THE UNIVERSITY OF HONG KONG

STAT4601 TIME-SERIES ANALYSIS

Sketching Cats with ARIMA
A Time Series Analysis into Cat Sketches

Project Report

By

Jayawardana Wickramasinghe Pathiranage Lakindu Ransika

(3036094631)

Course Coordinator

Dr. LI, Guodong

School of Computing and Data Science

November 28, 2025

Acknowledgement

I would like to take this time to express my heartfelt and sincere gratitude to Dr. LI, Guodong for captivating my heart with the beauty of time series analysis. 谢谢！！

I would also like to thank the tutorial team for their hard work throughout the entire semester. 你辛苦了！！

Table of Contents

1. Introduction

1.1. Background

The advent of deep learning models such as diffusion transformers (Peebles & Xie, 2023) has shown the ability of machine learning models in generating images by developing a semantic understanding of the real world. This project explores the application of statistical models in image understanding and generation. More specifically, this project focuses on a simple cat drawing chosen from the Quick Draw dataset (Google, n.d.-a). Cats were selected due to their cuteness (England, 2025).

The Quick Draw dataset contains 50 million doodle drawings (Google, n.d.-a). These drawings were collected via the “Quick, Draw!” online game (Google, n.d.-b) from 15 million people (Google, n.d.-a) all over the world. In each game, the player is prompted with a subject, such as a cat, and is given 20 seconds to draw a doodle.

Drawing a doodle involves a series of strokes. Each stroke is a sequence of inputs that sequentially mark the corresponding pixels on the screen. The causal and sequential nature of each stroke allows it and the entire doodle to be modelled as a time series.

1.2. Objectives

This project has two main objectives:

- Model the distance from the center of a cat doodle drawing as a time series using ARIMA
- Analyze the generative capabilities of VARIMA in cat doodle sketching

2. Dataset Overview and Finding an Appropriate Cat Drawing

The original cat drawings dataset consisted of 123,202 drawings. Each drawing consisted of a sequence of strokes, and each stroke consisted of a sequence of points that highlighted the stroke’s trajectory. The strokes were merged using np.column_stack into a single sequence in order to create a 2D time series of (x,y).

To find an appropriate cat drawing, various drawings were randomly sampled iteratively. The first drawing, which had a sufficient sample size, i.e., a sample size greater than 100, and a cat face was chosen. A cat face was prioritized due to the horizontal and vertical symmetries present with respect to the center of the drawing, which would potentially exhibit a pattern discoverable by ARIMA and VARIMA models. The cat sketch chosen has index 56430 and is showcased in Figure 1.

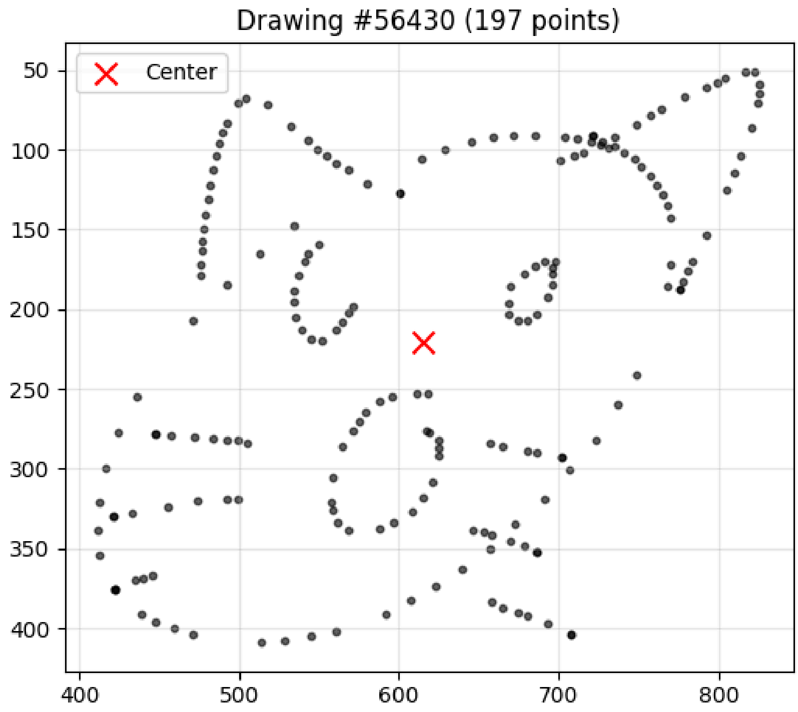


Figure 1: Chosen cat drawing

3. Modelling Distance from the Center of the Drawing using ARIMA

To model using 1D ARIMA, a 1D time series is needed. Therefore, the (x,y) coordinates of the 2D time series were used to calculate the distance from the center of the drawing for each point. Distance from the center was chosen because it incorporates information from both x and y, while being easily interpretable even after potential differencing. Figure 2 depicts the time series plot of the distance from the center.

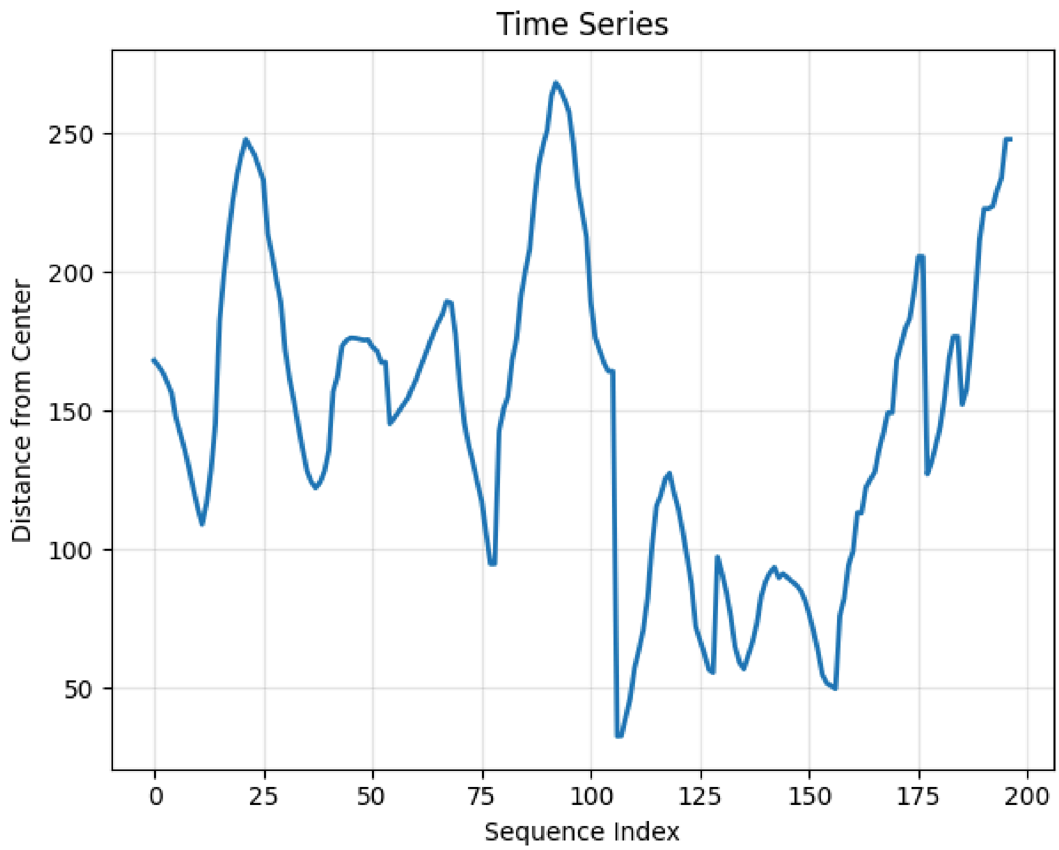


Figure 2: Time series of distance from the center

According to Figure 2, the time series data exhibits both non-stationarity in mean and variance. These are highlighted by the change in the mean after sequence index 100 and the fluctuating magnitude of the spikes, respectively. Non-stationarity in the mean highlights the need for differencing, and non-stationarity in the variance highlights the need for a variance-stabilizing transformation. Log was not directly chosen as the appearance of a cat's face is independent of the "human" element.

