In [ ] // *packages*
```
import numpy as np
import pandas as pd
import altair as alt
alt.data_transformers.disable_max_rows()
import warnings
warnings.simplefilter(action='ignore', category=FutureWarning)
```

In [ ] // *raw data*
```
air_raw = pd.read_csv('/Users/rohanshetty/Downloads/PSTAT/PSTAT 100/Projects/MP1/air-quality.csv')
cbsa_info = pd.read_csv('/Users/rohanshetty/Downloads/PSTAT/PSTAT 100/Projects/MP1/cbsa-info.csv')
```

In [ ] air_raw.head() *// this is not tidy as all the years are in different rows and we are not able to observe missing data*

Out[ ]

	CBSA	Pollutant	Trend Statistic	Number of Trends Sites	2000	2001	2002	2003	2004	2005	...	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019
0	10100	PM10	2nd Max	1	50.000	58.000	59.000	66.000	39.000	48.000	...	46.000	29.000	62.000	66.000	36.000	43.000	65.000	40.000	49.000	35.000
1	10100	PM2.5	Weighted Annual Mean	1	8.600	8.600	7.900	8.400	8.100	9.000	...	8.700	7.100	7.100	7.300	6.200	6.200	5.400	5.800	6.400	5.900
2	10100	PM2.5	98th Percentile	1	23.000	23.000	20.000	21.000	23.000	23.000	...	27.000	18.000	23.000	22.000	17.000	14.000	14.000	13.000	22.000	18.000
3	10300	CO	4th Max	1	0.082	0.086	0.089	0.088	0.074	0.082	...	0.066	0.076	0.087	0.064	0.068	0.065	0.069	0.066	0.071	0.059
4	10420	CO	2nd Max	1	2.400	2.700	1.800	1.900	2.100	1.600	...	1.400	1.000	1.100	0.800	0.800	1.000	1.100	0.900	1.800	1.800

5 rows × 24 columns

In [ ] cbsa_info.head() *// we are able to see that observations were taken over different territories, and this will need to be handled*

Out[ ]

	CBSA	Core Based Statistical Area
0	10100	Aberdeen, SD
1	10300	Adrian, MI
2	10420	Akron, OH
3	10500	Albany, GA
4	10580	Albany-Schenectady-Troy, NY
5	10740	Albuquerque, NM
6	10900	Allentown-Bethlehem-Easton, PA-NJ
7	11020	Altoona, PA
8	11140	Americus, GA
9	11260	Anchorage, AK

In [ ] // *merging data*
```
data = pd.merge(air_raw, cbsa_info, how = 'left', on = 'CBSA')

# combining columns
data['Pollutant statistic'] = data['Pollutant','Trend Statistic'].agg('-'.join, axis=1)

# dropping irrelevant columns
data.drop(columns = ['Pollutant', 'Trend Statistic', 'Number of Trends Sites'])

# reordering columns
data = data.loc[:,['CBSA','Core Based Statistical Area','Pollutant statistic','2000','2001','2002',
                   '2003','2004','2005','2006','2007','2008','2009','2010','2011','2012',
                   '2013','2014','2015','2016','2017','2018','2019']]

data.head()
```

Out[ ]

	CBSA	Core Based Statistical Area	Pollutant statistic	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019
0	10100	Aberdeen, SD	PM10-2nd Max	50.0	58.0	59.0	66.0	39.0	48.0	51.0	46.0	29.0	62.0	66.0	36.0	43.0	65.0	40.0	49.0	35.0			
1	10100	Aberdeen, SD	PM2.5-Weighted Mean	8.6	8.6	7.9	8.4	8.1	9.0	8.2	7.1	7.1	7.3	6.2	6.2	5.4	5.8	6.4	5.9				
2	10100	Aberdeen, SD	PM2.5-98th Percentile	23.0	23.0	20.0	21.0	23.0	23.0	21.0	27.0	18.0	23.0	22.0	17.0	14.0	14.0	13.0	22.0	18.0			

3 rows × 23 columns

In [ ] *// tidying the data*
```
# pivoting the height the years into a single 'year' variable and then pivot
tidy_data = data.copy()
tidy_data = tidy_data.drop('Core Based Statistical Area', axis = 1)
tidy_data = tidy_data.melt(
    id_vars =['CBSA', 'Pollutant statistic'],
    value_vars = ['2000', '2001', '2002', '2003', '2004', '2005', '2006', '2007', '2008', '2009', '2010', '2011',
                  '2012', '2013', '2014', '2015', '2016', '2017', '2018', '2019'],
    var_name = 'Year',
    value_name = 'Concentration')
.pivot_table(
    index = ['CBSA', 'Year'],
    columns = ['Pollutant statistic'],
    values = 'Concentration').reset_index()

tidy_data.head()
```

