To perform the analysis for the assignment, I am using Python 3.7.2 along with the Pandas, NumPy, and Matplotlib libraries. I'm using fairly ubiquitous libraries here to (hopefully) keep my analysis simple to reproduce. To run the analysis, simply run 'python windResource.py' at the terminal.

Question 1

We begin by loading the power curve and met tower data into memory and cleaning negative values from the wind speed data. To extrapolate 80-m wind speeds from the data, we use the wind profile power law:

$$v_2 = v_1\left(\frac{z_2}{z_1}\right)^{\alpha}$$

Since we are fortunate enough to have wind speeds at two different heights (30m and 60m above the surface), we can compute the wind shear coefficient for each data point rather than assuming a fixed value throughout. The wind shear exponent can be quite dynamic, changing considerably throughout the day along with atmospheric stability; so it really is a luxury to have the information available to compute the exponent. Rearranging the power law equation and making use of the properties of logarithms, we solve for the wind shear coefficient:

$$\frac{v_2}{v_1} = \left(\frac{z_2}{z_1}\right)^{\alpha} \text{ -> } ln\left(\frac{v_2}{v_1}\right) = ln\left[\left(\frac{z_2}{z_1}\right)^{\alpha}\right] \text{ -> } \alpha = \frac{ln\left(\frac{v_2}{v_1}\right)}{ln\left(\frac{z_2}{z_1}\right)} \text{ -> } \alpha = \frac{ln(v_2) - ln(v_1)}{ln(z_2) - ln(z_1)}$$

Where the wind is zero, ln(0) is negative infinity and so the exponent is undefined. When the wind is zero, we expect our calculated 80-m wind speed to be zero as well (substituting any real value for the exponent in the very first equation makes this clear). Therefore, we replace NaN values in our 80-m wind speeds with zero. One of the benefits of using Pandas is that Pandas inherits much of the elementwise functionality from NumPy, enabling us to avoid ugly for loops both when computing the wind shear coefficient and when implementing the power law, making for cleaner and more efficient code.

Pandas again serves us well when computing the monthly averages of hub-height wind speeds, giving us a very simple mechanism for isolating data from a given month. Getting the monthly averages only takes two lines of code!

	2011 80-m Average Wind Speed (m/s)
January	6.15
February	7.35
March	6.83
April	8.29
May	6.11
June	6.22

	2011 80-m Average Wind Speed (m/s)
July	4.94
August	4.75
September	5.88
October	6.40
November	8.58
December	6.83
Annual	6.52

Question 2

To devise an estimate of what production would have been in September 2011 with 10 of the 1.5-MW turbines, I wrote a generalized function that computes an estimate for any given month. First, we bin 80-m wind speeds from the specified month into 0.25-m/s-wide bins centered on the wind speed values given in PowerCurve,csv. We then convert the absolute counts into percentages, dividing by the total number of available data points in the month. Since the total energy production is the the integral of power across all time in the month, ignoring any missing data would make our estimate too low. We don't have too much missing data for September 2011, but to be thorough, we use the binned percentages as a probability density and multiply by the total number of hours in the month (30 days*24 hours/day) to get the expected number of hours spent in each wind speed bin. Lastly, we multiply the expected hours in each bin by the power at that bin's wind speed from the power curve and sum across all bins to get kilowatt-hours. Multiply by 10 for the 10 turbines and divide by 1000 to convert to megawatt-hours.

September 2011 energy production estimate: **2485.21 MWh**

We could also get more fancy and fit a function to our power curve (maybe a good idea if we're spending a lot of time in the steepest part of the power curve), but this binning method should provide a reasonable estimate. The biggest downside I foresee with fitting a function is that you then must also fit a function for the probability density, introducing a new source of uncertainty and possibly sacrificing any improvements in accuracy fitting a function to the power curve might offer. Would be curious to hear your thoughts on these approaches!

Question 3

In October 2018, we don't see anything too unusual, with things generally remaining near the expected power curve. Fluctuations in air density and static stability are likely sufficient to explain the spread in observed data around the expected curve.

In November, the most noteworthy feature is the mode of higher than rated efficiency. One possible explanation could be a period of colder weather and denser air, maybe combined with more stable, laminar, and less turbulent flow, maximizing the efficiency of the turbines. In the interest of not turning every problem into a nail when my hammer is meteorology, possibly there is an engineering reason for the heightened efficiency. Perhaps a turbine needed repairs and was outfitted with newer, longer or otherwise more efficient blades.

We see a curious tail of low power values in December. The first thing that comes to mind given the time of year is icing on the turbines forcing downtime. Another possibility would be a couple of rogue turbines suffering mechanical issues, or maybe extended curtailment periods when forecasts were poor and the wind farm was overproducing.

Question 4

This question takes me further out of my comfort zone/areas of expertise than the others, but I'll give it a crack…

I see across across all three months, remote meter energy tends to dip below turbine meter energy as we get nearer to capacity. Maybe small percentage line losses between the turbine and the remote meter are becoming magnified at these higher values.

In November, things deviate from the 1:1 line much more than the other two months. I assume the data is hourly-averaged data, in which case some of the larger differences of a couple hundred kWh between the two readings are difficult for me to explain. Perhaps one or both meters were poorly calibrated and readings excessively noisy. A power spectra plot could help identify if this were an issue. Depending on how the data sampling works for the different meters, volatility in the wind could explain some of the differences.

In December, we see some data points where remote meter energy hits a floor at around 750 kWh, remaining at this level even as turbine meter energy drops below 750 kWh. Possibly the remote meter data happened to flat-line for a bit while winds were ramping down.