Time Series Analysis
Thematic Assignment
(2023 - 2024)

Team 24

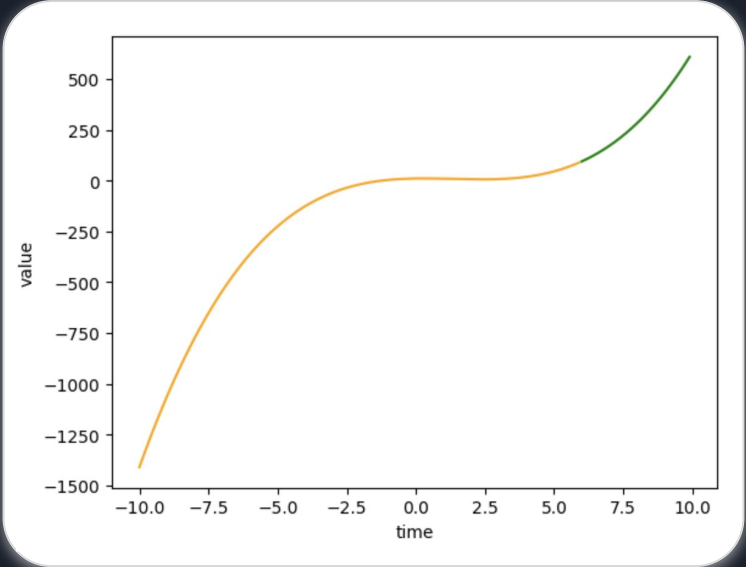
Kyparissis Kyparissis	10346	kyparkypar@ece.auth.gr
Fotios Alexandridis	9953	faalexandr@ece.auth.gr

The problem:

