ACM40080 HW3 Python WRF

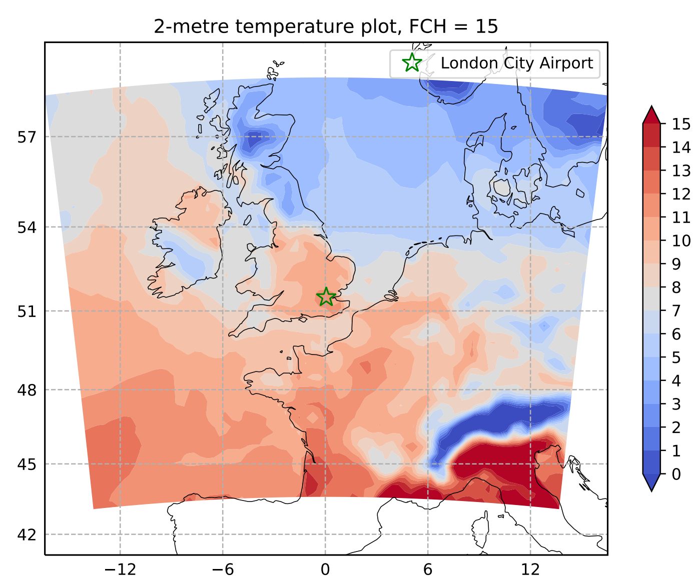
15532943

Forecast on 13th March, 2020 for the United Kingdom

2-Metre Temperature plots

We must first find the location of the nearest grid point to our Weather station at London City Airport, from where we ran our WRF forecast.This is done using 'XLAT' and 'XLONG' from the WRF output file, which detail the Latitude and Longitude of each grid point (at each time too despite being invariant about time). We use 'XLAT' and 'XLONG' to calculate the Euclidean distance between each grid point to the London City Airport weather station which has 51.502 N Latitude and 0.0553 E Longitude. Following this we observe the grid point with the minimum Euclidean distance and take it's the South-North and West-East coordinates, this is the grid point closest to the weather station at London City airport. This grid point has a South-North coordinate of 29 and West-East coordinate of 36. Checking we see that this grid point has a 51.373 N Latitude and 0.0553 E Longitude meaning that it is very close to our airport (within a few kilometres).

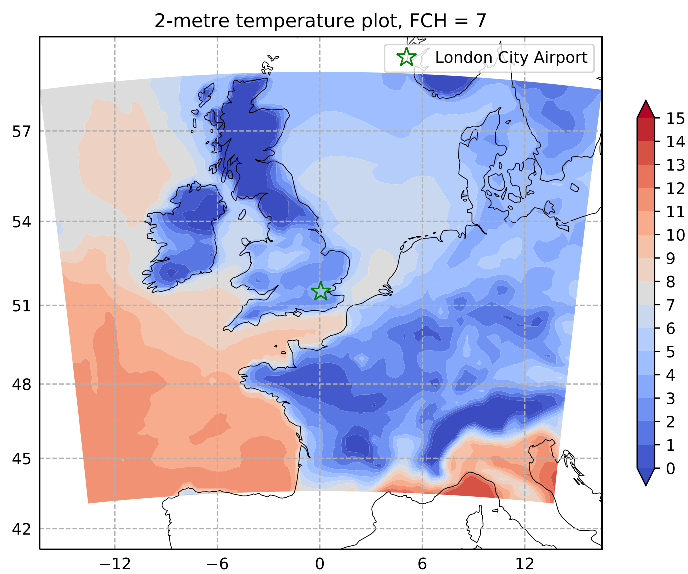
We run our London City Airport weather station forecasts from this grid point. Next we find the hours which give the maximum and minimum 2-m temperatures observed at the airport using the 'np.argwhere', 'np.min' and 'np.max' functions. The maximum temperature at our station occurred at hour 15 and the minimum at hour 7.



Observing the FCH = 15 plot above we see that London City Airport has a 2-metre temperature of around 11 Degrees Celcius. Note that in the code to find the temperature we had to subtract 273.15 in order to shift the units from Kelvin to Degrees Celcius.

We can see that the North Atlantic Ocean is several degrees hotter than the North Sea. This observation fits in well with our understanding of the North Atlantic Drift, where warm water originating from the Gulf of Mexico travels north. More generally we can see as anticipated that the further you move away from the equator the more the temperature tends to drop. This is evidently seen as across the United Kingdom as we have England and Wales are a pinkish-red whereas Scotland is blue.

One aspect that provided a lot of intuitive help when coding is the understanding of the cartopy library and it's functions. When plotting the axes we used Mercator projection introduced by cartographer Gerardus Mercator in 1569 with the input 'projection=ccrs.Mercator()' (Note cartopy.crs was imported as ccrs). The Plate Carreé transformation from the cartopy library is applied to the WRF forecast data to ensure that it plots consistently with the Plate Carreé projection. The Plate Carreé projection is a map projection which allows us to view the surface of the earth in a 2-Dimension visual like in the above and below plots.



