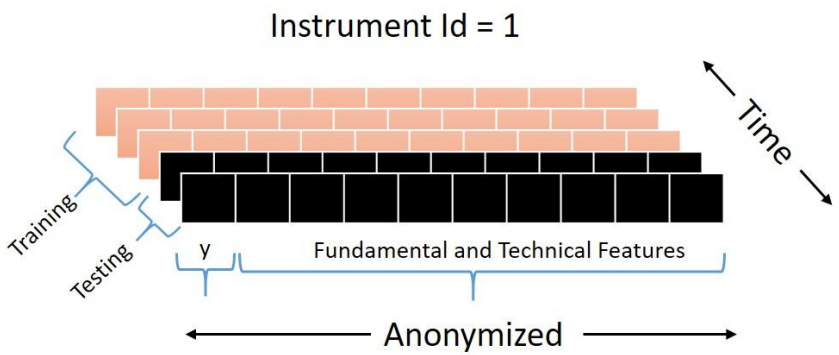
Model



Training and Testing Set

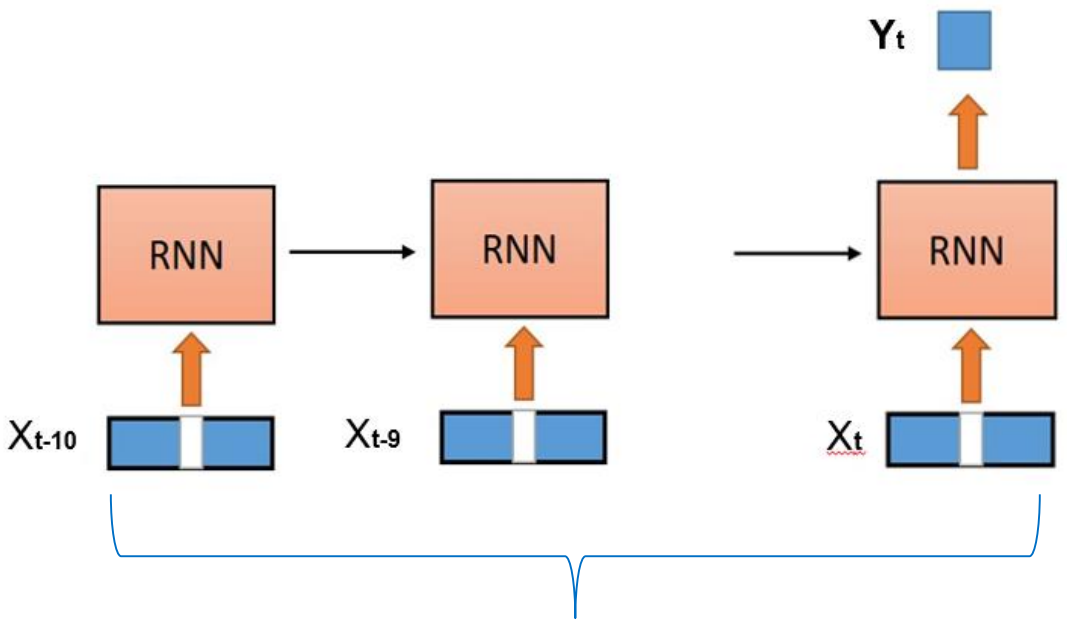


Training process



Tensor

(training_data_num,
n_time_steps = 10,
colum_num = 111)



n_time_steps = 10

LAB EVALUATION

Hyperparameters

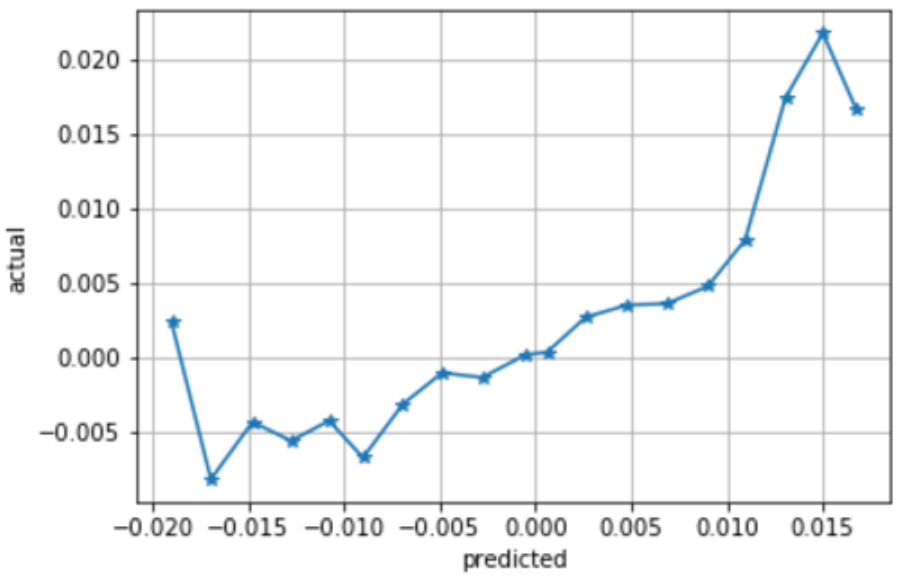
- Correlation should be around 0.045 when pre-trained model with 15 epochs is used.
- Correlation should improve to 0.048 (nearly) when pre-trained model with 30 epochs is used.

