Data augmentation techniques

An overview for time series analysis.

Gian Luca Vriz

June 15, 2023

Image generation

Data augmentation (DA) was first introduced in the field of computer vision, specifically in image processing and analysis.

Origin

- **Data augmentation** is a well-known method for image recognition. When distinguishing between cats and dogs, image recognition software must contend with a range of challenges, including perspective, lighting, obstructions, background, size, and more.
- **Assumption**: more information can be extracted from the original dataset through augmentations.
- **Early experiments** that established the efficacy of DA highlighted basic modifications like flipping images horizontally, altering colour space, and randomly cropping images.

Origin

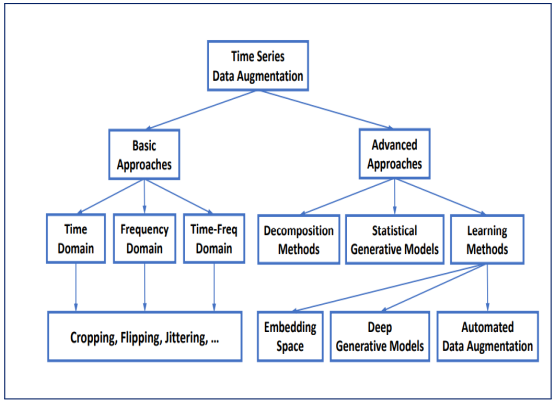
- **More complex**: most of the state-of-the-art Convolutional Neural Network (CNN) architectures used some form of data augmentation.
- The **AlexNet CNN architecture** developed by Krizhevsky, Sutskever, and Hinton (2017) revolutionized image classification by applying convolutional networks to the ImageNet dataset. Data Augmentation is used in their experiments to increase the dataset size. This approach reduced overfitting and the error rate of the model by over 1%.
- **Main massage**: DA is similar to imagination or dreaming. Just as humans imagine various situations built on their experiences, imagination assists us in comprehending our world more effectively.

Time series

“An economist is an expert who will know tomorrow why the things he predicted yesterday didn’t happen today.”
Evan Esar.

DA for time series

- **Basic approaches**: transformations in the time/frequency domain are the first proposed data augmentation methods for time series.
- **Advanced approaches**: decomposition-based time series augmentation was applied demonstrating positive results.
- **Advanced approaches 2.0**: sampling time series from feature distributions using generative models ➔ statistical and neural network-based methods.



Basic approaches: time domain

- **Jittering**: adding noise to time series.
- **Flipping**: rotating the time series or only specific time blocks.
- **Scaling**: changing the magnitude of a time series by a random scalar value.
- **Magnitude warping**: changing the magnitude of a time series by a smoothed curve.
- **Permutation**: rearranging segments of a time series to produce new patterns.

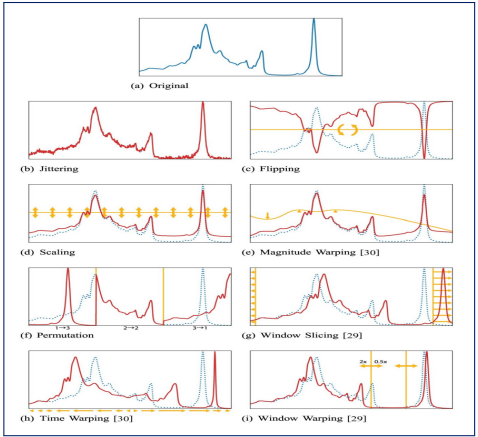


Figure: Random transformation-based methods.

Basic approaches: time domain

- **Slicing**: data augmentation technique equivalent to cropping for image data augmentation.
- **Time warping**: perturbing the pattern of a time series in the temporal dimension using a smooth warping path.
- **Window warping**: perturbing the pattern of a time series in the temporal dimension using a randomly located fixed window.