Viewing the plot of the lowest temperature at FCH = 7, we see that London City Airport has a 2-metre temperature of around 4 degrees celcius. Compared to the FCH = 15 plot all land regions are much colder, an explanation for this is that it is not yet daytime. This will be confirmed later when we see the time in the UK at this date relative to UTC. It is also worth noticing that the North Atlantic Ocean and North Sea remain a similar temperature to the daytime plot at FCH = 15 and have not cooled as much as the land. We can attribute this to the Ocean being a better insulator of heat.

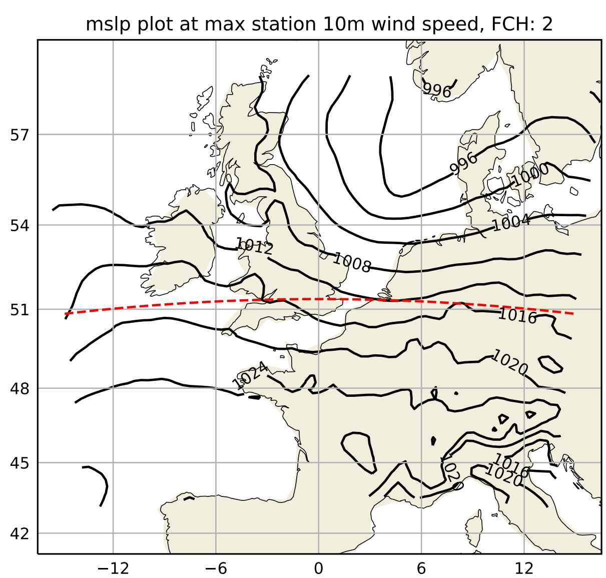
Mean Sea-Level Pressure plot

The hour of our Mean Sea-Level Pressure (mslp) forecast is decided by when the maximum 10-metre wind at our station occurred. As a result our next step is to calculate the 10-metre speed variable at our station.

First we create the new 10-metre wind speed variable which equals $\sqrt{U10^2 + V10^2}$. Then using the 'np.argwhere' and 'np.max' functions similarly to with the previous plot we find that the maximum 10-metre wind speed which occurs at hour 2.

To calculate mslp, we need to use the 'HGT', 'PSFC' and 't2m' variables as outlined in the our notes. mslp is plotted as hPa and contoured every 4 hPa from 900 to 1100. We needed to insert an array of values for the hPa to be matched to. It was easiest to use the 'np.zeros' function to create an array of 51 zeros and then loop through the array assigning values starting with 900 and adding the value 4 to the value being appended every time the loop iterates.

We added a transect through the station by keeping South-North coordinate fixed at 29 fixed and allowing the line to move through the West-East coordinates. As an important aside from a formatting stand point, the trick to colour in the land is to use 'cfeature.LAND'.



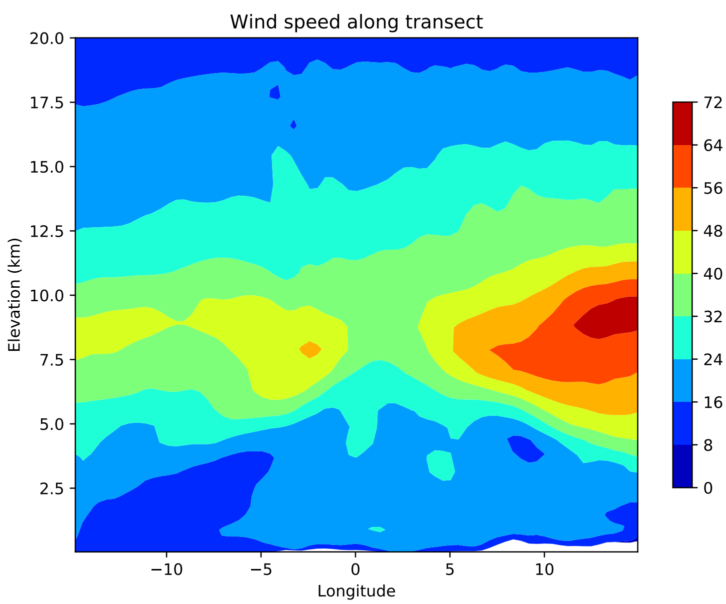
From observing the plot we see areas of low pressure around Norway and Denmark. This tells us that there is more likely a chance of high winds and rain in those regions compared to around England and Ireland where the mslp is greater than 1000 hPa (i.e. where we observe higher pressure).

It is important to note that the different areas of pressure in the atmosphere create wind as wind is air moving from high pressure to low pressure. The quicker the pressure changes from spot to spot, the higher the wind speed is. As a result dense isobars are an indicator of high wind speeds. Dense isobars can be seen in the area above Northern Italy, so we can make an educated guess that the wind speed will be higher in that region.

Wind Transect plot

The wind variables 'U' and 'V' representing the vertical and horizontal elements of the wind speed are stored in Arakawa-style grids. This presents a challenge as we need to unstagger them by finding the values at the centre of the grid for example the 'U' variable stored either side (left and right i.e. West and East) can be used to find the value at the center. We can do similarly with the 'V' variable such that all data points are stored in the same location. Once we have completed this task the wind speed is calculated by $WindSpeed = \sqrt{U^2 + V^2}$.

