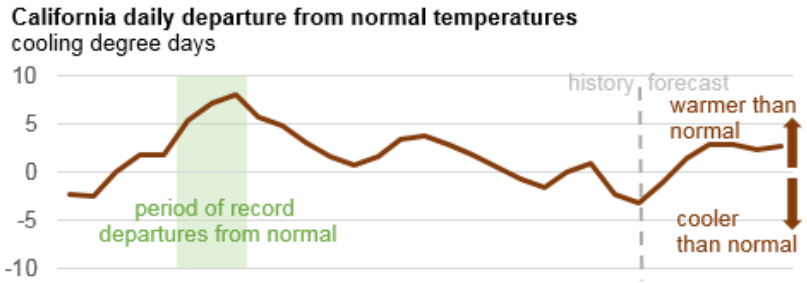
Accessing and mapping NOAA weather data using an interface developed in Python

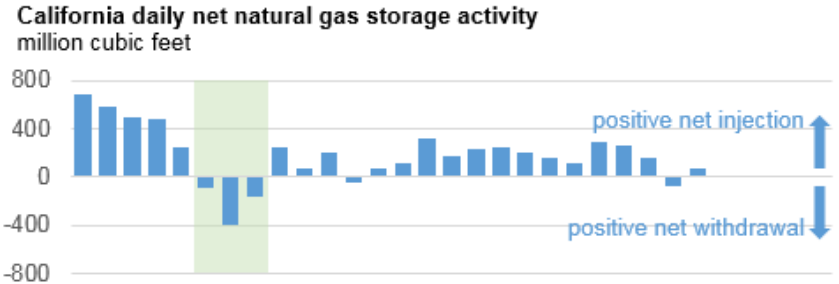
For
EIA Map Users Group
April 17, 2018 | Washington, D.C.

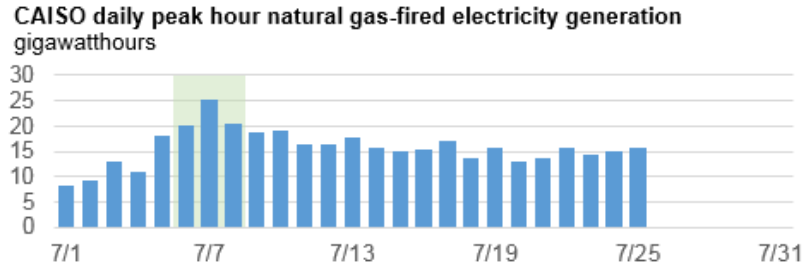
By
Jason Upchurch

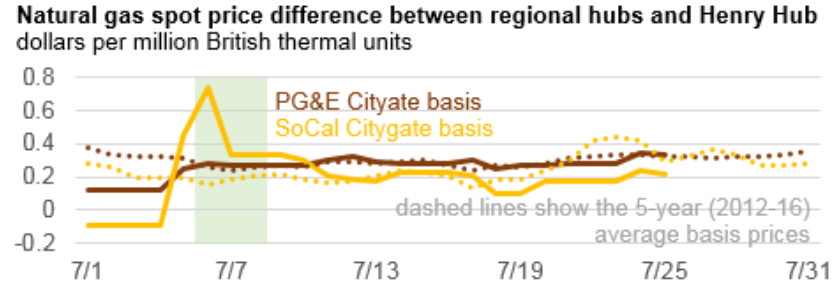
Why are we bothering to map weather?

- Weather is generally a key determinant of changes in electric and natural gas prices and loads







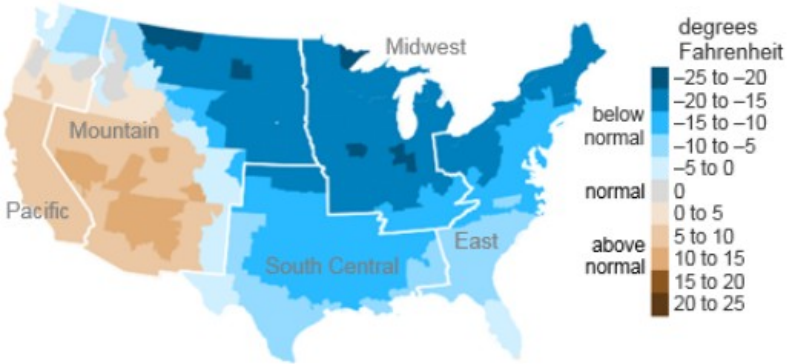


Sources: NOAA (upper left), Ventyx (lower left), PointLogic (upper right), NGI (lower right)

