Bike Forecast

David Starkey

August 27, 2018

1 Introduction and Data Ingestion

This project provides the details of bike hires in Montreal Canada dating from 2014 up to August 2017. The goal is to use statistical models to forecast the expected daily number of bike hires between two stations (and) for one weeks worth of hires between xxx and xxx. Each bike hire records the date, departure station and arrival station. This data is contained within several csv files which collectively total 15.3 million entries.

This summary is presented as follows. Section xx details the data ingestion process. Section xx presents the findings of the initial exploratory data analysis and includes several figures that motivate the model fitted to the global sample discussed in Section xxx. The results of the model fitting are provided in Section xxx.

2 Data Ingestion

3 Exploratory Data Analysis

3.1 Time Series

Before deciding how best to model the forecasting problem, a universally sound first step is to visualise the data. I use Pythons matplotlib module to plot the number of bike hires as a time series, concatenating all four years of observations together. Figure 1 presents some very important information on the Time Series

- The time series is periodic. It exhibits a similar annual pattern with bike hires becoming more popular in the summer months.
- The amplitude of the variations appears to be increasing over time (smooth red line Figure 1). This suggests bike hires are increasing in popularity.

While Figure 1 provides useful information on the periodicity of the time series, this information is much clearer to see when presented as a power spectrum. The power spectrum as a function of frequency $P(f)$ can be computed from the fourier transform of time-series data $F(f)$ where

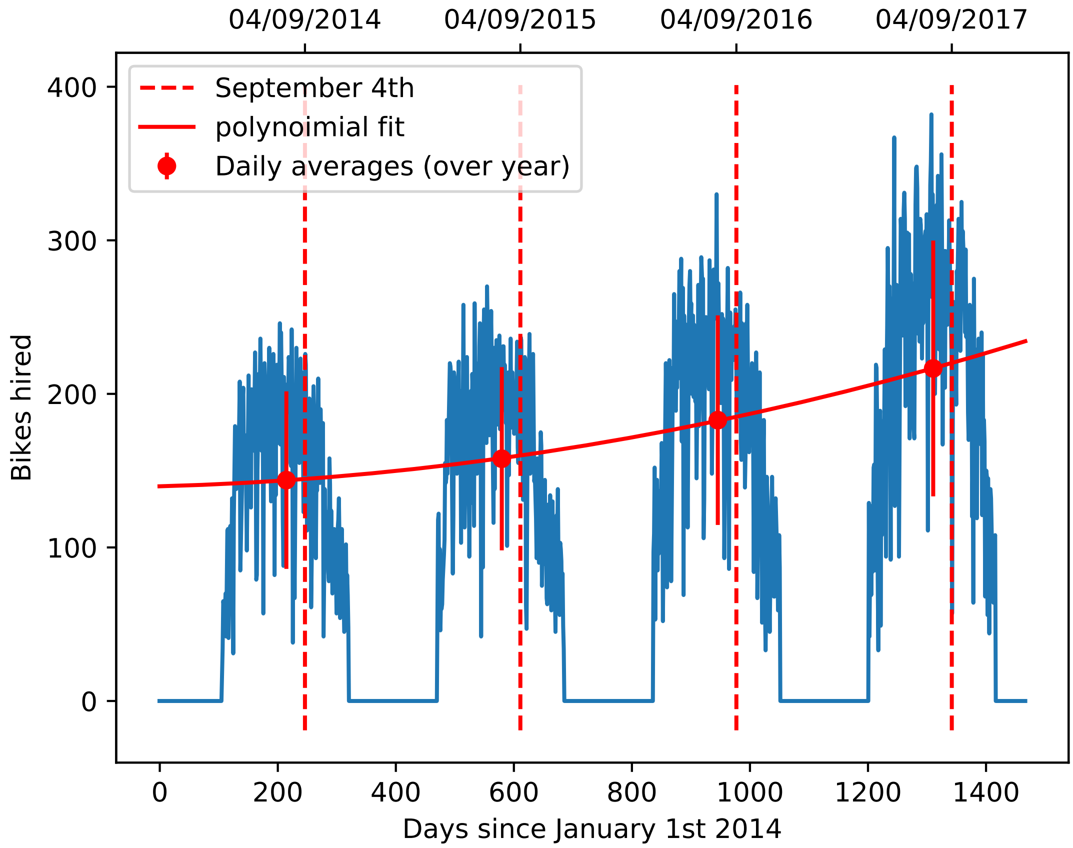

Figure 1: Time series of bike hires. Y axis plots the number of bikes hired per day as a function of day number (days are measured relative to 1st January 2014). The vertical red dashed lines show 4th September of each year (the start of the forecast week) and the smooth vertical polynomial fit shows how the average daily number of bike hires increases over the four years of data.