To further assess the non-stationarity in the mean, a SACF plot was constructed as shown in Figure 3.

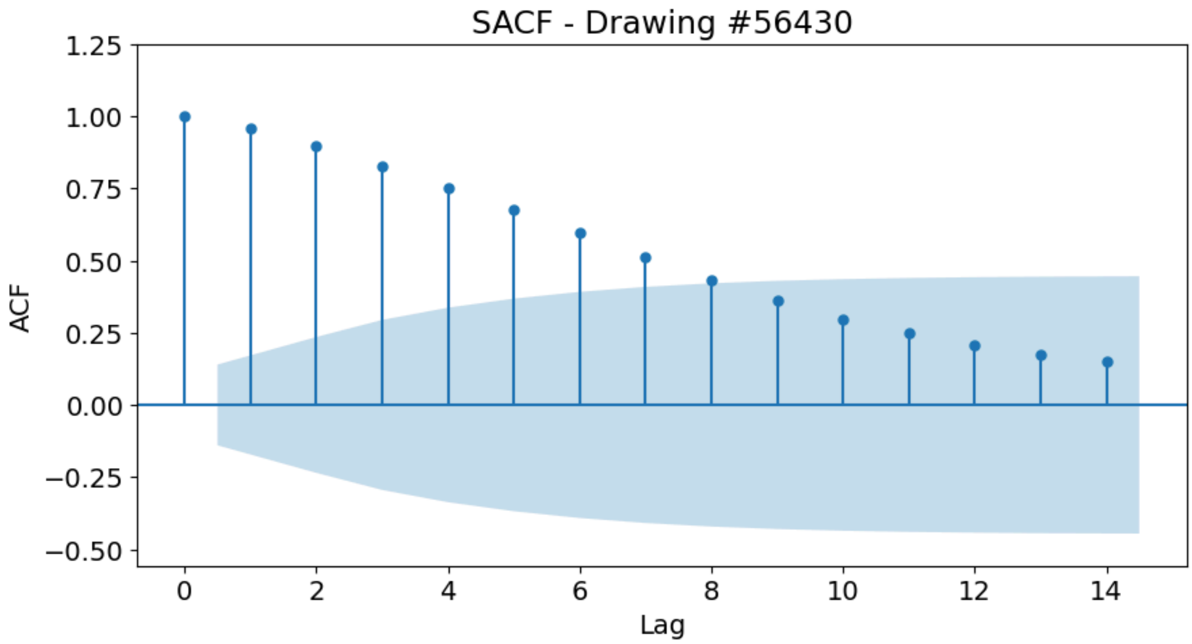
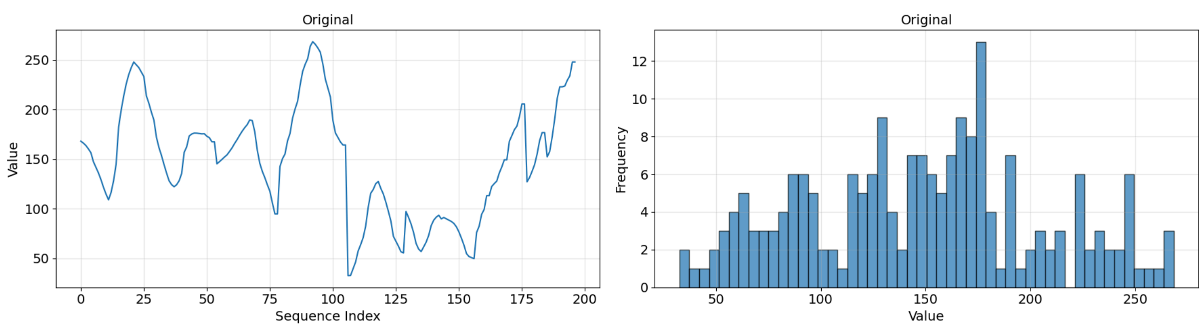


Figure 3: Sample ACF

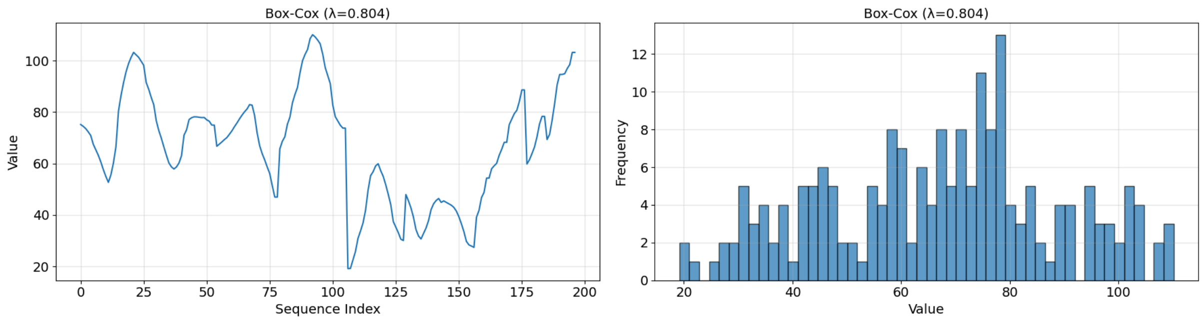
Slow decay of the sample ACF is observed in Figure 3. This highlights potential non-stationarity in the mean, further reinforcing the previous deduction from the time plot.

3.1. Handling Non-Stationarity in the Variance

Python's boxcox() optimizer was used to find the optimal Box-Cox transformation. The optimiser suggested the optimal λ to be 0.8037. Additionally, various common transformations such as square root, cube root, fourth root, natural log, and square were also tested. Figure 4 showcases a comparison between the original (a) vs the transformed (b) time series for the time and distribution plots. The plots for all the transformations can be found in Appendix A.



(a)



(b)

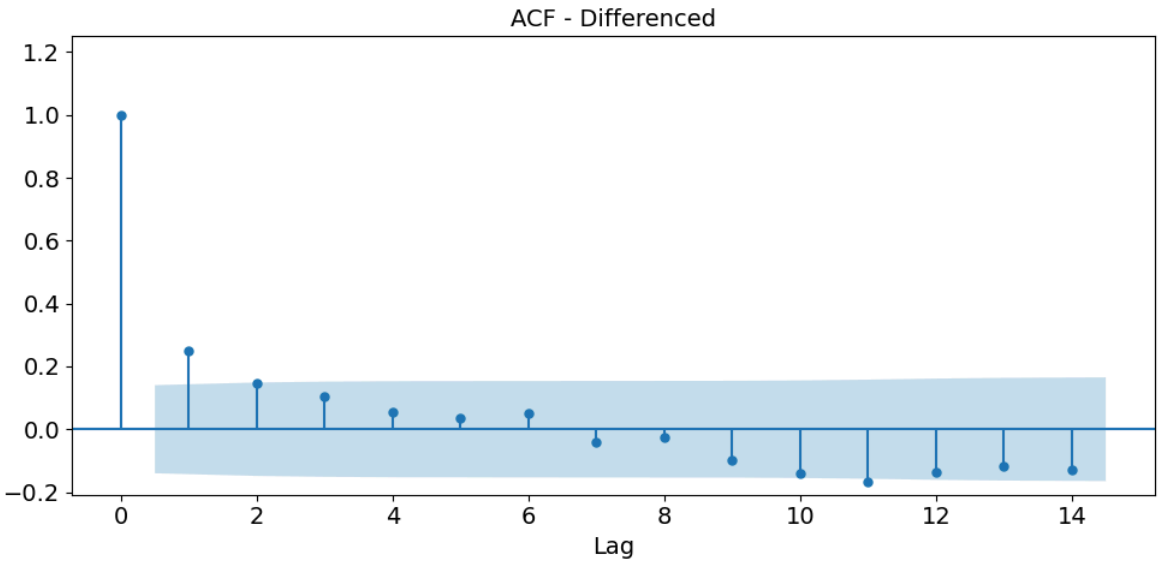
Figure 4: Comparison between the original time series and the Box-Cox optimal transformed time series. (a) Original. (b) Box-Cox Optimal

From Figure 4, it is evident that the optimal Box-Cox transformation fails to noticeably reduce the non-stationarity in variance. A similar observation can be made regarding the other transformation (as illustrated in Appendix A).