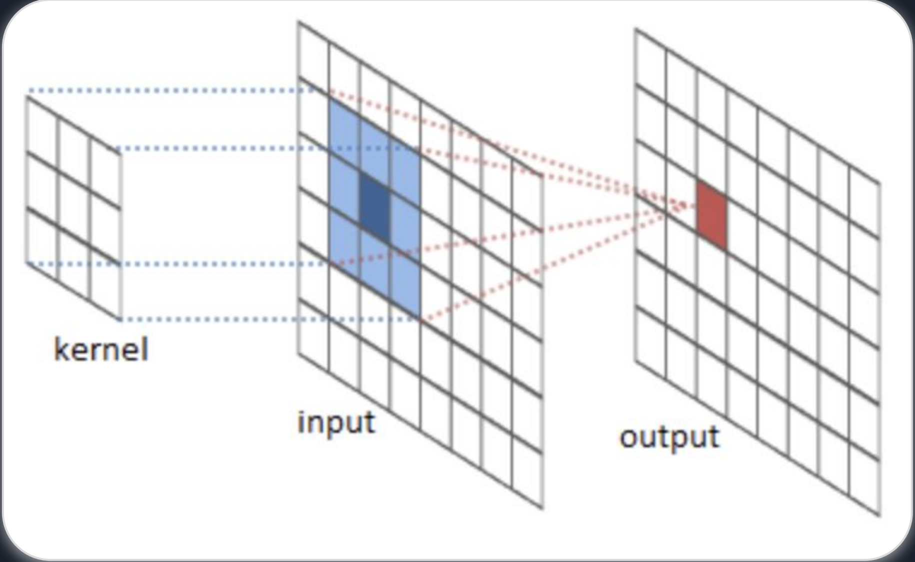
Time series forecasting using CNNs

Time series forecasting using CNNs

Predicting the future based on the past



Neural Networks based on **discrete convolutions**



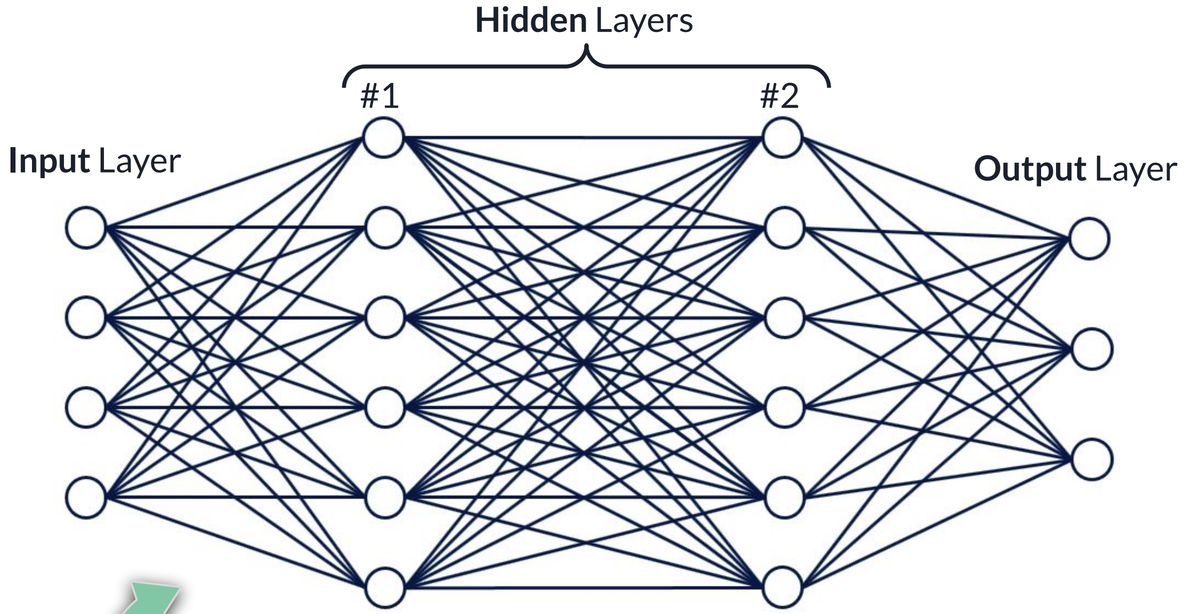
What are Neural Networks

- Also known as **Artificial Neural Networks** (ANNs)
- Subset of Machine Learning and heart of Deep Learning algorithms
  - Rely on **training data to learn** and improve their accuracy over time
  - Classify and cluster data at a high velocity

- Name and structure inspired by the human brain
  - Mimicking the way that biological neurons signal to one another

- Utilize the concept of **(node) layers**
  - Represent structured **computations** / data **transformations**
  - Containing:
    - → **1 INPUT** layer
    - → ≥ 1 **HIDDEN** layers
    - → **1 OUTPUT** layer
  - Each node / artificial neuron connects to another
    - With an associated **weight** and **threshold**
      - ➢ If the output of any individual node **>** specified threshold value
        - Node is activated
        - Sends data to the next layer of the network
      - ➢ Otherwise, no data is passed along to the next layer of the network

What are Neural Networks



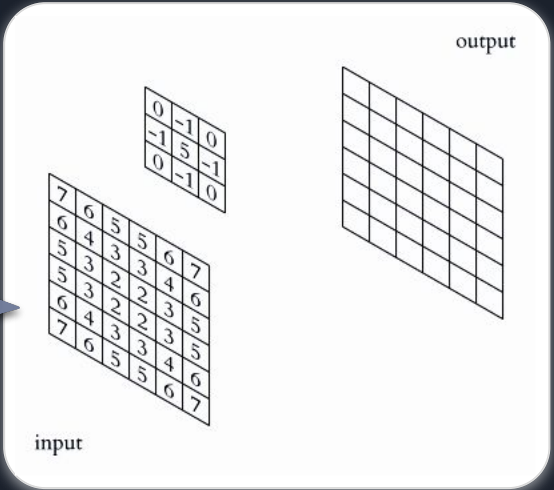
Fully-Connected (FC) Layers

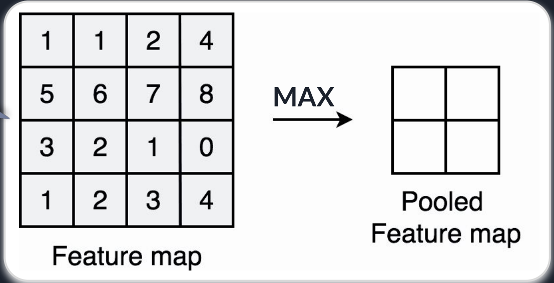
- Applies a **linear transformation** to the input vector through a weight matrix
  - Input vector (x) · Weights matrix (W)
  - All possible connections layer-to-layer are present