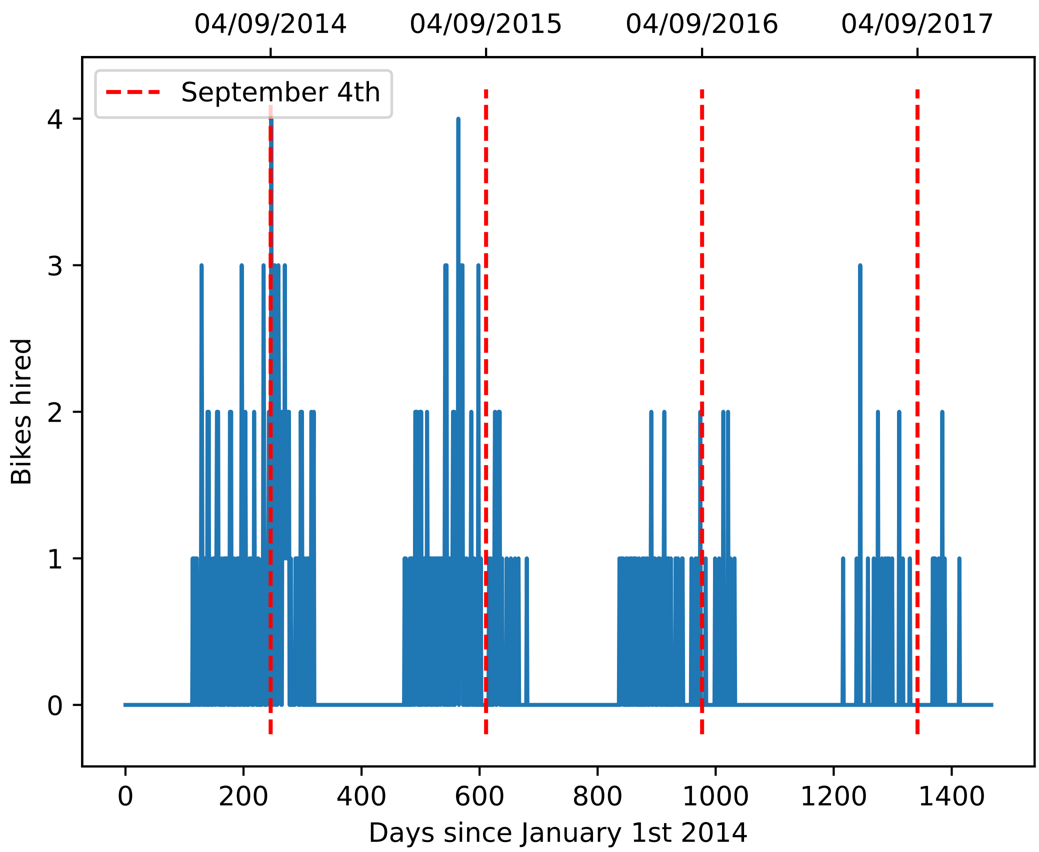

Figure 2: Same as Figure 1 but restricting the time series only to bike hires departing at station xx and ending at station xx