Examples of illustrating weather in our products

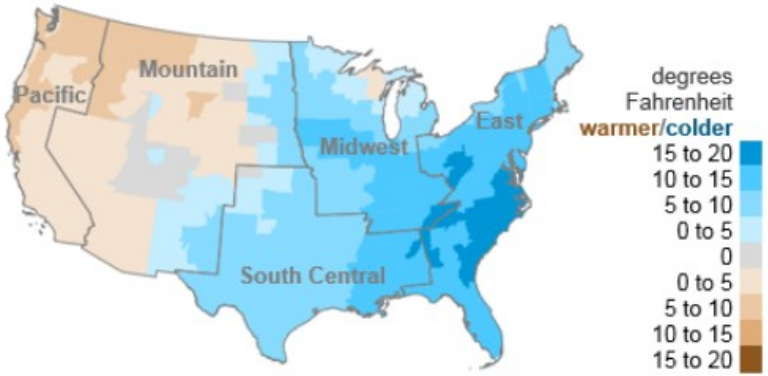
From Natural Gas Weekly Update's *In the News:*
Below normal temperatures last week increased natural gas consumption

Average departure from normal temperatures
Dec 24, 2017–Jan 3, 2018



From Natural Gas Weekly Update's *In the News:*
Working gas stocks post all-time record weekly withdrawal

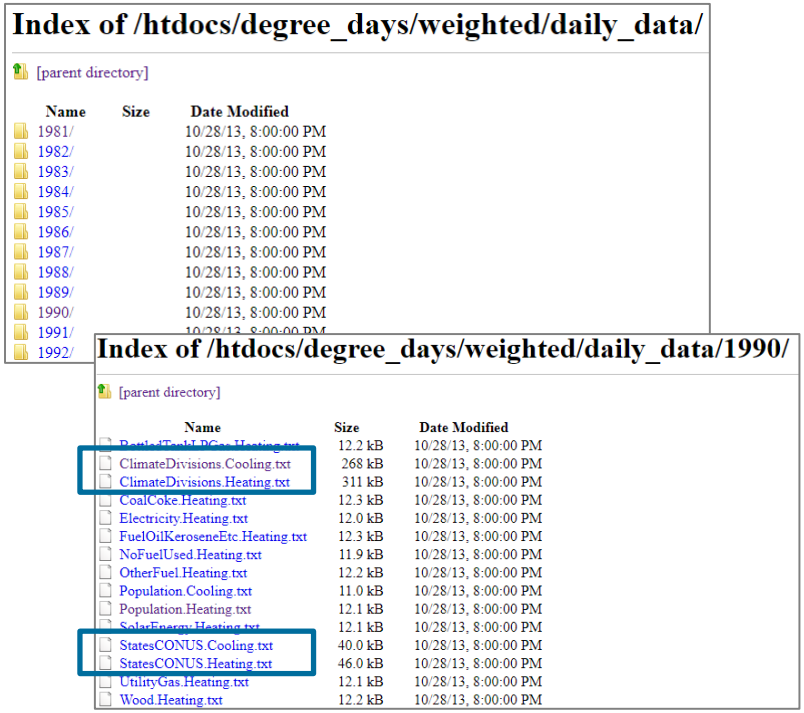
Weekly change in average temperature by EIA storage region
Dec 30, 2017–Jan 5, 2018 vs Dec 23–29



What benefits does the mapping interface offer?