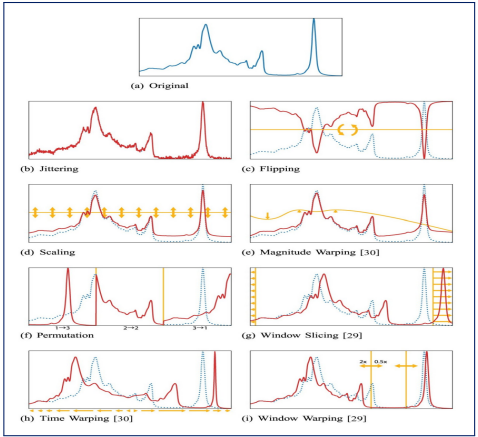


Figure: Random transformation-based methods.

Basic approaches: frequency domain

- **Frequency warping**: method of data augmentation used in audio and speech recognition (Jaitly and Hinton 2013). The idea is to simulate changes in the underlying dynamics of the time series distorting the frequency content (periodicity and oscillations).
- **Fourier transform-based method**[1]: perturbations in both amplitude spectrum and phase spectrum of the frequency domain (Gao et al. 2020).
- **Time-frequency domain**: all methods which combine two or more time/frequency techniques.

[1] Fourier transformation is a mathematical technique used to transform a time-domain signal into its frequency-domain representation.

Adv. approaches: decomposition methods

- The idea of **decomposition methods** is to separate time series' signals by extracting features or underlying patterns (Bergmeir, Hyndman, and Benitez 2016).
- The most common decomposition method is the **Seasonal and Trend decomposition using Loess** (STL).
- New time series are created with a **deterministic component** (trend and seasonality) and a **stochastic component** (residuals).

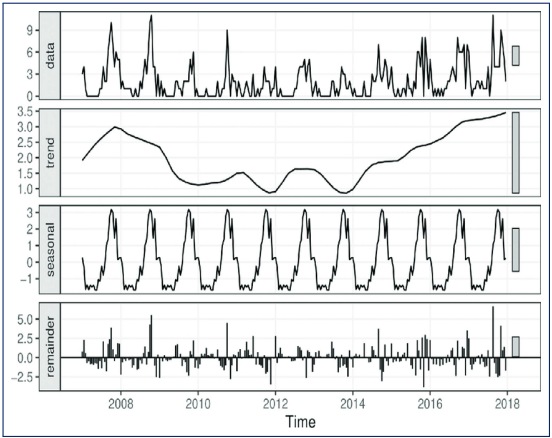


Figure: STL decomposition.

Adv. approaches 2.0: statistical models

- Approaches based on **statistical generative models** simulate the dynamics of the time series with a model.
- **Assumption**: describing the conditional distribution of the time series by assuming that the value at time t depends on previous points.
- Once the initial value is chosen, a **synthetic time series** could be generated by the model.
- **Local and Global Trend** (LGT) uses nonlinear global trends and reduced local linear trends to model the data (Smyl and Kuber 2016).
- **GeneRAting TIme Series** (GRATIS) uses a Mixture AutoRegressive (MAR) model in order to simulate new time series (Kang, Hyndman, and Li 2020).

Adv. approaches 2.0: learning-based methods

- **Idea**: Time series data augmentation methods should be able to mimic the underline features of real data and not only generate new samples.
- **Encoder-decoder networks**: taking a structural input, encoding it into a latent space, and then decoding it back to a high-dimensional or structural output. Data augmentation methods generate new patterns by decoding vectors sampled from the latent space (Tu et al. 2018).
- **Deep Generative Models** (DGM): Generative Adversarial Networks (GANs) are well-established machine learning methods to generate synthetic samples and increase the training set. Yoon, Jarrett, and Van der Schaar (2019) proposed TimeGAN, a novel framework for generating realistic time series data in various domains.

GANs

- Typically, GAN consists of two neural networks - a **generator** and a **discriminator** - that works against each other in a two-player **min-max game** to generate new data that is similar to the training data.
- The generator tries to **create** new data samples that are indistinguishable from the real data.
- The discriminator tries to correctly **identify** which samples are real and which ones are fake.

