**Week 2 Quiz**

1. **What is a windowed dataset?**

**A fixed-size subset of a time series**

**There’s no such thing**

**The time series aligned to a fixed shape**

**A consistent set of subsets of a time series**

2. **What does ‘drop\_remainder=true’ do?**

**It ensures that all rows in the data window are the same length by cropping data**

**It ensures that all rows in the data window are the same length by adding data**

**It ensures that the data is all the same shape**

**It ensures that all data is used**

3. **What’s the correct line of code to split an n column window into n-1 columns for features and 1 column for a label**

**dataset = dataset.map(lambda window: (window[n-1], window[1]))**

**dataset = dataset.map(lambda window: (window[:-1], window[-1:]))**

**dataset = dataset.map(lambda window: (window[-1:], window[:-1]))**

**dataset = dataset.map(lambda window: (window[n], window[1]))**

4. **What does MSE stand for?**

**Mean Series error**

**Mean Slight error**

**Mean Second error**

**Mean Squared error**

5. **What does MAE stand for?**

**Mean Average Error**

**Mean Advanced Error**

**Mean Absolute Error**

**Mean Active Error**

6. **If time values are in time[], series values are in series[] and we want to split the series into training and validation at time 1000, what is the correct code?**

**time\_train = time[split\_time]**

**x\_train = series[split\_time]**

**time\_valid = time[split\_time]**

**x\_valid = series[split\_time]**

**time\_train = time[:split\_time]**

**x\_train = series[:split\_time]**

**time\_valid = time[split\_time:]**

**x\_valid = series[split\_time:]**

**time\_train = time[split\_time]**

**x\_train = series[split\_time]**

**time\_valid = time[split\_time:]**

**x\_valid = series[split\_time:]**

**time\_train = time[:split\_time]**

**x\_train = series[:split\_time]**

**time\_valid = time[split\_time]**

**x\_valid = series[split\_time]**

7. **If you want to inspect the learned parameters in a layer after training, what’s a good technique to use?**

**Assign a variable to the layer and add it to the model using that variable. Inspect its properties after training**

**Decompile the model and inspect the parameter set for that layer**

**Run the model with unit data and inspect the output for that layer**

**Iterate through the layers dataset of the model to find the layer you want**

8. **How do you set the learning rate of the SGD optimizer?**

**Use the RateOfLearning property**

**Use the Rate property**

**Use the lr property**

**You can’t set it**

9. **If you want to amend the learning rate of the optimizer on the fly, after each epoch, what do you do?**

**Use a LearningRateScheduler and pass it as a parameter to a callback**

**Callback to a custom function and change the SGD property**

**Use a LearningRateScheduler object in the callbacks namespace and assign that to the callback**

**You can’t set it**