- The NOAA interface can be used as a template out of the box with very little programming experience and no mapping experience required
  - Produces consistent maps with no formatting required
- Or it can be customized by experienced users for a variety of other applications
  - e.g., raw data pulls, adding other templates like mapping cdd or hdd

Problem:
I want to map weather using NOAA data over arbitrary dates



Index of /htdocs/degree_days/weighted/daily_data/

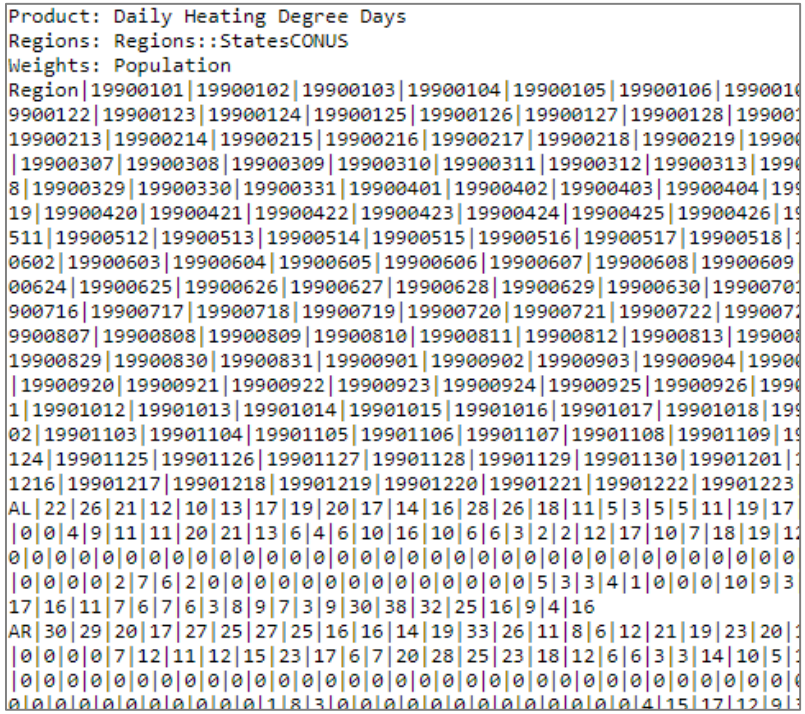
[parent directory]

Name	Size	Date Modified
1981/		10/28/13, 8:00:00 PM
1982/		10/28/13, 8:00:00 PM
1983/		10/28/13, 8:00:00 PM
1984/		10/28/13, 8:00:00 PM
1985/		10/28/13, 8:00:00 PM
1986/		10/28/13, 8:00:00 PM
1987/		10/28/13, 8:00:00 PM
1988/		10/28/13, 8:00:00 PM
1989/		10/28/13, 8:00:00 PM
1990/		10/28/13, 8:00:00 PM
1991/		10/28/13, 8:00:00 PM
1992/		10/28/13, 8:00:00 PM

