Recap: Time Series

In this reading, we will do a quick recap of time series from Course 2 as we will be leveraging that knowledge for a slightly different application than what we discussed previously (i.e., anomaly detection): gesture recognition with a custom-built magic wand.

Supervised vs. Unsupervised Learning

So far, we have tackled many interesting machine learning tasks that cover a variety of sensory **modalities** (e.g., vision, audition). All of these tasks required a labeled dataset - a known output value for a given set of inputs - and were thus comfortably situated within the **supervised learning paradigm**, which we first discussed in Course 1.

However, frequently, we encounter datasets in which our output variable is unknown. Whether because we do not have access to the output variable, or we are merely interested in the internal structure of our dataset, we cannot resort to our standard machine learning framework which relies on training and test sets. Instead, we must adopt an alternative, equally useful, paradigm known as **unsupervised learning** to approach these tasks, which we discussed in Course 2.

There are still similarities between the supervised and unsupervised learning paradigms. Firstly, both assume the existence of a dataset that is used to train our machine learning algorithm. The performance of both generally improves as more data becomes available. Nevertheless, unsupervised learning is distinct in that we do not require the existence of a predictor variable. Unsupervised learning looks at the internal structure of a dataset and attempts to either group this data systematically, known as **clustering**, or look for abnormalities within the dataset, known as **outlier analysis**, or **anomaly detection**.

Anomaly Detection

In course 2, we were mostly interested in anomaly detection, predominantly due to its potential for predictive maintenance (i.e., noticing abnormalities in machines that can be resolved before catastrophic events). Anomaly detection can be performed **spatially** or **temporally**, either looking at associations between different physical locations, or the same location at different points in time.

Most sensory devices are fixed in discrete locations, such as those monitoring the temperature and vibrational characteristics of a piece of machinery. Inherently, these

devices output a temporal signal that can be monitored and used to look for anomalies over time. These anomalies manifest in myriad ways: a sudden temperature change, a sudden increase in vibration or sound, or a sudden drop in pressure. Given sufficient explanatory variables, anomaly detection algorithms allow us to detect outliers and perform corrective actions in a timely manner to mitigate the possibility of catastrophic events, such as an error, a machine breakdown, or even an explosion.

Many algorithms exist for the detection of anomalies in time series data. One algorithm that was discussed in the previous course was **k-means**, which looks at the distance of a data point in feature space from a set of discrete clusters that represent the majority of data points. We saw that one drawback of k-means was its poor performance on high-dimensional data due to the **curse of dimensionality**. However, we were able to improve performance by reducing the dimensionality of the data with **t-SNE**. Anomaly detection can also be achieved using neural network structures known as **autoencoders**, which attempt to collapse knowledge of the dataset in a small set of neurons located at the midlayer of the autoencoder (this formulation is very similar to principal component analysis). Autoencoders tend to have superior performance to traditional methods due to their ability to capture complex non-linear associations.

Magic Wand

In the remainder of this section, we will leverage this prior knowledge for a slightly different purpose: building in gesture recognition using a custom-built magic wand.