What are CNNs

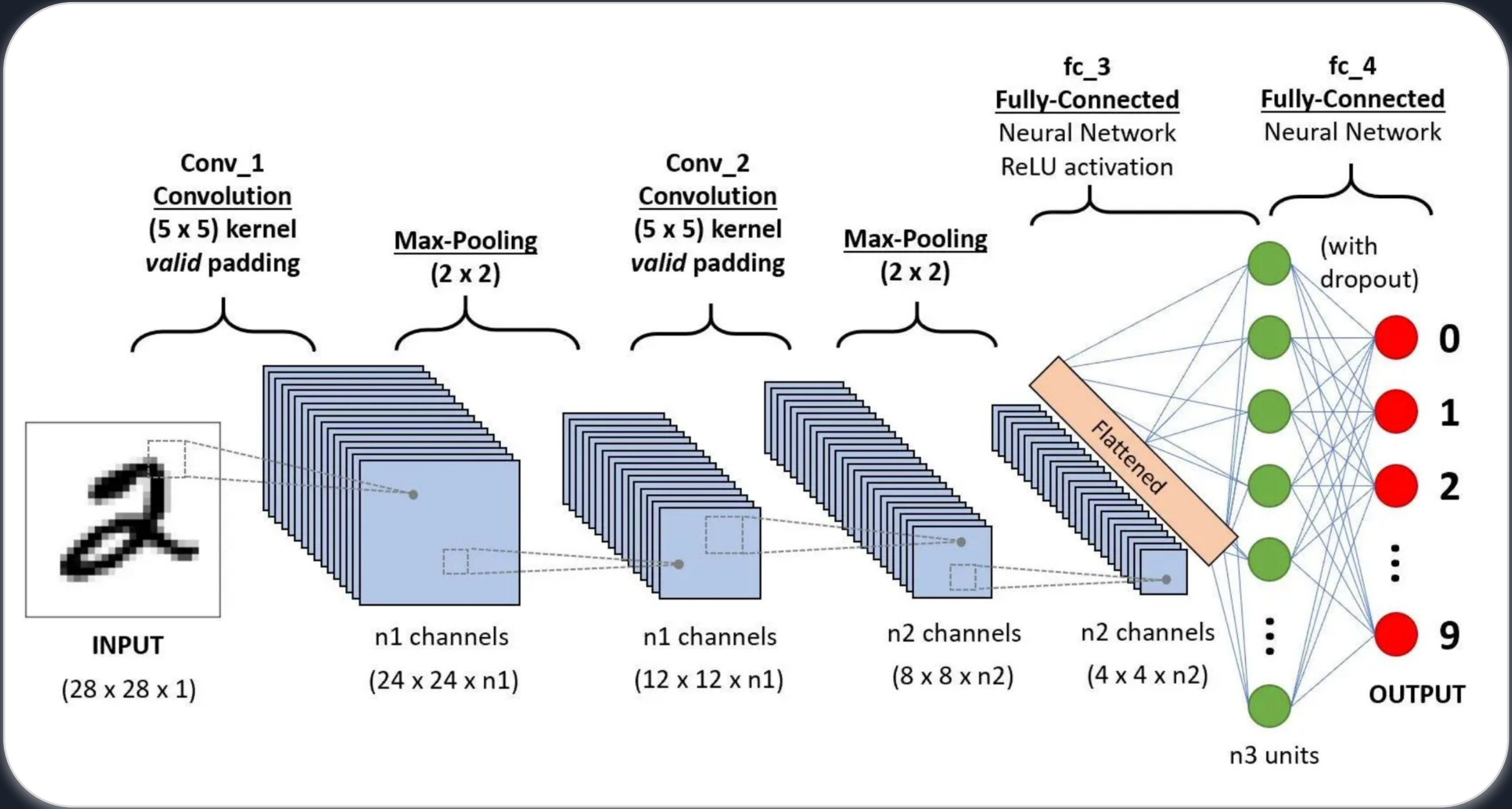
Convolutional Neural Networks (CNNs) are Artificial Neural Networks **based mainly on Convolutional layers**

- They have **3 main types of layers**, which are:
    - **Convolutional layer**
        - Based on **discrete convolutions**
        - Moves around a filter called a kernel
            - Takes the sum of the weighted multiplication
            - Produce a feature map
                - Captures local patterns and spatial dependencies in different parts of the input
        - Can have multiple filters in one layer
    - **Pooling layer**
        - Reduce dimensionality by downsampling
        - Move around a filter
            - Keep max / average value
    - **Fully-Connected (FC) layer**
        - Responsible for the final classification task
- CNNs are used extensively in **Computer Vision** problems
    - Image (or video data) classification problems





What are CNNs



Benefits of CNNs for time series forecasting

Local Pattern Extraction

- CNNs are effective in **capturing local patterns**
    - Crucial for understanding short-term dependencies in time series data

Parameter Sharing

- Sharing parameters across different regions of the sequence
- Allows the network to **generalize** well