Index of /htdocs/degree_days/weighted/daily_data/1990/

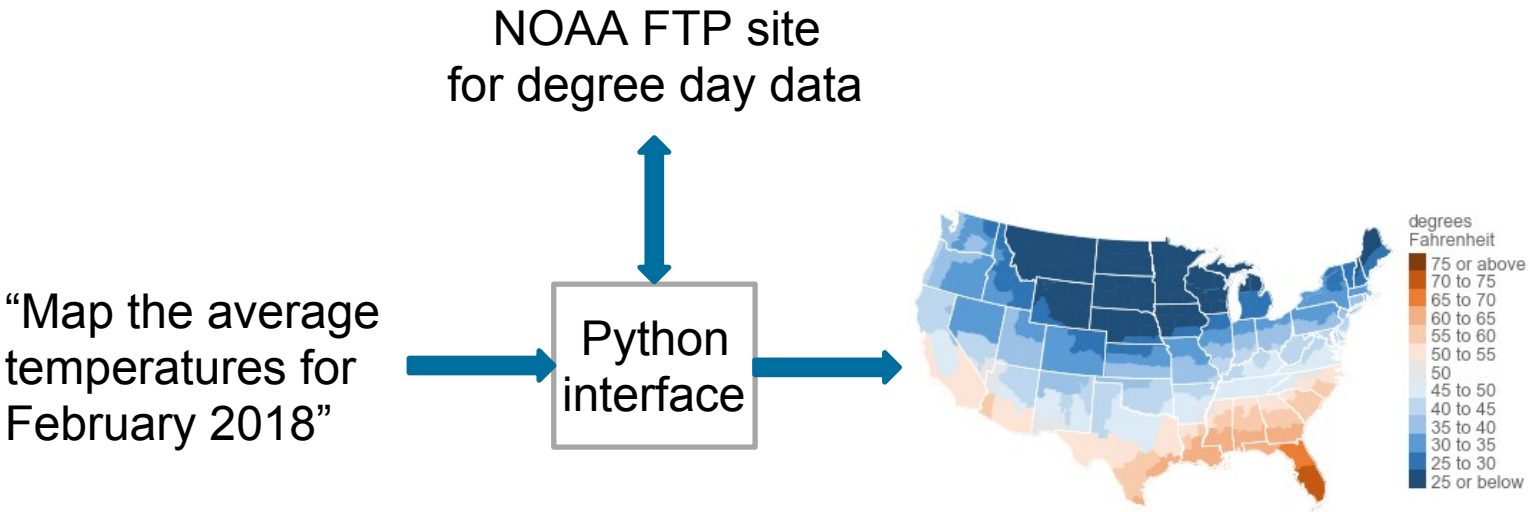
[parent directory]

Name	Size	Date Modified
BottledTankLPGas.Heating.txt	12.2 kB	10/28/13, 8:00:00 PM
ClimateDivisions.Cooling.txt	268 kB	10/28/13, 8:00:00 PM
ClimateDivisions.Heating.txt	311 kB	10/28/13, 8:00:00 PM
CoalCoke.Heating.txt	12.3 kB	10/28/13, 8:00:00 PM
Electricity.Heating.txt	12.0 kB	10/28/13, 8:00:00 PM
FuelOilKeroseneEtc.Heating.txt	12.3 kB	10/28/13, 8:00:00 PM
NoFuelUsed.Heating.txt	11.9 kB	10/28/13, 8:00:00 PM
OtherFuel.Heating.txt	12.2 kB	10/28/13, 8:00:00 PM
Population.Cooling.txt	11.0 kB	10/28/13, 8:00:00 PM
Population.Heating.txt	12.1 kB	10/28/13, 8:00:00 PM
SolarEnergy.Heating.txt	12.1 kB	10/28/13, 8:00:00 PM
StatesCONUS.Cooling.txt	40.0 kB	10/28/13, 8:00:00 PM
StatesCONUS.Heating.txt	46.0 kB	10/28/13, 8:00:00 PM
UtilityGas.Heating.txt	12.1 kB	10/28/13, 8:00:00 PM
Wood.Heating.txt	12.2 kB	10/28/13, 8:00:00 PM