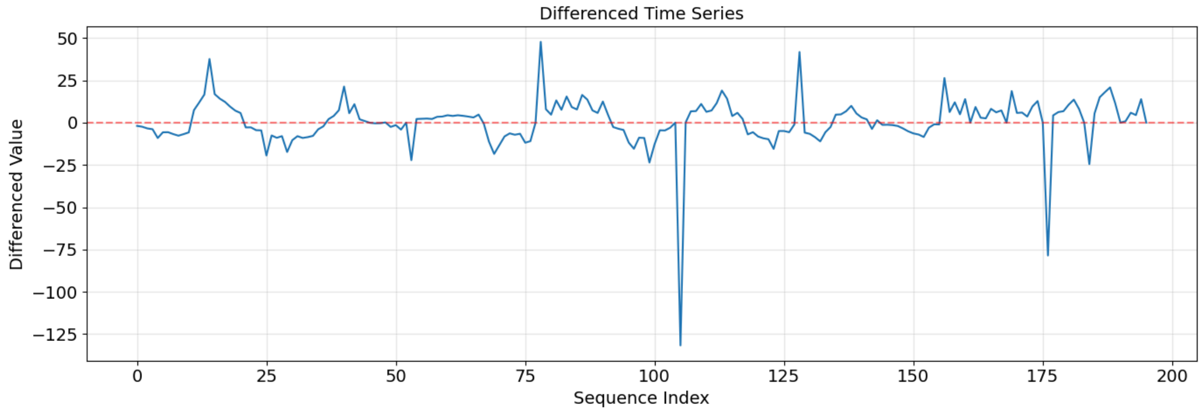
Due to the lack of noticeable effect from any of the tested transformations, no transformation was used.

3.2. Handling Non-Stationarity in the Mean

First order differencing was applied to the time series. Figure 5 depicts the time series plot and SACF after differencing.



(a)



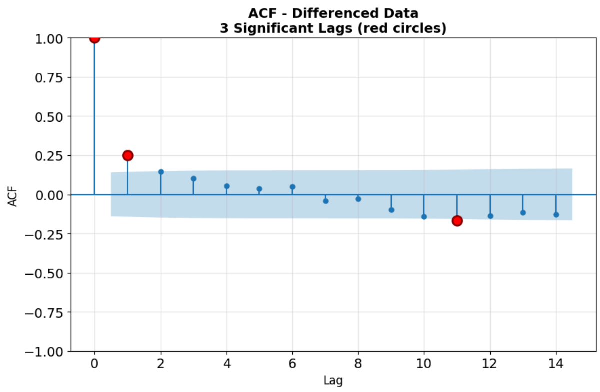
(b)

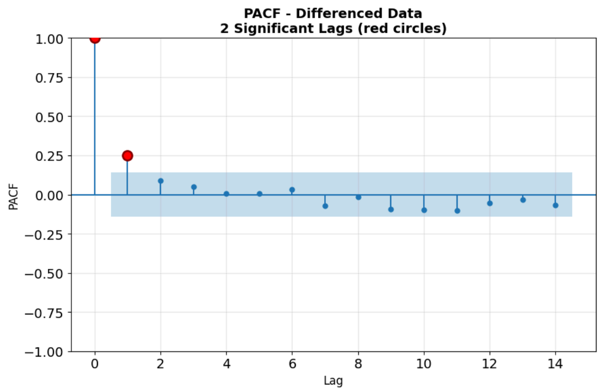
Figure 5: Plots after differencing. (a) SACF. (b) Time plot.

According to Figure 5a, the ACF decays exponentially. This highlights the stationarity in the differenced sequence. This deduction is further supported by Figure 5b, where no trend can be observed. Therefore, first-order differencing is sufficient and has resulted in a stationary sequence.

3.3. Determining the first model

To determine the first model, both ACF and PACF plots were constructed as illustrated in Figure 6. Additionally, the statistically significant lags were highlighted with a red marker.





(a) ***(b)***

Figure 6: ACF and PACF plots on the differenced data. (a) ACF. (b) PACF.

According to Figure 6, the ACF highlights 2 significant lags, while the PACF only highlights 1 significant lag, excluding lag 0. Additionally, one of the ACF's significant lags is borderline and is in a later time step.

The current time series uses differenced position. In drawing, the next change in position is affected by the previous change in position. For example, if you are drawing a diagonal, you are more likely to drag the cursor along that diagonal. This applies to other shapes as well, and drawings are composed of multiple smaller shapes. Therefore, an AR model is more suited.

Given the significant ACF lag at 11, considering an MA model would either involve fitting a very complex MA(11) model or making an additional justification to disregard the significant 11th ACF lag. Considering the previous deduction that the AR model is more suited for differenced drawing data and the need for an additional justification to fit a more parsimonious MA model, an AR(1) model would be fitted.

AR(1) for the distance from the center sequence would be referred to as ARIMA(1,1,0).

3.4. Fitted AR(1)

Equation 1 showcases the fitted AR(1) model, where y[t] corresponds to the differenced time series at the t-th time step and a[t] corresponds to the white noise at the t-th time step.

$$y[t] = 0.317 + 0.250*y[t-1] + a[t] \tag{1}$$

3.5. Fitted AR(1) Diagnostics: Residual Analysis

3.5.1. Assessing Time Plot and Normality

To find the potential limitations of the fitted model AR(1) model, diagnostics were carried out. Various plots were constructed to assess the residuals, as shown in Figure 7.