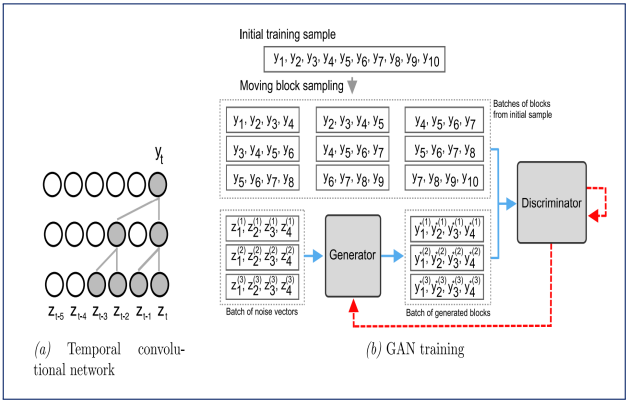


Figure: GANs archetype (Dahl and Sørensen 2022).

Example: time series

Figure: Real vs synthetic data, log returns.

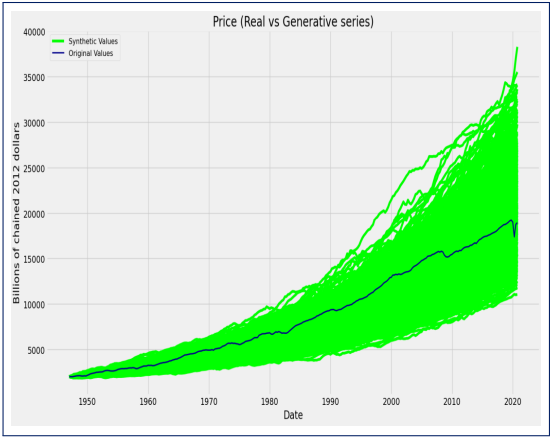


Figure: Real vs synthetic data, original series.

Example: error distribution

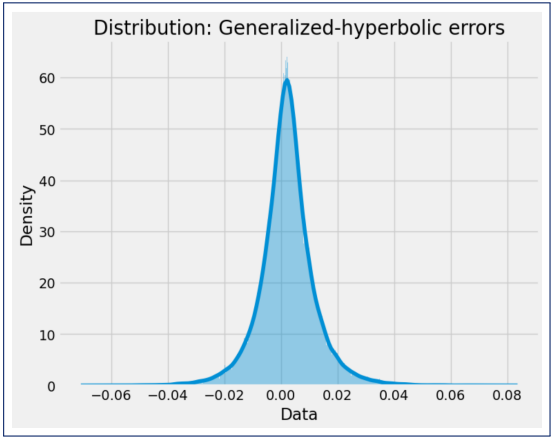


Figure: Original unknown distribution 128,000 sampels.

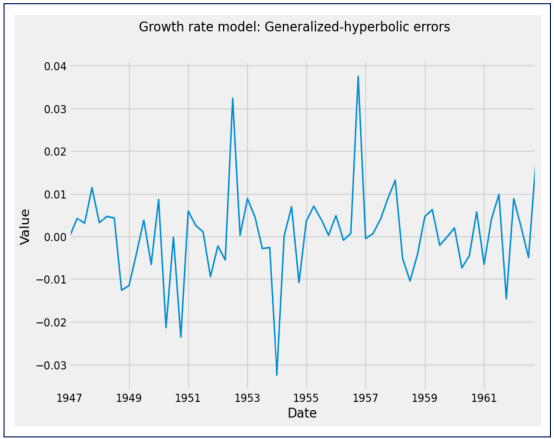


Figure: Time series.

Example: error distribution

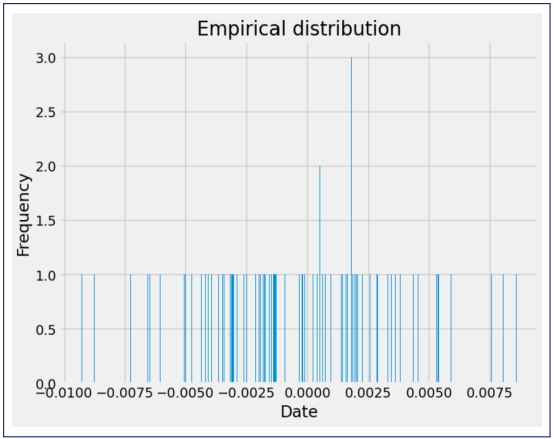


Figure: Empirical distribution.