Translation Invariance

- CNNs can be translationally invariant
    - Can recognize patterns **regardless of their position** in the sequence

Benefits of CNNs for time series forecasting

Ability to Capture Spatial Dependencies

- Convolutional operations are primarily designed for spatial data

HOWEVER

- Can effectively capture spatial dependencies in the temporal domain
  - Aiding in understanding how different time steps relate to each other

Applicability to Multivariate Time Series

- CNNs can be adapted for multivariate time series forecasting
  - By extending the convolutional operations across different dimensions
  - Providing a versatile framework

Past studies & Examples

Conditional time series forecasting with convolutional neural networks - 2018

This study supports:

- Dilated convolutions can be used to learn time series:
  - Long term **correlations**
  - **Dependencies**
- This is **<u>important</u>** since in many environments (e.g. economy)
  - Long term patterns often appear in data
  - Understanding them is crucial to analyze and predict future values

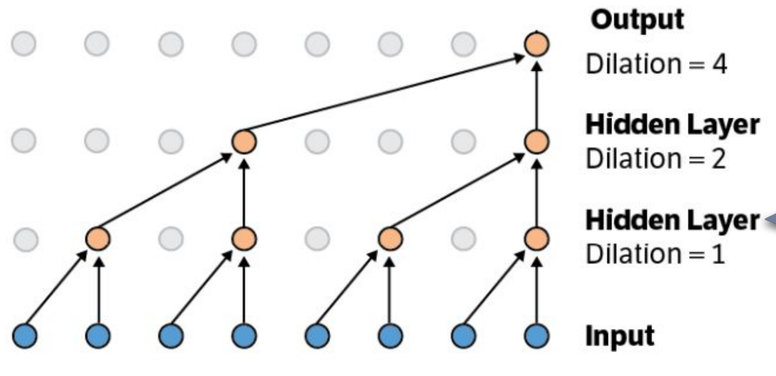


Figure 2.2: A dilated convolutional neural network with three layers.

Loss Function:
Mean Absolute Error

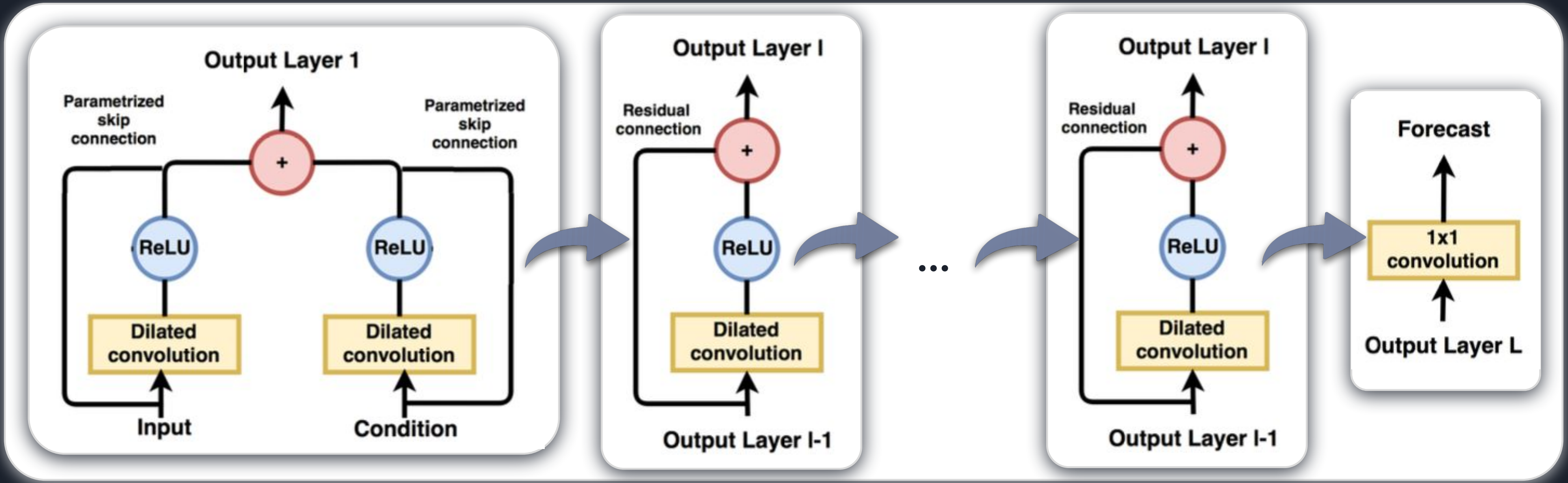
Optimization Algorithm:
Gradient Descent

Activation Function:
Rectified Linear Unit (ReLU)

Conditional time series forecasting with convolutional neural networks - 2018

Another very important point → **CONDITIONING**

- Able to forecast future values of time series A
  - Knowing **past values** of both time series A and another, related, time series B
- Can be extended to *M* related time series
  - Extremely useful for multivariate time series.