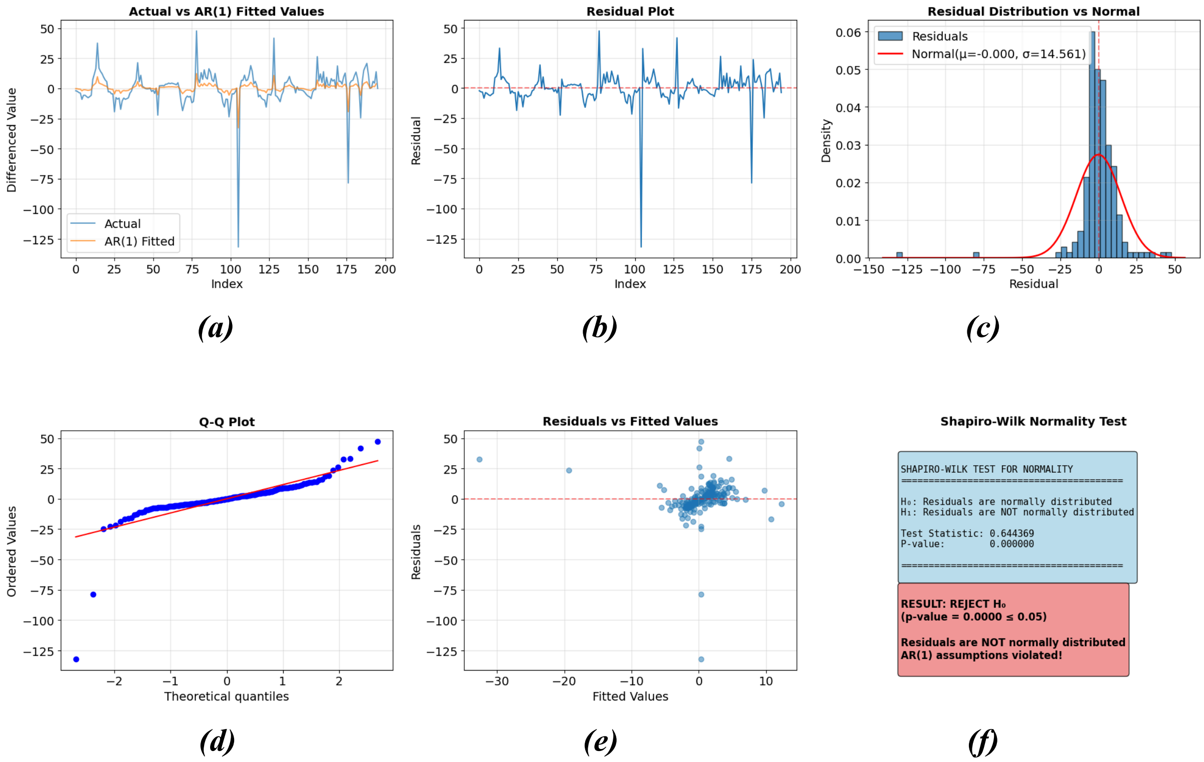


Figure 7: Time plot and assessing normality. (a) Differenced sequence vs AR(1) fitted sequence time plot. (b) Residual time plot. (c) Residual distribution vs normal distribution. (d) Residual Q-Q plot. (e) Residual vs fitted values scatter plot. (f) Shapiro-Wilk normality test results

According to Figures 7(a) and (b), the residual plot is mostly random noise except for the occasional spikes. These occasional spikes may be due to the presence of outliers.

From Figure 7(c), it can be observed that the residual distribution is much more peaked than the reference normal distribution, while being both negatively skewed (-4.194) and heavy-tailed (37.574). The heavy-tailed nature further supports the presence of outliers.

Visual examination of the Q-Q plot in Figure 7(d) highlights the presence of outliers while also indicating potential normality in the case that the outliers were removed.

Additionally, the Shapiro-Wilk test results in Figure 7(f) highlight the lack of normality in the residuals. However, considering the result from the Q-Q plot, residual plot, and normal distribution comparison plot, I hypothesized that this was most likely due to the presence of the outlier. Further exploration regarding the validity of this hypothesis can be found in sub-section 3.5.3. Reassessing Normality: Post-Outlier Removal.

3.5.2. Assessing Autocorrelations

Residual autocorrelations were assessed both individually (as shown in Figure 8) and jointly using the residual ACF and the Ljung-Box test respectively.

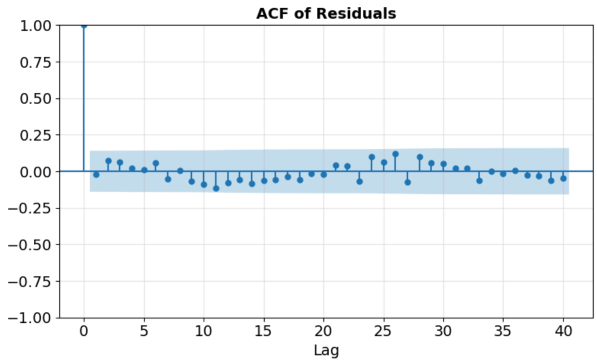


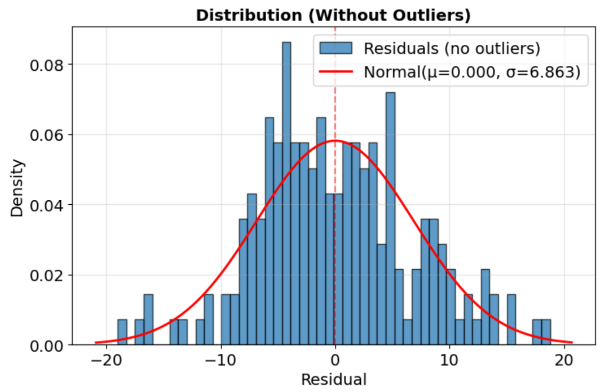
Figure 8: Residual ACF plot

The residual ACF plot (Figure 8) indicates no significant signal in the residuals, i.e., ACF at all lags is within the Bartlett's approximation bounds. The results from the ACF plot are further supported by the Ljung-Box test p-value 0.585. Therefore, according to the results, the residuals of the fitted AR(1) model are uncorrelated white noise.

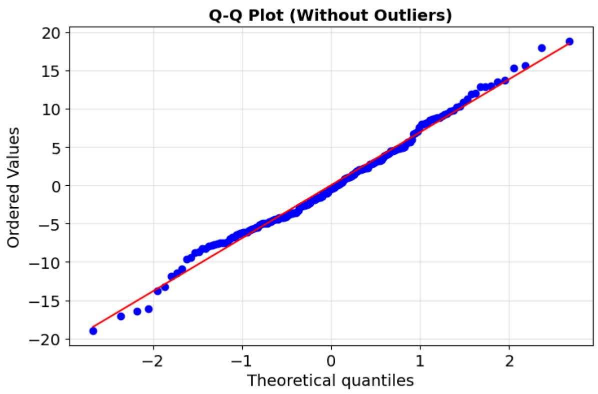
3.5.3. Reassessing Normality: Post-Outlier Removal

To test whether the lack of normality was due to the presence of outliers, normality was reevaluated after removing the residual outliers. A residual was defined as an outlier using the IQR method. According to the IQR method, an outlier is defined as: residual > 1.5 * |Q3 - Q1|, where Q3 and Q1 are the 3rd and 1st quartiles, respectively.

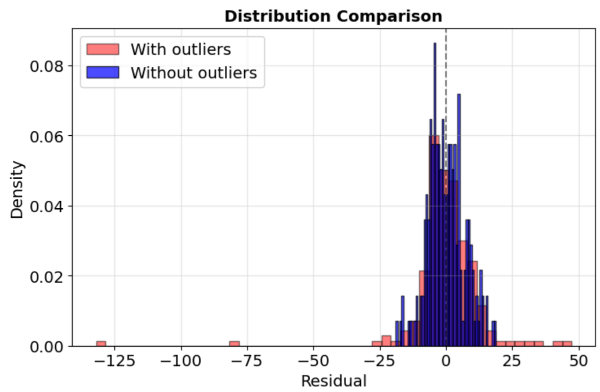
11 outliers were detected and removed, i.e., if there is an outlier at index t, index t-1 would now precede index t+1. Additionally, mean correction was applied to the residuals to ensure that their mean is zero. This is important as white noise is defined as having a mean of 0. The results are showcased in Figure 9.



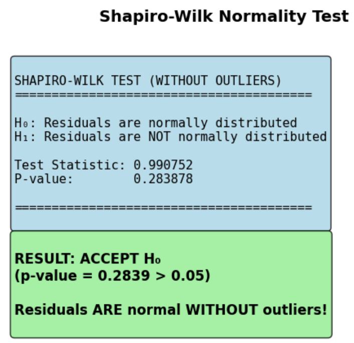
(a)



(b)



(c)



(d)

Figure 9: Plots for reassessing normality. (a) Residual distribution vs normal distribution post-outlier removal. (b) Residual Q-Q plot post-outlier removal. (c) Distribution comparison with and without outliers. (d) Shapiro-Wilk normality test results post-outlier removal.

Skewness (0.157) and kurtosis (0.090) are both low and seem acceptable according to Figure 9(a). Therefore, according to the results, skewness and kurtosis are considered acceptable without the influence of outliers.

