Code Appendix

```python
[3]: import pandas as pd
     import matplotlib.pyplot as plt
     import numpy as np
     pd.set_option('display.max_columns', None)
     pd.set_option('display.max_rows', 50)
```

```python
[4]: data = pd.read_stata('matched.dta')
```

```python
[5]: temp = ['l2avgtemp', 'l3avgtemp', 'lavgtemp', 'temp6t4', 'le1avgtemp',
     →'le2avgtemp',
             'le3avgtemp', 'le3temp6t4', 'le2temp6t4', 'letemp6t4', 'ltemp6t4',
     →'l2temp6t4',
             'l3temp6t4', 'avgtemp', 'yearaftertemp', 'gtemp', 'yeartemp', 'heat']

     for col in temp:
         new_col = col + str(10)
         data[new_col] = data[col]/10


     weatherdaily =  ['avgtemp10', 'skycover', 'pressureavgsealevel', 'windspeed',
     →'precipitationwaterequiv', 'avgdewpt']
     weatherdailyt = ['skycover', 'avgdewpt', 'pressureavgsealevel', 'windspeed',
     →'precipitationwaterequiv']
     weathertemp =   ['press6t4', 'dew6t4', 'prcp6t4', 'wind6t4', 'skycover']
     weather6t4 =    ['temp6t410', 'press6t4', 'dew6t4', 'prcp6t4', 'wind6t4',
     →'skycover' ]
     heat =          ['heat10', 'press6t4', 'prcp6t4', 'wind6t4', 'skycover']
     dailyheat =     ['dailyheat', 'skycover', 'pressureavgsealevel', 'windspeed',
     →'precipitationwaterequiv' ]
     dummies =       ['dayofweek', ' nati', 'type', 'year', 'cm', 'chair']
     pollutants =    ['ozone', 'co', 'pm25']
```

```python
[ ]: dropped_data = data.copy().dropna(how = 'any', subset = weather6t4 + pollutants)
```

```python
[7]: sum_stat_columns = ['res', 'temp6t4', 'heat'] + weathertemp + pollutants
     mean = dropped_data[sum_stat_columns].mean()
     sd = dropped_data[sum_stat_columns].std()
```

```
new_index = ['Grant Indicator',
             'Temperature ($^{\circ}$F)',
             'Heat Index ($^{\circ}$F)',
             'Air Pressure (pa)',
             'Dew point ($^{\circ}$F)',
             'Precipitation (mm)',
             'Wind speed (km/h)',
             'Sky cover (percent)',
             'Ozone (ppm)',
             'CO (ppm)',
             'PM$_{2.5}$ ($\mu$ / $m^3$)']
```

[8]:
```
sum_stat_table = pd.DataFrame({'Mean':mean.values, 'Standard Deviation':sd.
  ↪values}, index = new_index)
```

[9]:
```
sum_stat_table
```

[9]:

	Mean	Standard Deviation
Grant Indicator	0.164261	0.370854
Temperature ($^{\circ}$F)	57.370564	15.720450
Heat Index ($^{\circ}$F)	57.776386	16.411543
Air Pressure (pa)	29.692633	0.759302
Dew point ($^{\circ}$F)	49.371086	17.202080
Precipitation (mm)	0.002822	0.013569
Wind speed (km/h)	4.556642	3.440875
Sky cover (percent)	0.554441	0.275636
Ozone (ppm)	0.021971	0.011932
CO (ppm)	0.917596	0.496442
PM$_{2.5}$ (μ / m^3)	14.956901	11.570034

[12]:
```
paper_fig3, ax = plt.subplots(figsize = (10,10))
ax.grid(axis = 'y', zorder=0)
ax.hist(dropped_data['temp6t4'], bins = np.arange(0,110,10), color =
  ↪'lightgrey', edgecolor = 'black', zorder=3)
ax.set_xlabel("Temperature ($^\circ$F)")
ax.set_xticks(np.arange(5,100,10))
ax.set_xticklabels(['<10'] + [str(i) + '-' + str(i+10) for i in range(10, 90,
  ↪10)] + ['>90'])
ax.set_ylabel("Counts")
ax.set_ylim([0, 65000]);
paper_fig3.savefig('paper_fig3.jpg')
```