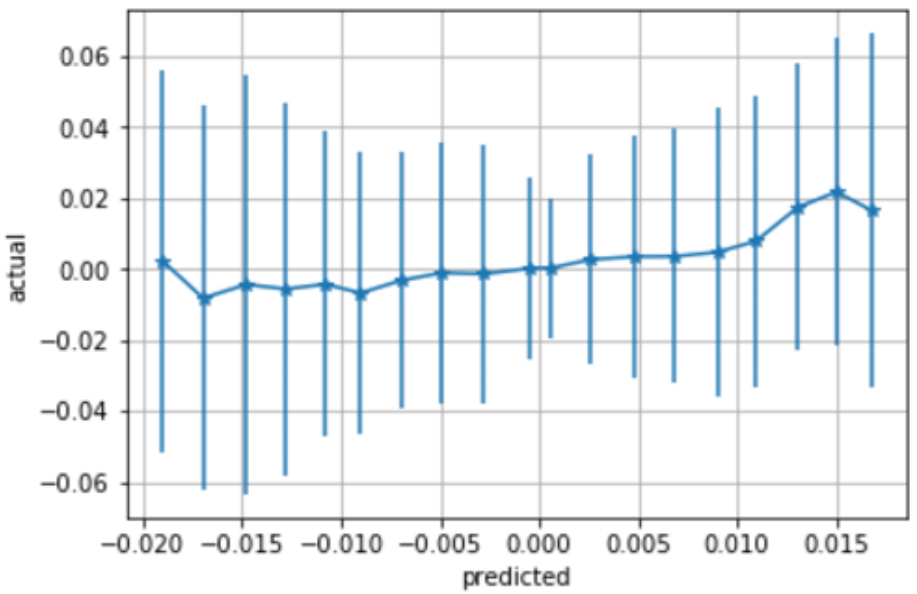
What are other Hyperparameters?

What are other Hyperparameters?

- Window size (10)
- Learning rate (0.00XX)
- Optimizer (Adam)
- Loss (l2 norm with regulized term)
- Mini_batch_limit (1300)
- Model (1 LSTM + 1 FC)

Predicted Signal





Next Steps

- We recommend you to try the following steps after the lab.
- Try using other machine learning techniques such as random forest, ridge regression, xgboost and compare the correlation with LSTM based predictor.
- Try using autoencoder to extract fewer features than the original dataset provides and use the features as input to the deep learning model. Analyze the performance.

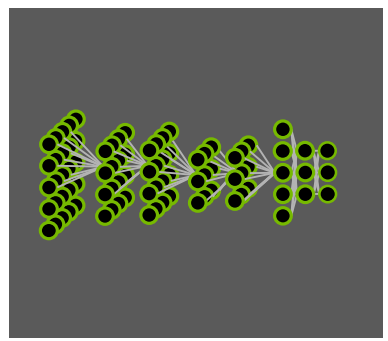
LSTM for Predictive Maintenance

PREDICTIVE MAINTENANCE

Information lies with time series vector

LEARNING ALGORITHM

"millions of trillions of FLOPS"



"Remaining useful life 3 cycles, part replacement recommended"

DATASET

Id / cycle / static / sensor

A. Saxena and K. Goebel (2008). "Turbofan Engine Degradation Simulation Data Set", NASA Ames Prognostics Data Repository (http://ti.arc.nasa.gov/tech/dash/pcoe/prognostic-data-repository/), NASA Ames Research Center, Moffett Field, CA

Index	Data fields	Type	Descriptions
1	id	Integer	aircraft engine identifier, range [1, 100]
2	cycle	Integer	time, in cycles
3	setting1	Double	operational setting 1
4	setting2	Double	operational setting 2
5	setting3	Double	operational setting 3
6	s1	Double	sensor measurement 1
7	s2	Double	sensor measurement 2
...	...		
26	s21	Double	sensor measurement 21