According to Figure 9(b), the Q-Q plot seems acceptable as it is on / close to the red line (normal distribution). Additionally, the residual distribution seems to mostly match its ideal normal distribution counterpart in Figure 9(a). Furthermore, the Shapiro-Wilk test in Figure 9(d) suggests the residuals are normally distributed with a p-value of 0.284. Therefore, the results collectively suggest that the data is normally distributed post-outlier removal. Hence, the evidence against normality in sub-section 3.5.1. Assessing Time Plot and Normality were caused by the presence of outliers.

3.6. Fitted AR(1) Diagnostics: Analysis of Over-Parameterized Models

For the analysis of over-parameterized models, AR(2) was chosen as no AR model was given up, while ARMA(1,1) was chosen as MA(1) was given up (with justification for the borderline MA(11) case). Table 1 indicates the model coefficients for the baseline model AR(1) and its over-parameterized counterparts.

Model	Intercept	AR Coefficient 1	AR Coefficient 2
AR(1)	0.317±1.043	0.250±0.069	na
AR(2)	0.307±1.044	0.228±0.072	0.091±0.072
ARMA(1,1)	0.408±2.256	0.608±0.210	-0.387±0.220

Table 1: Overparameterized model analysis results

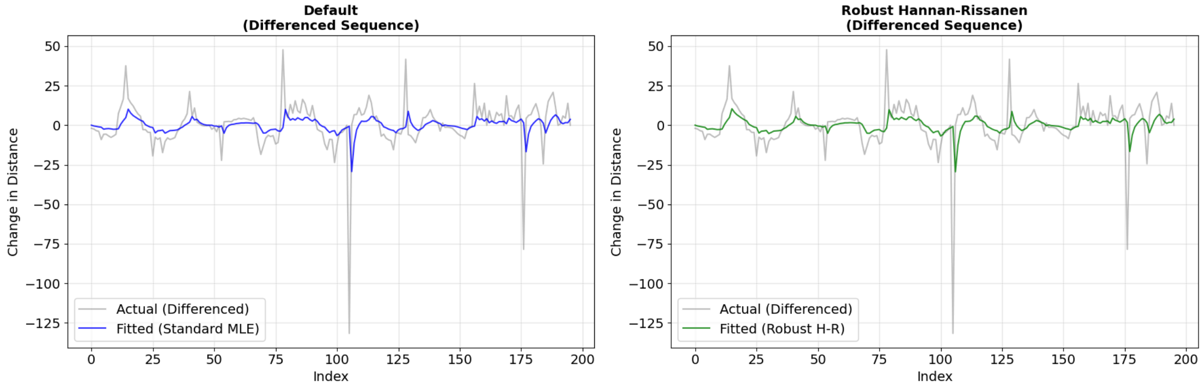
According to Table 1, AR(1) and AR(2) have similar intercept and AR coefficient 1. This highlights that the more parsimonious AR(1) model is sufficient.

On the other hand, both the intercept and AR coefficient 1 of ARMA(1,1) differ significantly from those of AR(1). Considering these results, an ARMA(1,1) was trained.

For simplicity, an ARIMA(1,1,1) was directly trained on the original distance from the center time series. Due to the presence of residual outliers in the estimated AR(1) model, two different ARIMA parameter estimation methods were tested. The default method was chosen as the baseline, and Hannan–Rissanen was chosen as the robust estimation method.

3.6.1. Determining Optimal ARIMA(1,1,1) model parameter estimation method

To determine the optimal parameter estimation method, the fitted time series from both methods were plotted as shown in Figure 10. Additionally, for analysing the model fit, the AIC was calculated.



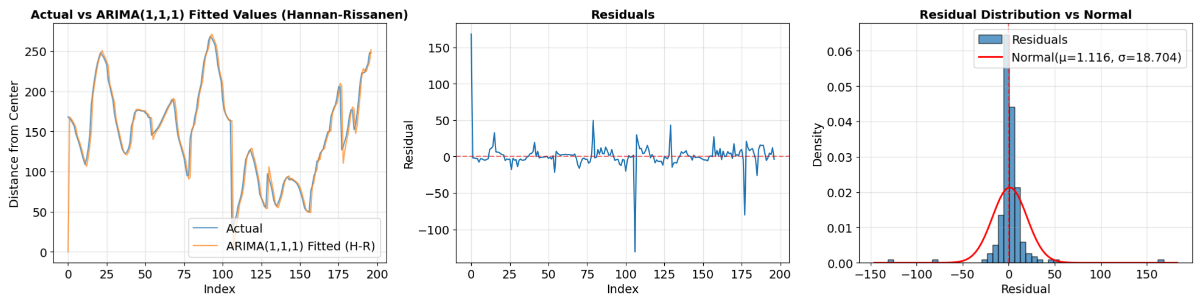
(a) ***(b)***

Figure 10: Differenced time series plot comparison between the two parameter estimation methods. (a) Default. (b) Robust.

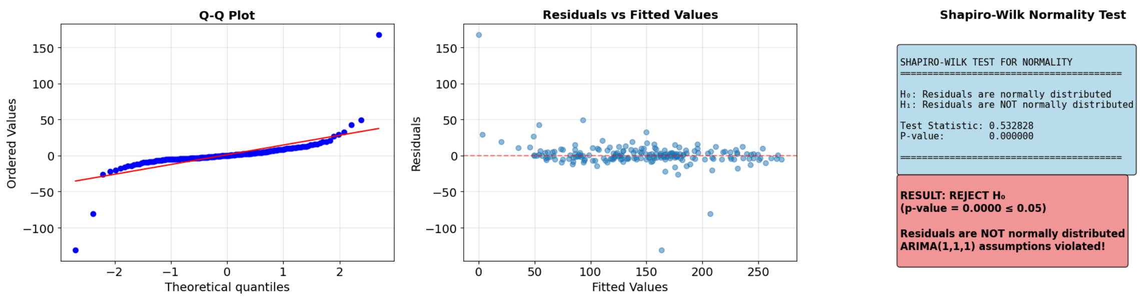
According to Figure 10, both differenced time series plots seem similar. Additionally, the difference in AIC between the two models is very small (Default: 1609.230, Robust: 1610.317). Therefore, the more robust Hannan–Rissanen parameter estimation method was chosen.

3.6.2. Fitted ARIMA(1,1,1) Diagnostics Comparison

To find the optimal model between the ARIMA(1,1,0) and the ARIMA(1,1,1) model, diagnostics on the fitted ARIMA model were carried out. Various plots were constructed to assess the residuals, as shown in Figure 11.



(a) ***(b)*** ***(c)***



(d) ***(e)*** ***(f)***

Figure 11: ARIMA(1,1,1) model diagnostics. (a) True sequence vs ARIMA(1,1,1) fitted sequence time plot. (b) Residual time plot. (c) Residual distribution vs normal distribution. (d) Residual Q-Q plot. (e) Residual vs fitted values scatter plot. (f) Shapiro-Wilk normality test results

Comparison between Figure 11(b) and Figure 7(b) highlights the less spiky nature of the ARIMA(1,1,1) residual plot. ARIMA(1,1,1) also has a skew of 1.581, which is closer to 0 than ARIMA(1,1,0) (-4.194).

However, ARIMA(1,1,0) is less heavy-tailed with a kurtosis of 37.574 in comparison to ARIMA(1,1,1) (44.149). ARIMA(1,1,0) also has a lower AIC (1603.930) than ARIMA(1,1,1) (1610.317).

Additionally, comparisons between Figures 11(c) and 7(c) alongside 11(d) and 7(d) do not show many differences. Therefore, considering these similarities and trade-offs between different model evaluation criteria, the more parsimonious AR(1,1,0), also referred to as AR(1) for the difference sequence, is chosen.

3.7. Discussion

The selection of the ARIMA(1,1,0) model, also referred to as the differenced AR(1) model, aligns well with the physical constraints of drawing a cat face doodle, where the current change in position is primarily influenced by the immediately preceding change in position.