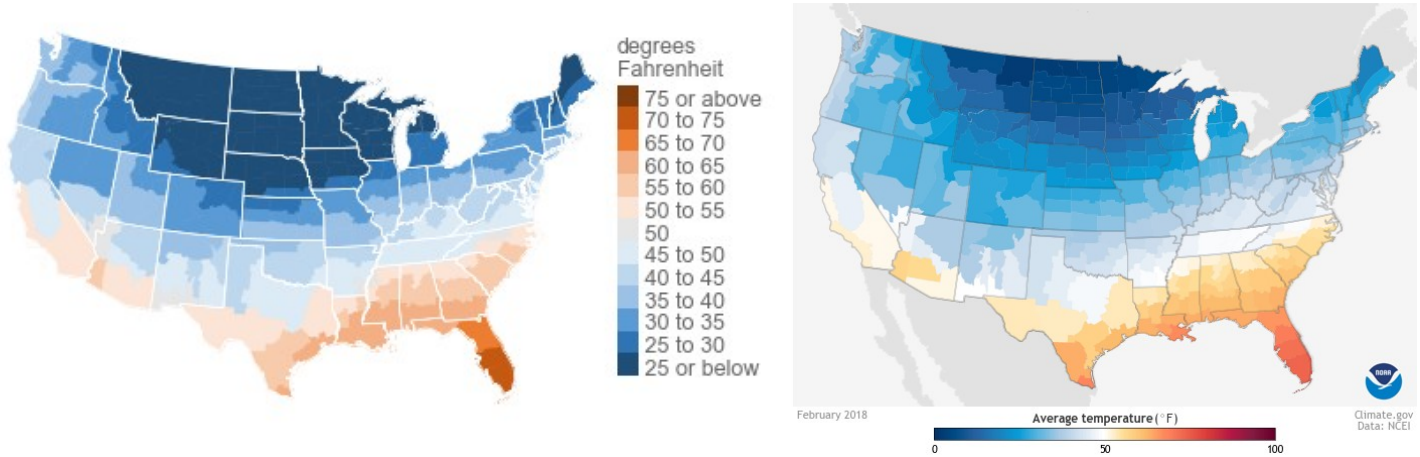


```
Product: Daily Heating Degree Days
Regions: Regions::StatesCONUS
Weights: Population
Region|19900101|19900102|19900103|19900104|19900105|19900106|1990010
9900122|19900123|19900124|19900125|19900126|19900127|19900128|19900
19900213|19900214|19900215|19900216|19900217|19900218|19900219|1990
|19900307|19900308|19900309|19900310|19900311|19900312|19900313|199
8|19900329|19900330|19900331|19900401|19900402|19900403|19900404|19
19|19900420|19900421|19900422|19900423|19900424|19900425|19900426|1
511|19900512|19900513|19900514|19900515|19900516|19900517|19900518|
0602|19900603|19900604|19900605|19900606|19900607|19900608|19900609
00624|19900625|19900626|19900627|19900628|19900629|19900630|1990070
900716|19900717|19900718|19900719|19900720|19900721|19900722|199007
9900807|19900808|19900809|19900810|19900811|19900812|19900813|19900
19900829|19900830|19900831|19900901|19900902|19900903|19900904|1990
|19900920|19900921|19900922|19900923|19900924|19900925|19900926|199
1|19901012|19901013|19901014|19901015|19901016|19901017|19901018|19
02|19901103|19901104|19901105|19901106|19901107|19901108|19901109|1
124|19901125|19901126|19901127|19901128|19901129|19901130|19901201|
1216|19901217|19901218|19901219|19901220|19901221|19901222|19901223
AL|22|26|21|12|10|13|17|19|20|17|14|16|28|26|18|11|5|3|5|5|11|19|17
|0|0|4|9|11|11|20|21|13|6|4|6|10|16|10|6|6|3|2|2|12|17|10|7|18|19|12
0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0
|0|0|0|0|2|7|6|2|0|0|0|0|0|0|0|0|0|0|0|0|0|5|3|3|4|1|0|0|0|10|9|3
17|16|11|7|6|7|6|3|8|9|7|3|9|30|38|32|25|16|9|4|16
AR|30|29|20|17|27|25|27|25|16|16|14|19|33|26|11|8|6|12|21|19|23|20|1
|0|0|0|0|7|12|11|12|15|23|17|6|7|20|28|25|23|18|12|6|6|3|3|14|10|5|1
|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0
0|0|0|0|0|0|0|0|0|0|1|8|3|0|0|0|0|0|0|0|0|0|0|0|0|4|15|17|12|9|3
```

Solution:
the high-level interface requires very little programming experience, no mapping experience, and saves time



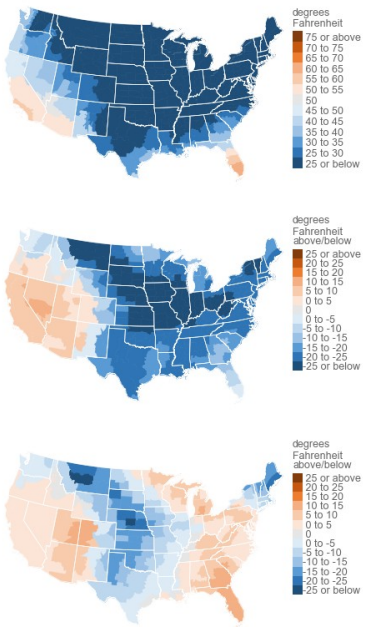
Validating the approach:
a comparison to the monthly NOAA-produced version