DATASET IN DETAILS

Static data

Sensor data

	id	cycle	setting1	setting2	setting3	s1	s2	s3	s4	s5	...	s16	s17	s18	s19	s20	s21
0	1	1	0.459770	0.166667	0.0	0.0	0.183735	0.406802	0.309757	0.0	...	0.0	0.333333	0.0	0.0	0.713178	0.724662
1	1	2	0.609195	0.250000	0.0	0.0	0.283133	0.453019	0.352633	0.0	...	0.0	0.333333	0.0	0.0	0.666667	0.731014
2	1	3	0.252874	0.750000	0.0	0.0	0.343373	0.369523	0.370527	0.0	...	0.0	0.166667	0.0	0.0	0.627907	0.621375
3	1	4	0.540230	0.500000	0.0	0.0	0.343373	0.256159	0.331195	0.0	...	0.0	0.333333	0.0	0.0	0.573643	0.662386
4	1	5	0.390805	0.333333	0.0	0.0	0.349398	0.257467	0.404625	0.0	...	0.0	0.416667	0.0	0.0	0.589147	0.704502

⋮

	id	cycle	setting1	setting2	setting3	s1	s2	s3	s4	s5	...	s16	s17	s18	s19	s20	s21
20626	100	196	0.477011	0.250000	0.0	0.0	0.686747	0.587312	0.782917	0.0	...	0.0	0.750000	0.0	0.0	0.271318	0.109500
20627	100	197	0.408046	0.083333	0.0	0.0	0.701807	0.729453	0.866475	0.0	...	0.0	0.583333	0.0	0.0	0.124031	0.366197
20628	100	198	0.522989	0.500000	0.0	0.0	0.665663	0.684979	0.775321	0.0	...	0.0	0.833333	0.0	0.0	0.232558	0.053991
20629	100	199	0.436782	0.750000	0.0	0.0	0.608434	0.746021	0.747468	0.0	...	0.0	0.583333	0.0	0.0	0.116279	0.234466
20630	100	200	0.316092	0.083333	0.0	0.0	0.795181	0.639634	0.842167	0.0	...	0.0	0.666667	0.0	0.0	0.178295	0.218172

WHAT TO PREDICT

Go back to domain expert!

- **Regression**
  - Predict the Remaining Useful Life (RUL), or Time to Failure (TTF).
  -
- **Binary classification**
  - Predict if an asset will fail within certain time frame.

- **Multi-class classification**
  - Predict if an asset will fail in different time windows or
  - Predict different malfunction mode (if you believe that pattern do exists within input!!)

LABEL

What to predict?

Label

	id	cycle	setting1	setting2	setting3	s1	s2	s3	s4	s5	...	s15	s16	s17	s18	s19	s20	s21	RUL	label1	label2
0	1	1	-0.0007	-0.0004	100.0	518.67	641.82	1589.70	1400.60	14.62	...	8.4195	0.03	392	2388	100.0	39.06	23.4190	191	0	0
1	1	2	0.0019	-0.0003	100.0	518.67	642.15	1591.82	1403.14	14.62	...	8.4318	0.03	392	2388	100.0	39.00	23.4236	190	0	0
2	1	3	-0.0043	0.0003	100.0	518.67	642.35	1587.99	1404.20	14.62	...	8.4178	0.03	390	2388	100.0	38.95	23.3442	189	0	0
3	1	4	0.0007	0.0000	100.0	518.67	642.35	1582.79	1401.87	14.62	...	8.3682	0.03	392	2388	100.0	38.88	23.3739	188	0	0
4	1	5	-0.0019	-0.0002	100.0	518.67	642.37	1582.85	1406.22	14.62	...	8.4294	0.03	393	2388	100.0	38.90	23.4044	187	0	0

⋮

	id	cycle	setting1	setting2	setting3	s1	s2	s3	s4	s5	...	s15	s16	s17	s18	s19	s20	s21	RUL	label1	label2
20626	100	196	-0.0004	-0.0003	100.0	518.67	643.49	1597.98	1428.63	14.62	...	8.4956	0.03	397	2388	100.0	38.49	22.9735	4	1	2
20627	100	197	-0.0016	-0.0005	100.0	518.67	643.54	1604.50	1433.58	14.62	...	8.5139	0.03	395	2388	100.0	38.30	23.1594	3	1	2
20628	100	198	0.0004	0.0000	100.0	518.67	643.42	1602.46	1428.18	14.62	...	8.5646	0.03	398	2388	100.0	38.44	22.9333	2	1	2
20629	100	199	-0.0011	0.0003	100.0	518.67	643.23	1605.26	1426.53	14.62	...	8.5389	0.03	395	2388	100.0	38.29	23.0640	1	1	2
20630	100	200	-0.0032	-0.0005	100.0	518.67	643.85	1600.38	1432.14	14.62	...	8.5036	0.03	396	2388	100.0	38.37	23.0522	0	1	2

深度學習實作坊問券

請掃描右方QR code填寫問券

保留填寫完成畫面於離場
前換取精美禮物

Instructor: Andrew Liu, Ph.D.

www.nvidia.com/dli