After analysis, ARIMA(1,1,0) emerged as the optimal parsimonious model for modelling the distance from the center. After accounting for outliers, the model diagnostics support this selection: the residuals resemble random noise, satisfy normality assumptions, and show no significant autocorrelations.

However, the presence of outliers remains a significant factor. These outliers are most likely caused by the “new stroke” problem or a change in stroke direction. This is the main limitation of this model, and future work could address this by integrating seasonality into the model.

Finally, the scope of this study was limited to a single face of a cat doodle. To validate the generalizability of the modelling approach, future studies could extend the analysis to a broader set of animal face doodles.

4. Analysing the Effectiveness of Rolling Origin with Increasing Horizon Forecasting for Cat Doodle Sketching

Sketching a cat involves forecasting the (x,y) coordinates of the drawing given an initial point. This requires fitting a 2D VARIMA model. However, given the instability of ARIMA models at forecasting higher steps, it would be infeasible to forecast the entire sketch all at once. Therefore, in order to aid the forecasting process, rolling origin with increasing horizon forecasting was utilized.

A cat is sketched as follows. First, iterate over the training data. Then, at every iteration, train a model and forecast the next h-steps given the current and all previous time steps. Finally, plot the cat using only the h-th forecast step at every iteration. Note that the plotted cat is purely synthetic.

To conduct a reliable investigation, the optimal VARIMA hyperparameters have to be found. The first sub-section details the procedure of finding these hyperparameters and the selected hyperparameters, while the second sub-section outlines the investigation alongside its results.

4.1. Finding the Optimal VARIMA hyperparameters

Considering the infeasibility of finding the optimal hyperparameters for each subset of the training data, to maximise the fit to the entire dataset, the optimal hyperparameters were found for the entire time series. First, to determine the differencing order, the augmented Dickey-Fuller test was conducted over x and y separately for increasing values of d until the resulting time series was stationary. The termination differencing order was stored, and the maximum differencing order between x and y was chosen as the optimal differencing order for the VARIMA model. Afterwards, the AR and MA orders were chosen using hyperparameter grid search with AIC as the evaluation metric.

Following the results of the pipeline, VARIMA(1,1,1) was chosen as the optimal model.

4.2. Analysing VARIMA's Ability in Sketching a Cat Doodle

VARIMA's ability to sketch a cat doodle was evaluated by analysing the degradation of its drawings as the forecast horizon h increases. In order to do so, cats were sketched at increasing values of h, namely, 1, 2, 5, 10, 20, 30, 50, and 100. Figure 12 depicts the sketched cats in red while highlighting the original cat drawing in black for reference. The synthetic cats in isolation and the trajectory of the cat sketch can be found in Appendices B and C, respectively.

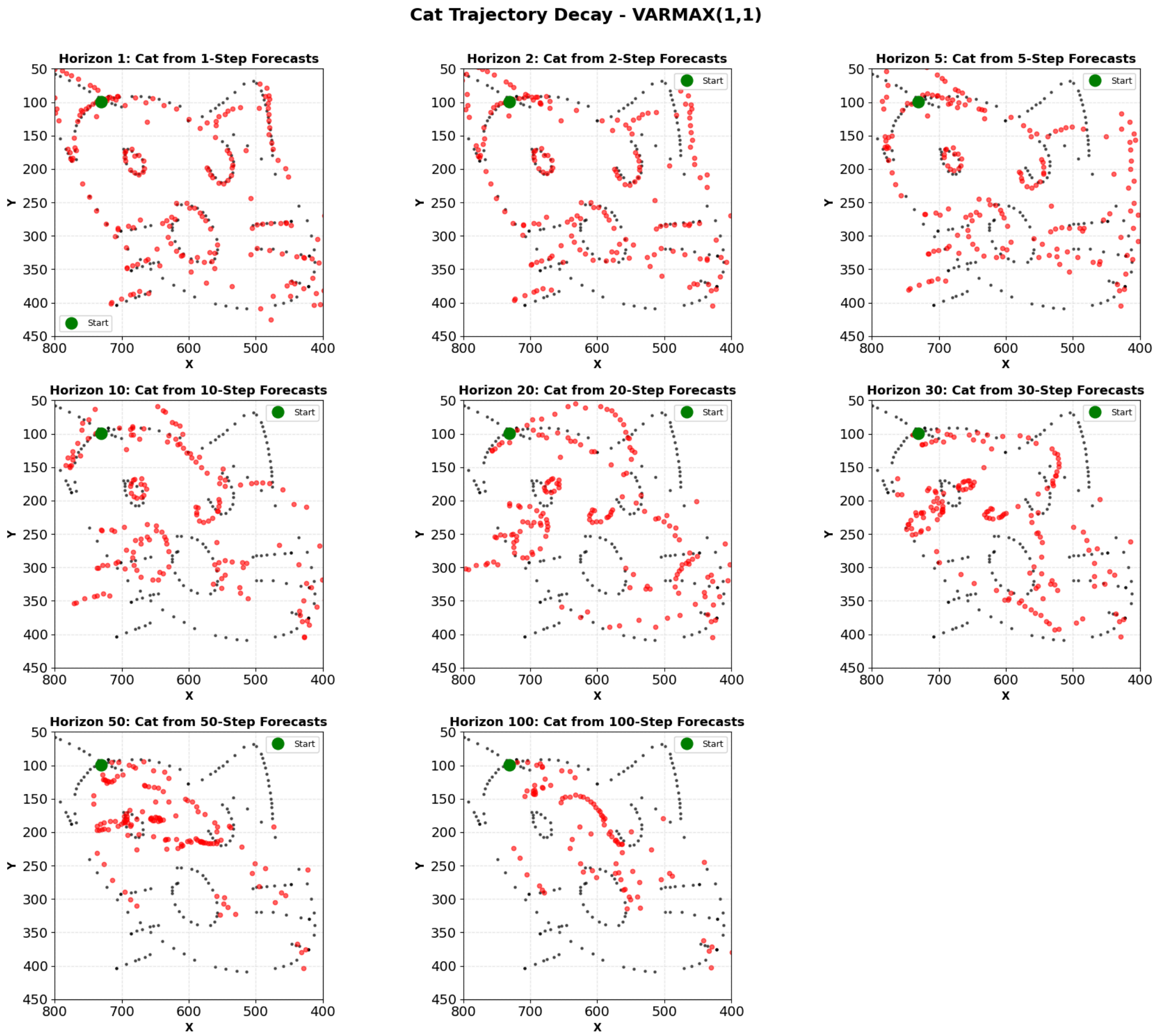


Figure 12: Sketched cat with original cat drawing as reference for increasing horizons forecasting at h=1,2,5,10,20,30,50,100

According to Figure 12, from h=1, as the forecast horizon increases, the sketched cat rotation from the original image increases. The rotation indicates the degradation in the synchronisation between x and y as lag increases. Additionally, the continued increase of this rotation reveals how VARIMA accumulates errors as the forecasting horizon increases.

Figure 12 also highlights the continuous degradation of the sketched cat's right ear as the forecasting step increases, where it seems to disappear after the 5th horizon. This accelerated degradation, in comparison to the left ear, is likely caused by the "new stroke" problem. Since a new stroke is drawn immediately after the old stroke, the previous time steps at the beginning of a new stroke are from the previous stroke. Consequently, they carry little information about the new stroke, accelerating the error degradation.

Additionally, another degradation can be observed regarding the lower boundary of the cat's face. It appears to degrade much faster than the top boundary between the ears. While the top boundary can be observed until h=10, the bottom boundary disappears after h=1. This highlights another asymmetry in VARIMA's ability to draw.

However, VARIMA seems particularly good at drawing the cat's eyes and mouth. Despite the rotation, they are identifiable and very clear until h=30. This might be due to the symmetry in these parts and their circular appearance.

4.3. Discussion

The VARIMA model demonstrates the ability to sketch a cat face doodle using rolling origin with increasing horizon forecasting at shorter horizons (h=1,2,5). However, the performance degrades significantly as the horizon expands, rendering the image unidentifiable beyond h=30.