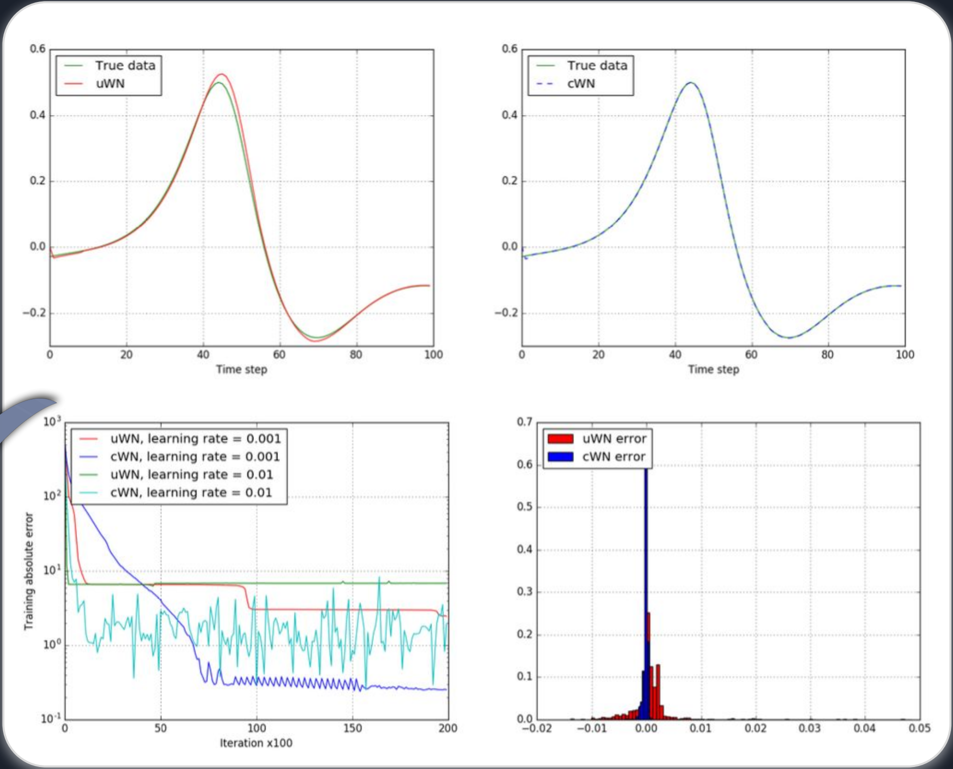


Conditional time series forecasting with convolutional neural networks - 2018 - Example 1

Example:

Lorenz Map:
$$\begin{cases}\dot{\mathbf{X}} = \sigma(\mathbf{Y} - \mathbf{X}) \\ \dot{\mathbf{Y}} = \mathbf{X}(\rho - \mathbf{Z}) - \mathbf{Y} \\ \dot{\mathbf{Z}} = \mathbf{XY} - \beta\mathbf{Y}\end{cases}$$

- For the generated time series **X**
  - **Unconditional forecasting** (uWN Model)
  - **Conditional forecasting** (cWN Model)



The **conditional model (cWN)** **performs really well** in estimating future members of the time series

Conditional time series forecasting with convolutional neural networks - 2018 - Example 2

Models were used in financial data forecasting for <u>different stocks</u>:

- **SP500 stock**
  - Trained for different time periods
  - Results are obtained for multiple models
    - **Conditional (cWN) and the unconditional (uCW) models included!**

	A		B		C	
Model	MASE	HITS	MASE	HITS	MASE	HITS
Naive	1	0.513	1	**0.504**	1	**0.555**
VAR	**0.698**	0.507	0.701	**0.505**	0.696	0.551
LSTM	0.873(0.026)	**0.525(0.006)**	1.067(0.021)	0.496(0.016)	0.929(0.021)	0.531(0.008)
uWN	**0.685(0.025)**	0.515(0.007)	**0.681(0.002)**	0.484(0.007)	**0.684(0.006)**	0.537(0.011)
cWN	0.699(0.042)	**0.524(0.009)**	0.693(0.014)	**0.500(0.009)**	0.701(0.015)	0.536(0.016)

Table 3.2: MASE and HITS (mean(standard deviation)) for a one-step ahead forecast over the periods A, B and C of the S&P500, both unconditional and conditional on the volatility index and the CBOE 10 year interest rate.