One difficult aspect of this plot was when 'np.tile' was used to create a 2-Dimension array by replicating a one dimensional array of longitude values. This is an essential step in plotting the wind speed transect as then we can correspond the changing elevation levels to different longitudes. The last code related tip is to change the colour map to 'jet' when plotting to get the below set of coulours.



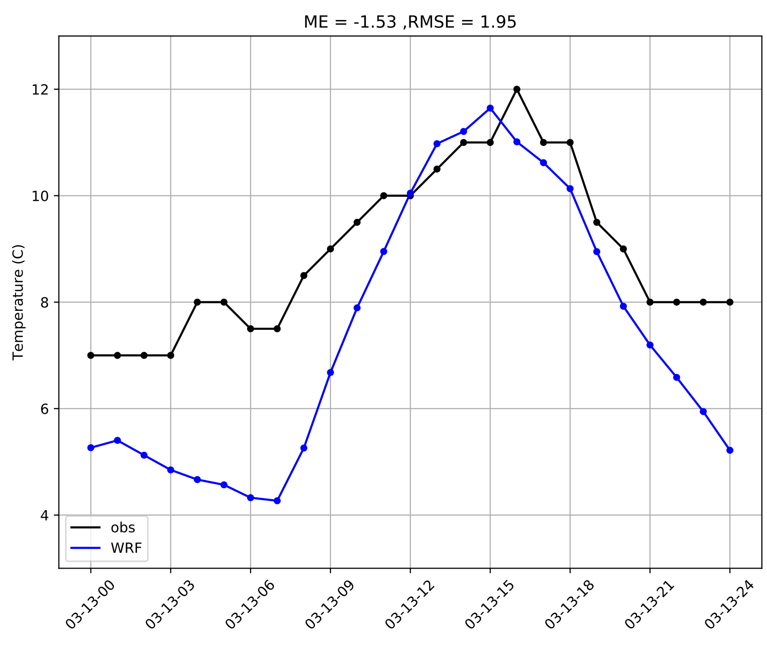
The vertical axis, Elevation (measured in Kilometres) is estimated by using the 'PH' and 'PHB' variables which both serve as approximate values for height.

Note that we do not use Mercator projections for the axis or Plate Carreé transforms for the forecast data because we are not plotting a section of the earth, rather a section of the atmosphere.

By viewing the plot, the maximum wind speeds occur in 5km to 12.5km of Elevation range. Interestingly we can see that the areas of highest speed also occur where the isobars are closest together in the mslp plot (along the transect on the right of the plot), where the Longitude is 5 Degrees and greater. This is a great demonstration of the theory underlying isobars which was outlined in the above section.

We can observe small white areas at the bottom of the curve. These white areas demonstrate how WRF follows the terrain of the land, and are evidence of areas of raised areas of land and mountains. We also observe that the highest wind speeds occur over the mountain beyond 5 degrees Longitude. It may be worth looking into whether the raised land and wid speed is a cause-effect relationship

Error plot



The first step taken in comparing the WRF forecast with the observed data is to ensure we are comparing the same time periods. The WRF forecast is as of UTC. We use a time converter website to check what the time difference in London is to UTC. This was very useful as the website took the date as an input so we knew that it was factoring in daylight savings and other changes of time which may otherwise trip us up. On the 13th March 2020 the time is London was identical to the UTC. This allowed for relative ease in comparing the WRF output with actual observed data.

We will use the 2-metre temperature variable to compare our forecast with the observed data at London City Airport. The London City Airport weather data archives do not store data exactly on the hour, instead data is stored every 30 minutes, at 20 past and 50 minutes past the hour. WRF output for the 2-metre temperature forecast provides a value for every hour on the hour. In order to compare the forecast with the observed data we interpolated the observed weather data by placing a 50% weighting on each of the two data points either side the hour. So for example I found the average between the 07:50 and 08:20 observed values for my 08:00 observation.

Forecast Validation: The Mean Error of forecasts (f) relative to analyses (a) can be defined as $ME = \overline{f - a}$. The Mean Error is not without its flaws as a poor forecast can gain a perfect Mean Error of zero, if large errors have opposite signs and cancel each other out. The ME for our forecast is -1.53. Since it is negative this tells us that we on average over underestimate the observed temperature.

Forecast Verification: We observe the Root Mean Squared Error ($RMSE$) is calculated as $\sqrt{\overline{(f-a)^2}}$ and measure the absolute fit of our weather forecasting model. Our $RMSE$ is 1.95. This signifies that our forecast is relatively accurate in predicting the observed data.

Lastly, we see from viewing the plot that the spike in weather peaking at FCH = 15 is predicted very accurately by our WRF forecast (in blue) however beforehand when the temperature was static at around 7 Degrees Celcius the forecasted values were underestimated. It may be worth observing this as our model may possibly have stronger predictive power during times of change. This is something we should look into further.