A key finding of this study is the various asymmetries present in VARIMA's generative ability. While the cat's left ear and top part (between the two ears) degrade slowly, the right ear and bottom boundary degrade rapidly. This could be due to the "new stroke" problem discussed before, and future work can investigate this further by comprehensively analysing other animal face doodles.

Finally, the results also revealed a "rotational drift", which distorts the image as a whole as h increases. This may be due to the degradation in synchronisation between the two dimensions of the VARIMA model. Given the impact of this issue on the entire generated image, future work could explore potential solutions that utilize alternative coordinate systems.

5. Conclusion

This project successfully explored the application of statistical models in modelling and generating images. Through modelling distance from the center and accounting for outliers, this study found ARIMA(1,1,0) to be the optimal parsimonious model that effectively captures the sequential nature of the drawing process, without violating any modelling assumptions. Furthermore, this study also showed the ability to generate sketches using 2D VARIMA by utilizing rolling origin with increasing horizon forecasting, at lower forecasting horizons. However, the influence of outliers on the ARIMA model and the rotational drift observed in the VARIMA model suggest that while these methods successfully met the study's objectives, they remain limited in practical applications.

References

England, A. (2025, June 24). *The Science Behind Why Your Cat is The Cutest*. https://www.kinship.com/uk/cat-lifestyle/why-cats-are-cute-science

Google. (n.d.-a). *Quick, draw! The data*. Quick, Draw!; Google. Retrieved November 29, 2025, from https://quickdraw.withgoogle.com/data

Google. (n.d.-b). *Quick, draw!* Quick, Draw!; Google. Retrieved November 29, 2025, from https://quickdraw.withgoogle.com/

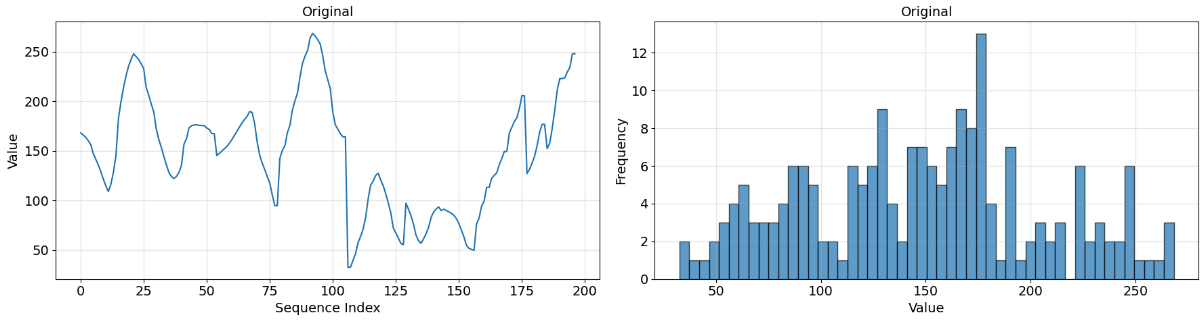
Google. (n.d.-c). *The Quick, Draw! Dataset*. GitHub; googlecreativelab. Retrieved November 29, 2025, from https://github.com/googlecreativelab/quickdraw-dataset

Google. (2017, May 18). *quickdraw_dataset*. Google Cloud Platform; Google. https://console.cloud.google.com/storage/browser/_details/quickdraw_dataset/full/raw/cat.ndjson;tab=live_object

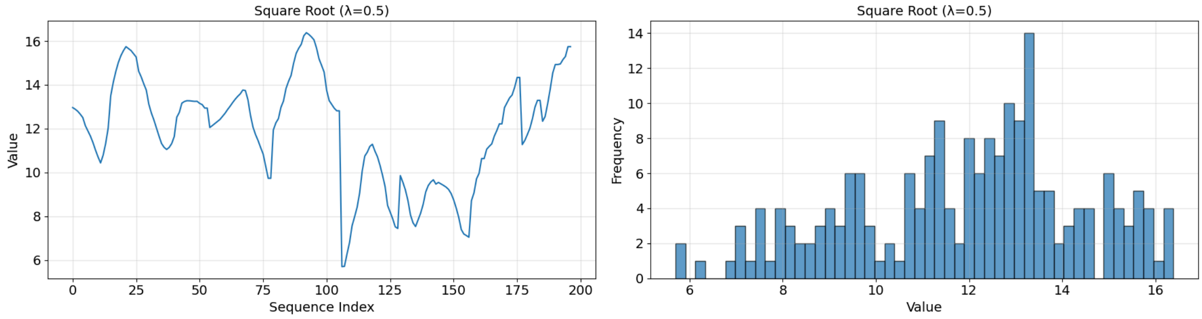
Peebles, W., & Xie, S. (2023, March 2). Scalable Diffusion Models with Transformers. The Computer Vision Foundation. https://openaccess.thecvf.com/content/ICCV2023/papers/Peebles_Scalable_Diffusion_Models_with_Transformers_ICCV_2023_paper.pdf

Appendix A

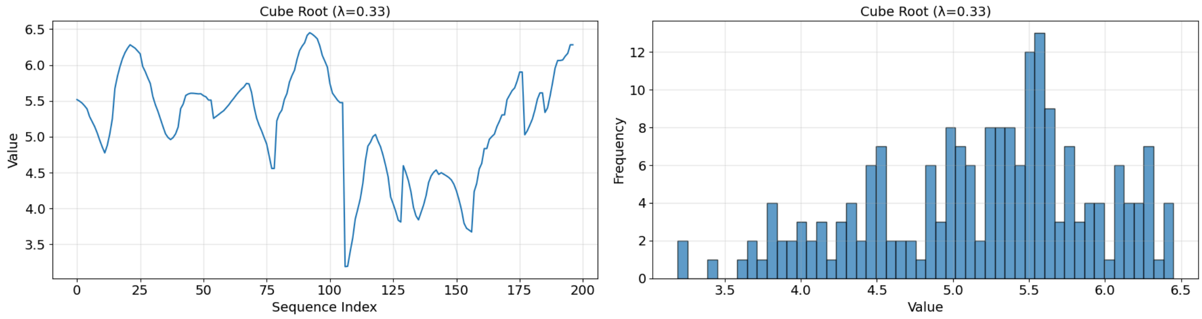
Comparison between original time series and other transformed time series