The new architectures **(uWN & cWN)** dominate over most metrics & time periods!

Conditional time series forecasting with convolutional neural networks - 2018 - Example 2

- **FOREX data**
  - Global economy is influenced in similar ways

Model	Period	EURUSD	EURJPY	GBPJPY	EURGBP	GBPUSD
VAR	A	1.105	1.176	1.446	1.348	1.832
	B	0.758	0.782	0.756	0.768	0.731
	C	0.716	0.738	0.737	**0.709**	**0.713**
LSTM	A	0.829(0.012)	0.863(0.005)	0.880(0.004)	0.868(0.005)	**0.893(0.007)**
	B	0.925(0.024)	0.911(0.029)	0.974(0.029)	0.948(0.023)	0.934(0.014)
	C	0.950(0.016)	1.031(0.022)	0.980(0.034)	0.839(0.034)	0.898(0.017)
cWN	A	**0.693(0.016)**	**0.667(0.021)**	**0.759(0.064)**	**0.728(0.014)**	**0.834(0.089)**
	B	**0.690(0.006)**	**0.693(0.006)**	**0.699(0.005)**	**0.717(0.015)**	**0.710(0.009)**
	C	**0.702(0.009)**	**0.716(0.029)**	**0.721(0.014)**	**0.709(0.004)**	**0.716(0.004)**

Table 3.4: MASE (mean(standard deviation)) one-step ahead multivariate forecast over the periods A, B and C of five foreign exhange rates.

The **conditional model (cWN)** dominated the charts

Expected since:

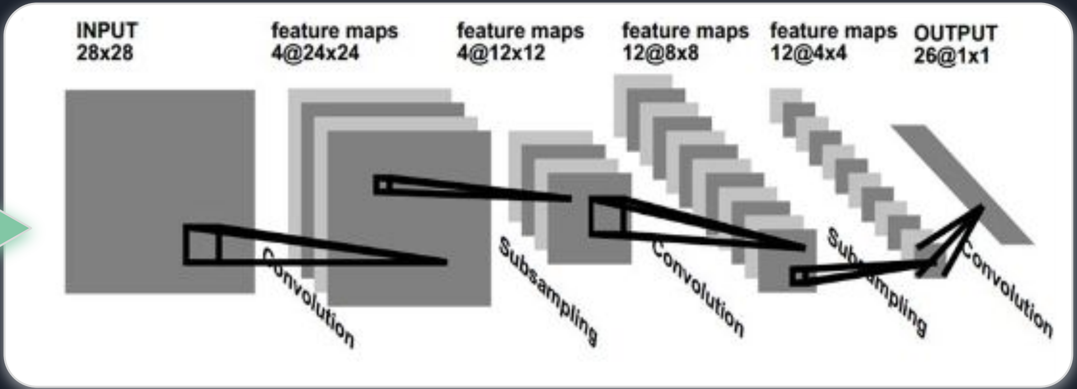
- Changes in one currency trading pair (EURUSD) often influence other currencies as well (EURJPY)

The conditional model further backs the assumptions of exceptional performance in multivariate forecast

Financial Time-series Data Analysis using Deep Convolutional Neural Networks - 2016

This study supports:

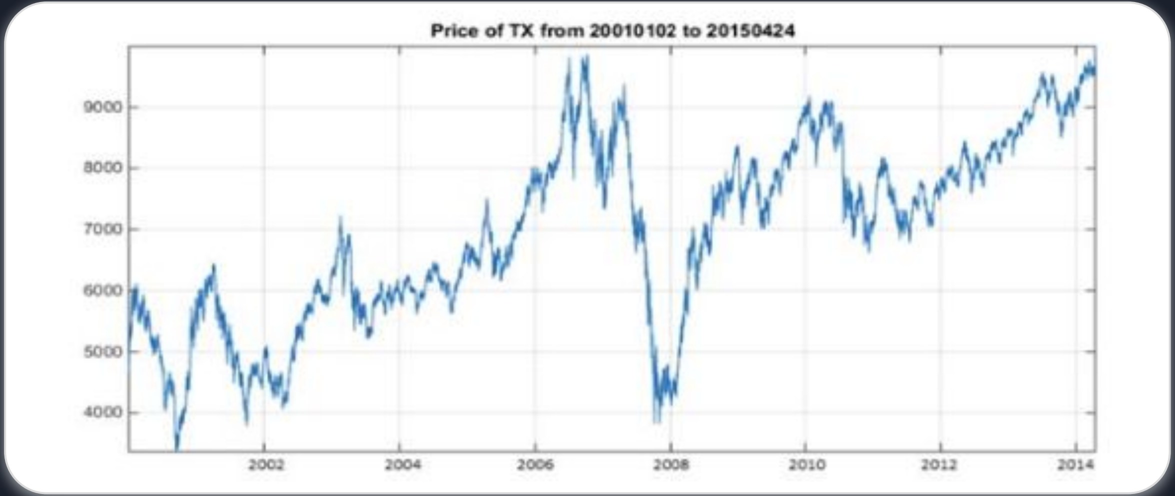
- **CNNs** have shown to be **state of the art** in problems that have **images** as data
- Main concept explored: **Transformation of 1D time series → 2D picture**
- Novel CNN architecture is used:
  - 2 Convolution layers
  - 2 Pooling layers
  - 1 final output layer



Financial Time-series Data Analysis using Deep Convolutional Neural Networks - 2016 - Example

The problem consists of **forecasting** the future price of Taiwan index futures

- Training dataset consists of ~1M historical intraday 1' data
- From 2/1/2001 to 24/4/2015
- Data are given in **OHLC** prices for each data point
    - OHLC = Open High Low Close
    - We have the Opening price, Closing price, Highest price and Lowest price for that specific 1' time window



Financial Time-series Data Analysis using Deep Convolutional Neural Networks - 2016 - Example

4 methods were used to convert the 1D time series → 2D image:

1. **Moving Average Mapping (MAM)**

$$G = \begin{bmatrix} MA_{t-D+1,1} & \cdots & MA_{t,1} \\ \vdots & \ddots & \vdots \\ MA_{t-D+1,D} & \cdots & MA_{t,D} \end{bmatrix}$$

2. **Double Moving Average Mapping (DMAM)**
   - Defined in the same way that the Moving Average Mapping

     BUT

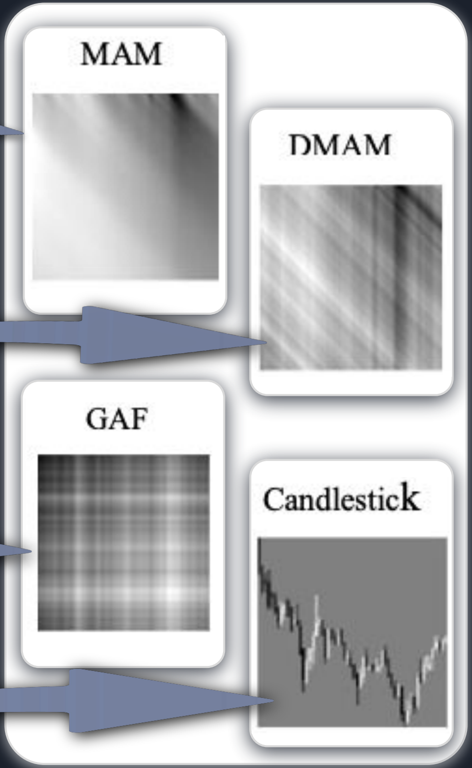
   - Uses the mean value of the opening price and closing price
     - Instead of simply the closing price (Moving Average Mapping case)

3. **Gramian Angular Field (GAF)**

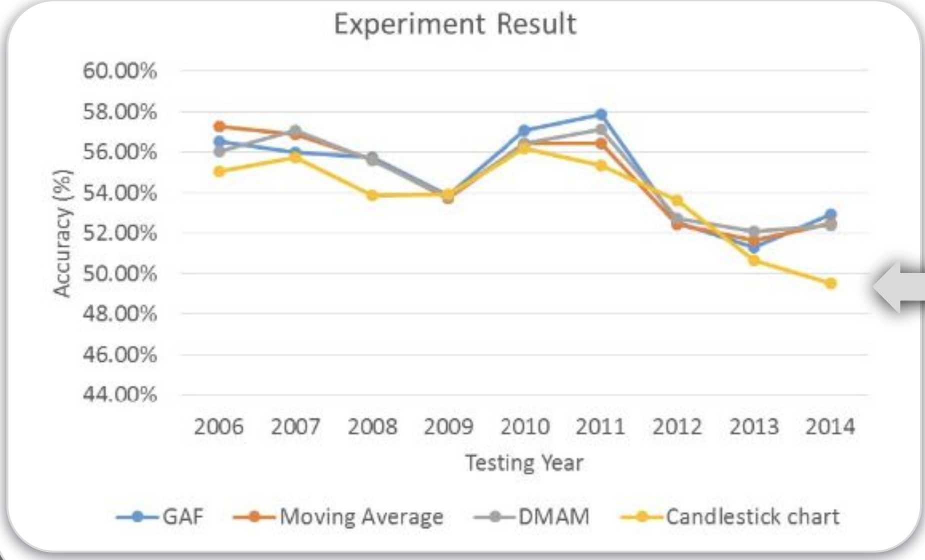
$$G = \begin{bmatrix} <\tilde{x}_1, \tilde{x}_1> & \cdots & <\tilde{x}_1, \tilde{x}_n> \\ \vdots & \ddots & \vdots \\ <\tilde{x}_n, \tilde{x}_1> & \cdots & <\tilde{x}_n, \tilde{x}_n> \end{bmatrix}$$

