http://terra-ai.sg
https://ai4impact.org
hi@ai4impact.org

AI Workshop – Lab 2

Sine Curve Prediction – The ARIMA model

Lab Summary

- The sine curve prediction problem.
- The ARIMA model, Persistence.
- Implementing, training & testing the model.
- Multiple tries and generalization.
- Discussion of results.

The Sine Curve Prediction Problem

The goal of this workshop is to help you build AI models for prediction. We'll start with a simple prediction model, one where we know the outcome perfectly.

The aim of this Lab is to help you understand the basic challenges in making predictions. The Sine Curve is a relationship:

$$y(t) = \sin(ft + \phi)$$

Where

- **y** is the value to be predicted,
- **t** is the time,
- **f** is the (fixed) frequency,
- ϕ is the (fixed) phase, which just shifts the curve left (+) or right (-).

Our model needs to predict a future **y** given one or more older **y**'s. For example, we might want to “predict” y(t+5) given y(t), y(t-1) and y(t-2).

This type of prediction problem is very common:

- We want to predict demand of an online cosmetic product 30 minutes into the future, given the demand within the last hour;
- We want to predict the electricity usage at a factory next month given data from the previous 3 months;
- We want to predict the world's temperature in the next 10 years given data from the last 1000 years.

The prediction horizon (30 mins, 3 months, 10 years, etc.) is called the **lead time**. Intuitively, the longer lead times, the harder the prediction challenge.

Counter-intuitively, *short* lead times are also tough prediction challenges. We'll see this towards the end of this Lab.

The ARIMA model

The ARIMA model is a simple linear model that combines previous data points to make a prediction. For example, a size n=3 ARIMA model would combine y(t), y(t-1) and y(t-2) linearly:

$$y_{predicted}(t+5)=w_1\, y_{actual}(t)+w_2\, y_{actual}(t-1)+w_3\, y_{actual}(t-2)$$

Quiz: In the ARIMA model above, what is the lead time?

The goal of "training" is then to determine the weights w_1 , w_2 and w_3 . There are a number of ways to do this, for example, using a pseudo-inverse or through gradient descent. We'll use gradient descent in this lab.

The ARIMA model is used frequently in many areas as a prediction "benchmark" against which other prediction models (eg Neural Networks) may be compared against.

Persistence

The simplest ARIMA model is using n = 1 and fixing the weight $w_1=1$. This gives the prediction model called **persistence**:

$$y_{predicted}(t+5)=y_{actual}(t)$$

Persistence essentially says the predicted value is the same as the one experienced now. While this is unlikely to be correct for long lead times, <u>for short lead times, the persistence prediction is very hard to beat!</u>

So, in all practical forecasting applications, you must compare your model's loss against those made by persistence.

Lab 2a: Training & Amending the ARIMA model

Go to the lab-2 folder in the Smojo editor.

Step 1: Install the CSV data – Uncomment the appropriate lines in lab-2.m and run it.

```
1.0 0.0 "sine/train.csv" sine>data
1.0 2.0 "sine/test.csv"  sine>data
```

The first parameter is the **frequency** and the second is the **phase**.

Step 2: Test the model for errors – Open up lab-2.m and make the necessary changes to test the model.

- Ensure you receive no error messages.
- Check that the "prototext" files are all not empty.

Step 3: Edit the configuration file – Open up the config.m file.

Quiz:

- How many configurations do we have defined?
- These configurations represent the sizes of the ARIMA models. Which configuration do you think would perform the best?
- Add a configuration for 16 ARIMA weights. Remember to put in the spaces.

Step 4: Train your model, then plot your model losses.

Quiz:

- Which model performed the best?
- Did the results confirm your expectations?

Lab 2b: Changing the prediction Lead Time

In Lab 2a, the prediction lead time was 5 time steps. We will now examine the effect of changing the lead time.

Step 1: We'll change the prediction lead time making it larger, to 10 steps. Open up **lab-2b.m**.

1. Uncomment the line to change the lead time.
2. Save the sine data to the server. You need to uncomment the appropriate lines.

Step 2: Train the model as usual and plot the test losses.

Quiz:

- Do you expect the new models to perform better than the old ones? Why?

Step 4: Repeat these steps with a lead time of 1.

Quiz:

- Do you expect the new models to perform better than the old ones? Why?

Discussion of Results

1. Which losses (training or testing) is indicative of the model's generalization?
2. Which configuration gives the best generalization?
3. In this situation, would feeding the model more training data improve performance? Explain your answer.
4. How do lab-2a's results (for lead time = 5 steps) compare to lab-2b's results for the same lead time but different test frequency? What conclusions can you draw?
5. Propose how you would estimate the persistence losses for lab-2a and lab-2b.
6. Did your models beat persistence?
7. What is a simple way to incorporate persistence loss into your loss calculation?
8. *Calculate your answers for (7). Hint: You may want to use the word **test-loss** which was encountered in Lab 1. Also, you may need this word:

```
: min-loss ( #losses – – n )
    9999999 swap #values ['] min reduce ;
```

For example to calculate the minimum test loss for configuration 2:

`2 test-loss min-loss` .

9. *If the frequency in the “test” dataset were changed to (say) 2.0 Hz, would you expect the performance to improve or decrease? Explain your answer.

10. *Do you think the experiments done so far present a fair demonstration of the ability of ARIMA to model sine data? Explain your answer.

11. **What makes sine curve prediction an “easy” prediction problem? Qualify and explain your answer.

12. **Optional**: ***Write a word **`best-config ( k -- n )`** that takes the maximum number of configurations (eg, 16) and searches them all to get the configuration with the best min-loss. Hint: `1 2 … k take` produces a sequence from 1,2…,k.