Sources: U.S. Energy Information Administration based on NOAA (left); NOAA (right)

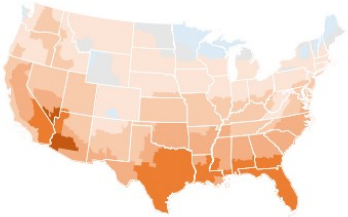
The high-level mapping interface provides three primary functions

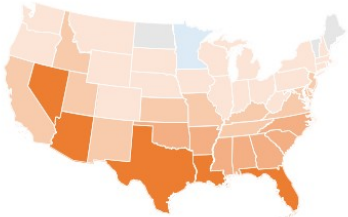
- Map average temperatures
  - What were average temperatures for January 1, 2018?
- Map departure from normal
  - How did temperatures on January 1, 2018 differ from the 1981–2010 normal?
- Map comparison of average temperatures
  - How did temperatures on January 1, 2018 differ from January 7, 2014?



What does the user have control over in the main interface?

- Resolution
  - State climate division (default)
  - State

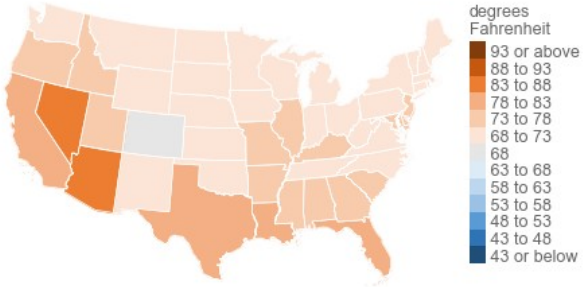




- Intervals of time
  - Single day or ranges of days/years
  - Definition of *normal*, e.g., previous 10 years, NOAA-defined 30-year normal 1981-2010
- Whether or not to include a legend
- Whether or not to save the map and the underlying data

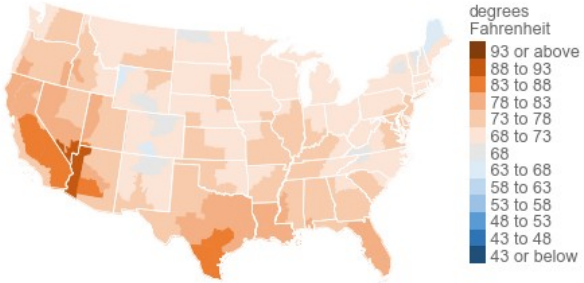
Example 1

```
'''
map average temps
'''
email = 'jason.upchurch@eia.gov'
date_start = datetime(2017,8,1).date()
date_end = datetime(2017,8,31).date()
plt_average_temps = map_average_temps(
          date_start, date_end, email,split_on = 68, how= 'state', legend=True, savepath=False)
plt_average_temps.show()
```



Example 2

```
'''
map average temps
'''
email = 'jason.upchurch@eia.gov'
date_start = datetime(2017,8,1).date()
date_end = datetime(2017,8,1 ).date()
plt_average_temps = map_average_temps(
          date_start, date_end, email,split_on = 68, how ='stcd', legend=True, savepath=False)
plt_average_temps.show()
```



How do I get started mapping in Python?

- Obtain a distribution of Python
  - Anaconda by Continuum has a lot of additional libraries included, and includes an integrated development environment (Spyder)
- Install Cartopy
  - Installing cartopy from conda-forge using the command line should include all the dependent packages, e.g., geos, proj4, shapely
- Copy the Python interface files into a new Python project

Additional resources

- Python documentation:
  - https://www.python.org/doc/
- Cartopy documentation:
  - http://scitools.org.uk/cartopy/
- Help using the interface or bug reporting:
  - jason.upchurch@eia.gov