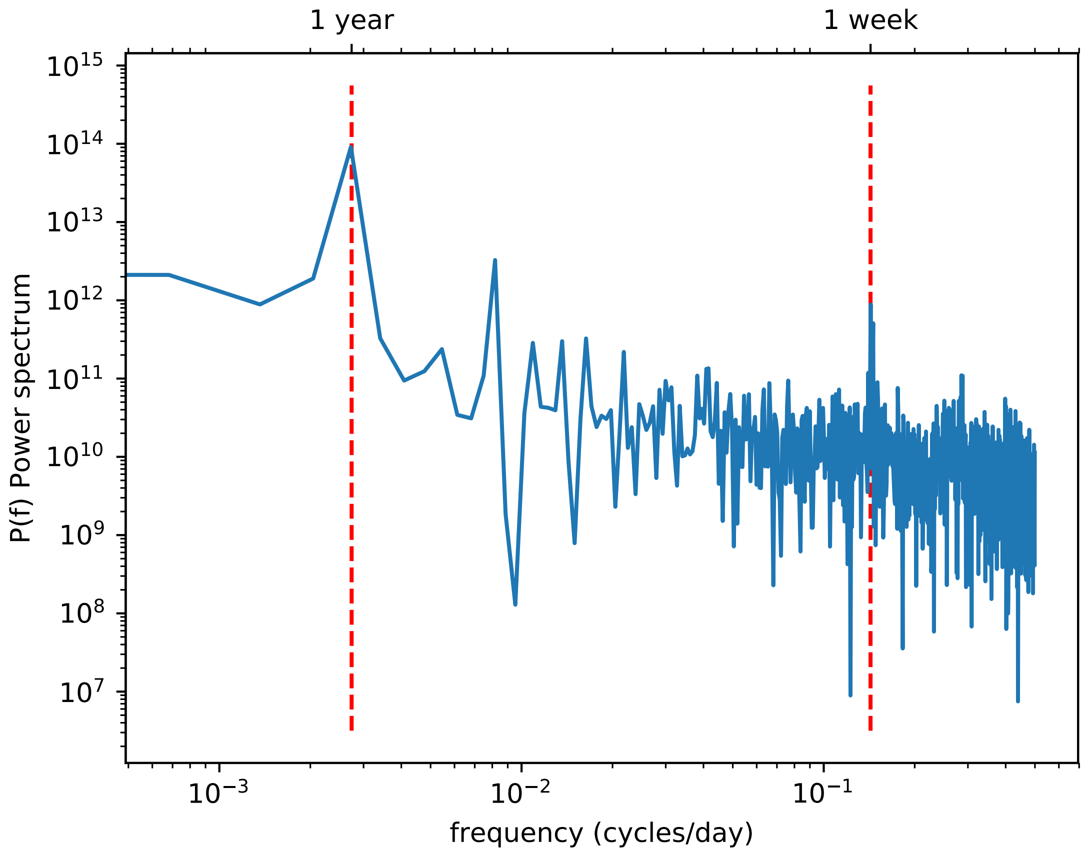

Figure 3: Power spectrum $P(f)$ versus frequency of the time series data presented in Figure 1. The peaks at one year and one week timescales (red dashed lines) show that bike hire goes through seasonal and weekly cycles of popularity.

$$F(f) = \int_{-\infty}^{\infty} f(t) e^{-2\pi f t} dt, \tag{1}$$

and $P(f)$ is then

$$P(f) = F^{*}(f) F(f), \tag{2}$$

where $*$ denotes complex conjugation. Qualitatively, the power spectrum $P(f)$ tells us if there are strong periodic features in our data set. Figure 3 demonstrates that the bike hires in this data set go through seasonal and weekly cycles of popularity, likely due to the seasonal weather / holiday season and weekend trend increases.

4 Model fitting (Auto Regressive Moving Average - ARMA)

Now that the periodicity of the bike hires is better understood, I will fit a model to the global sample to forecast the bike hires for the week beginning 4th September 2017. The model I use here is known as an Auto-Regressive-Moving-Average of ARMA model. AutoRegressive (the AR part) just means the model $Y(t_i)$ depends on the previous observation $Y(t_{i-1})$.

The moving average part (MA) just means the process depends also on some unobserved quantity that we model as a draw from a Gaussian distribution. The numerical details of the model are outlined in Section 4.1 but the reader is welcome to skip to the results in Section 5 if desired.

4.1 Theory

ARMA models are ideal for forecasting time series data and in general take the form

$$Y(t) = C + G(0, \sigma^2) + \sum_{i=1}^{p} A_i Y(t_{i-1}) + \sum_{i=1}^{p} B_i G(t_{i-1}) \tag{3}$$

where C is the background level of the time series, $G(0, \sigma^2)$ indicates a draw from a Gaussian distribution with mean 0 and variance σ^2.

As with most optimisation problem, we want to find the best values of the parameters θ that maximize a likelihood function $L(Y(t = 1...T)|\theta)$, where the parameter vector θ is explicitly given by

$$\theta = \theta \left(C, \sigma^2, A_{i=1,p}, B_{i=1,q} \right), \tag{4}$$

and the likelihood function $L(\mathbf{Y}|\theta)$. In general, one can optimize the parameters by differentiating a cost function (often $-2 \log L$ is used) with respect to each parameter and forming a 'Hessian' Matrix $\underline{\underline{\mathbf{H}}}$ [1] out of the resulting system of equations such that.

$$\underline{\underline{\mathbf{H}}}\theta = \mathbf{c}(\mathbf{Y}), \tag{5}$$

where $c(Y)$ is a constant vector dependent only on the observations Y, but not the parameters θ. The parameter vectors are then given by inverting this matrix and rearranging to form

$$\theta = \underline{\underline{\mathbf{H}}}^{-1} \mathbf{c}(\mathbf{Y}). \tag{6}$$

Depending on the order of the ARMA process (the value p and q), the model may be too complex to form a Hessian matrix and solve analytically. In this case here, numerical methods are used to optimize the parameters of the ARMA process used to fit the bike hire data. [2]

5 Results

The ARMA model is trained on the history of bike hire time series observations (Figure 1) up to the 31st August 2017. The remaining entries in the csv file serve only as a bench mark test data set to test the models's accuracy. In Figure xx, I first use the entire of the 2017

[1] Double underline here means a N by N matrix where N is just the number of model parameters

[2] Various texts exist for numerical optimization of ARMA parameters (see for example http://www.phdeconomics.sssup.it/documents/Lesson12.pdf).

entries as the test data set to illustrate the predictive power of the ARMA model (in other words the model sees no data from 2017). Figure xx shows that the RNN is able to forecast the entire frequency of bike hires for the whole of 2017 and is able to replicate many of the observed features quite remarkably.

Given the objective here is to predict the freqwuency of bike hires only between stations xx and xx, one could train the RNN using only entries between these stations. However, Figure xx shows that this would throw out almost all of the data and leave only a very small number of samples on which to train the network.