where $< x, y > = x \cdot y - \sqrt{1 - x^2} \cdot \sqrt{1 - y^2}$

4. **Image of candlestick chart**
   - More traditional graph
   - Often seen in trading views of brokers and exchanges



Financial Time-series Data Analysis using Deep Convolutional Neural Networks - 2016 - Example



Testing Year	GAF	MAM	DMAM	Candlestick chart
2006	56.53%	57.28%	56.05%	55.04%
2007	55.97%	56.86%	57.06%	55.72%
2008	55.73%	55.62%	55.57%	53.87%
2009	53.84%	53.71%	53.81%	53.91%
2010	57.06%	56.42%	56.42%	56.20%
2011	**57.88%**	56.42%	57.10%	55.36%
2012	52.52%	52.44%	52.75%	53.60%
2013	51.28%	51.63%	52.07%	50.66%
2014	52.93%	52.47%	52.36%	49.50%
Average	54.86%	54.76%	54.80%	53.76%

Results showcase that in **almost all cases** the ranking for the representation is as follows:

GAF > DMAM > MAM > Chandlestick

Comparisons with
other NNs and nonlinear
models

CNNs **vs** Fully Connected Neural Networks

Pros of **CNNs**

- Able to **learn patterns** regardless of their position in the time series
  - Have temporal invariance
- Run a smaller risk of **overfitting**
- Often **less computationally intensive**
  - Fully connected networks need to be sufficiently large to process a large time frame of the timeseries

Pros of **Fully Connected Neural Networks**

- Understand the full **global context** of data
  - Process the entirety of the data
- Often have a **simpler, layered structure**
- Can handle **variable length sequences**
  - No major reshapings required

CNNs **vs** Recurrent Neural Networks

Pros of **CNNs**

- Often **prevent overfitting** in problems with **limited training data**
- Usually **less computationally intensive** to train and run
- Handle local **patterns** and spatial relationships **better**

Pros of **Recurrent Neural Networks**

- Very good in handling temporal order of events
- Can generally handle long term trends and relationships very effectively
- Can easily handle variable length input
  - With minimal changes

CNNs **vs** Random Forest

Pros of **CNNs**

- Way **less computationally intensive**
- Able to **learn larger patterns** that may be **more apart** from each other
- Able to **learn sequential data** more efficiently

Pros of **Random Forest**

- Ensemble method that can **capture** complex, non-linear **relationships**
- They provide **feature importance** rankings
  - Aiding in understanding the impact of different variables on the forecast
- They are **robust** in regards to **outliers**

[1] Borovykh, Anastasia, Sander Bohte, and Cornelis W. Oosterlee. "**Conditional time series forecasting with convolutional neural networks.**" arXiv preprint arXiv:1703.04691 (2017)

[2] J. -F. Chen, W. -L. Chen, C. -P. Huang, S. -H. Huang and A. -P. Chen, "**Financial Time-Series Data Analysis Using Deep Convolutional Neural Networks,**" 2016 7th International Conference on Cloud Computing and Big Data (CCBD), Macau, China, 2016, pp. 87-92, doi: 10.1109/CCBD.2016.027.

[3] "**What Are Neural Networks?**" IBM, www.ibm.com/topics/neural-networks. Accessed 11 Jan. 2024.

[4] "**What are convolutional neural networks?**" IBM, https://www.ibm.com/topics/convolutional-neural-networks. Accessed 11 Jan. 2024.

REFERENCES

Thank You!

For Your Attention

Any questions?