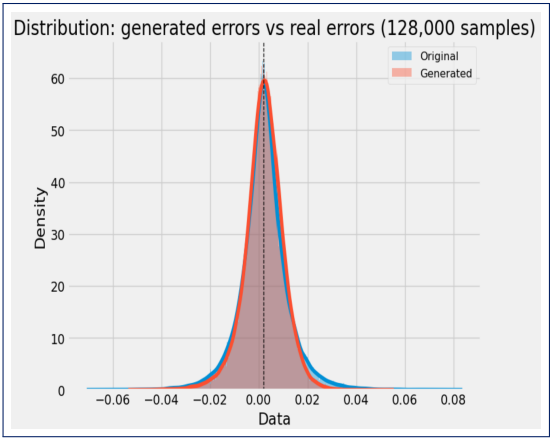


Figure: Real vs synthetic data distribution.

Final remarks

Augmented reality will take time to get right, but it will be one of the biggest technological revolutions of our life.

Pros and cons

- **Time/frequency domain transformations** offer a good balance between simplicity and credibility. However, if the underlying structure of the time series is not properly learned, synthetic data may be corrupted by additional noise.
- **Decomposing** a time series into trend, seasonality, and noise components can produce augmented data that faithfully reflects the original patterns. Yet, complex or irregular time series may not be effectively analyzed using such methods.
- **Statistical generative models and learning-based methods** can capture complex time series features. Nevertheless, these models can be computationally costly and unstable due to hyperparameters and training configurations.

Pros and cons

Approach	Advantages	Disadvantages
Time/frequency domain transformation	Simple and no computationally expensive	Poor data quality
Decomposition methods	Data augmentation based on underlying time series patterns	Poor data quality in complex time series
Statistical generative models and learning-based methods	High data quality	Computationally expensive and unstable

Table: Summary table.

- **Future directions** ➔ there are other data augmentation techniques used in the image domain, but not in time series

References

- Bergmeir, Christoph, Rob J Hyndman, and José M Benitez (2016). “Bagging exponential smoothing methods using STL decomposition and Box–Cox transformation”. In: *International journal of forecasting* 32.2, pp. 303–312.
- Dahl, Christian M and Emil N Sørensen (2022). “Time series (re) sampling using generative adversarial networks”. In: *Neural Networks* 156, pp. 95–107.
- Gao, Jingkun et al. (2020). “Robusttad: Robust time series anomaly detection via decomposition and convolutional neural networks”. In: *arXiv preprint arXiv:2002.09545*.
- Jaitly, Navdeep and Geoffrey E Hinton (2013). “Vocal tract length perturbation (VTLP) improves speech recognition”. In: *Proc. ICML Workshop on Deep Learning for Audio, Speech and Language*. Vol. 117, p. 21.

References

- Kang, Yanfei, Rob J Hyndman, and Feng Li (2020). “GRATIS: GeneRAting TIme Series with diverse and controllable characteristics”. In: *Statistical Analysis and Data Mining: The ASA Data Science Journal* 13.4, pp. 354–376.
- Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton (2017). “Imagenet classification with deep convolutional neural networks”. In: *Communications of the ACM* 60.6, pp. 84–90.
- Smyl, Slawek and Karthik Kuber (2016). “Data preprocessing and augmentation for multiple short time series forecasting with recurrent neural networks”. In: *36th international symposium on forecasting*.
- Tu, Juanhui et al. (2018). “Spatial-temporal data augmentation based on LSTM autoencoder network for skeleton-based human action recognition”. In: *2018 25th IEEE International Conference on Image Processing (ICIP)*. IEEE, pp. 3478–3482.

References

- Wen, Qingsong et al. (2020). “Time series data augmentation for deep learning: A survey”. In: *arXiv preprint arXiv:2002.12478*.
- Yoon, Jinsung, Daniel Jarrett, and Mihaela Van der Schaar (2019). “Time-series generative adversarial networks”. In: *Advances in neural information processing systems* 32.

Figure: GitHub repository.


Thanks for your attention!