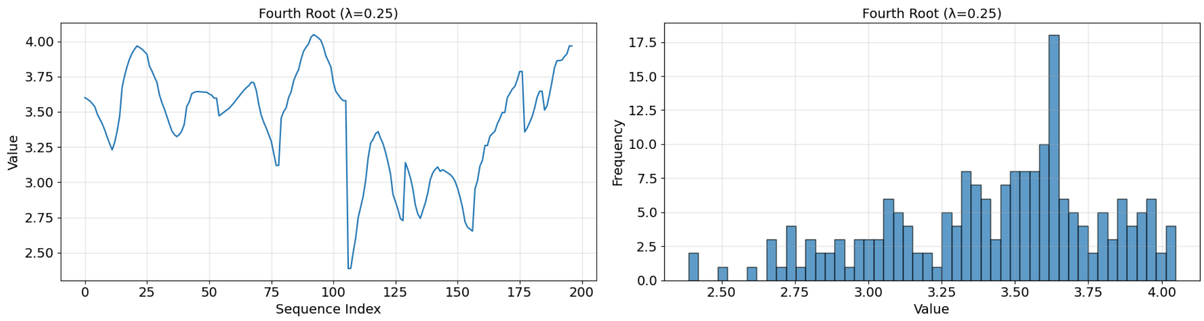
(a) Original



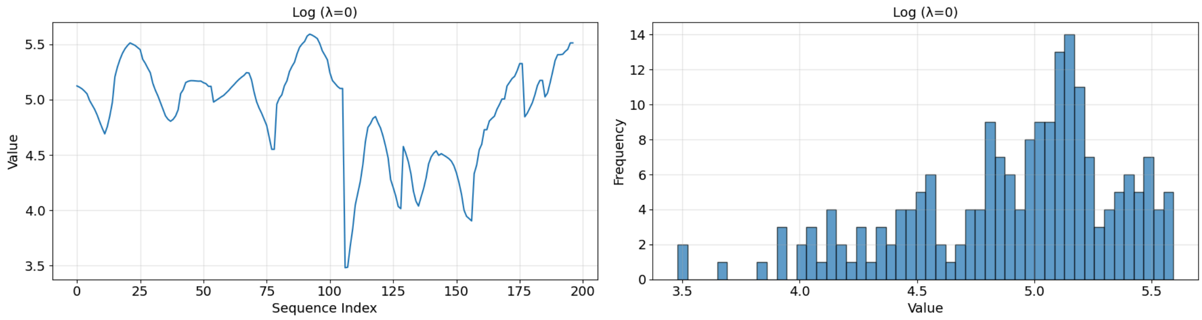
(b) Square root



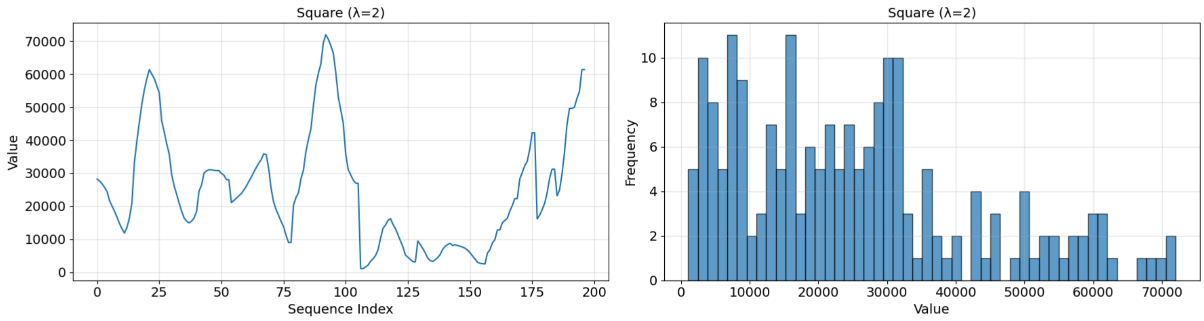
(c) Cube root



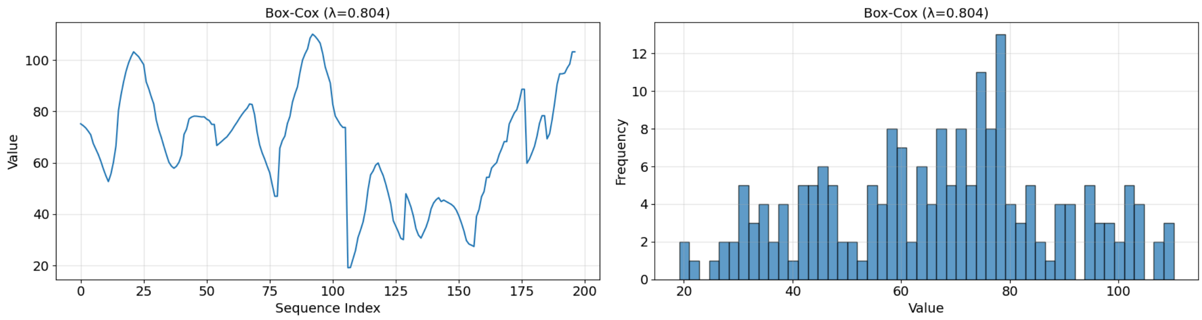
(d) Fourth root



(e) Natural log



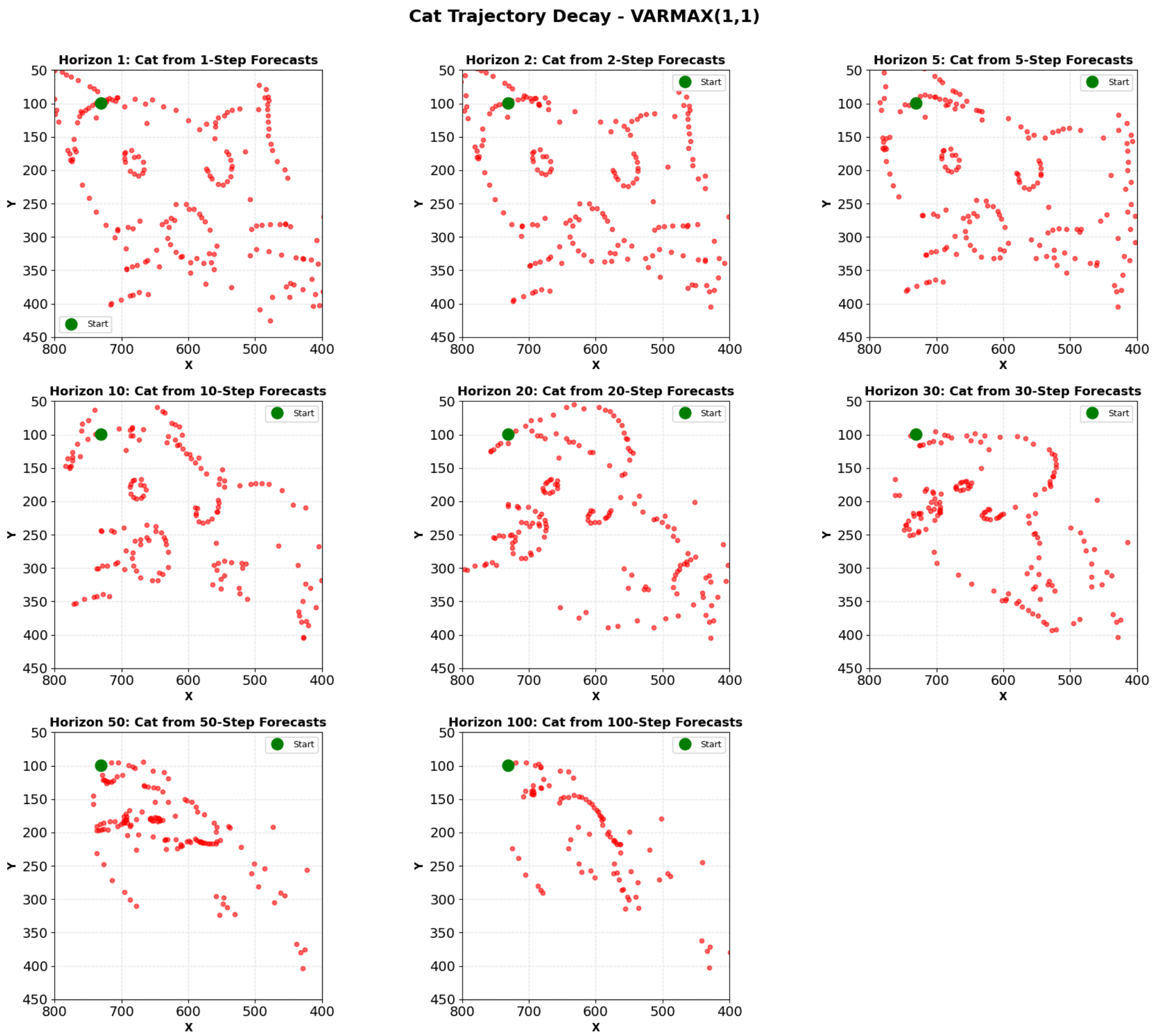
(f) Square



(g) Box-Cox

Appendix B

Sketched Cat for Increasing Horizon Forecasting



Appendix C

Sketched Cat Trajectory for Increasing Horizon Forecasting