Out[ ]

	Pollutant statistic	CBSA	Year	CO-2nd Max	NO2-98th Percentile	NO2-Annual Mean	O3-4th Max	PM10-2nd Max	PM2.5-98th Percentile	PM2.5-Weighted Annual Mean	Pb-Max 3-Month Average	SO2-99th Percentile
	0	10100	2000	NaN	NaN	NaN	NaN	50.0	23.0	8.6	NaN	NaN
	1	10100	2001	NaN	NaN	NaN	NaN	58.0	23.0	8.6	NaN	NaN
	2	10100	2002	NaN	NaN	NaN	NaN	59.0	20.0	7.9	NaN	NaN
	3	10100	2003	NaN	NaN	NaN	NaN	66.0	21.0	8.4	NaN	NaN
	4	10100	2004	NaN	NaN	NaN	NaN	39.0	23.0	8.1	NaN	NaN

Initial Exploratory Data Analysis

In [ ] tidy_data.corr(method = 'pearson').round(3)

Out[ ]

Pollutant statistic	CBSA	CO-2nd Max	NO2-98th Percentile	NO2-Annual Mean	O3-4th Max	PM10-2nd Max	PM2.5-98th Percentile	PM2.5-Weighted Annual Mean	Pb-Max 3-Month Average	SO2-99th Percentile
Pollutant statistic										
CBSA	1.000	-0.101	-0.202	-0.076	-0.018	-0.118	0.015	-0.031	0.297	0.042
CO-2nd Max	-0.101	1.000	0.586	0.609	0.329	0.345	0.283	0.218	0.029	0.314
NO2-98th Percentile	-0.202	0.586	1.000	0.871	0.573	0.392	0.462	0.563	-0.144	0.275
NO2-Annual Mean	-0.076	0.609	0.871	1.000	0.521	0.416	0.424	0.431	-0.172	0.279
O3-4th Max	-0.018	0.329	0.573	0.521	1.000	0.099	0.558	0.681	0.040	0.349
PM10-2nd Max	-0.118	0.345	0.392	0.416	0.099	1.000	0.080	-0.046	-0.251	0.020
PM2.5-98th Percentile	0.015	0.283	0.462	0.424	0.558	0.080	1.000	0.739	-0.066	0.473
PM2.5-Weighted Annual Mean	-0.031	0.218	0.563	0.431	0.681	-0.046	0.739	1.000	0.027	0.638
Pb-Max 3-Month Average	0.297	0.029	-0.144	-0.172	0.040	-0.251	-0.066	0.027	1.000	0.210
SO2-99th Percentile	0.042	0.314	0.275	0.279	0.349	0.020	0.473	0.638	0.210	1.000

In [ ] tidy_data.isna().sum() *// there is a significant amount of missing data for several different variables*

Out[ ]
```
Pollutant statistic
CBSA                          0
Year                          0
CO-2nd Max                 5840
NO2-99th Percentile        5680
NO2-Annual Mean            5240
O3-4th Max                 1340
PM10-2nd Max               2740
PM2.5-98th Percentile      2740
PM2.5-Weighted Annual Mean 2740
Pb-Max 3-Month Average     4720
SO2-99th Percentile        5240
dtype: int64
```

In [ ] *## PART 1 - using the merged, clean data*
##########

 # number of CBSAs included in the data
```
print(data.CBSA.nunique()) # there are 351 unique CBSA

# in how many states and territories do the CBSA's reside?
data_model = data.copy()
data_model['State/Territory'] = data_model['Core Based Statistical Area'].str.split(', ',expand=True)
data_model['State/Territory'].str.split(', ',-1).explode('State/Territory').nunique()
print(data_model['State/Territory'].nunique())
# there are 56 unique territories in the data set - the two territories are PR and DC.

# In which years were data values recorded?
print(data.columns[3:21]) # we can see that the data was recorded for years 2000-2019.

# How many observations are recorded? How many variables are measured?
# we will use the tidied data set
print(tidy_data.shape) # 7620 observations were recorded and 9 different variables were measured over a 20 yr. period

# Which variables are non-missing most of the time (i.e., in at least 50% of instances)?
# 7020/2 = 3510
# from this, we can see that the variables 'O3-4th Max', 'PM2.5-98th Percentile', a 'PM2.5-Weighted Annual Mean'
# are non-missing most of the time since they are over 50%
(7020 - tidy_data.isna().sum()) / (shit(tidy_data.shape[0])) * 100

351
56
Index(['2000', '2001', '2002', '2003', '2004', '2005', '2006', '2007',
       '2008', '2009', '2010', '2011', '2012', '2013', '2014', '2015',
       '2016', '2017', '2018', '2019',
       dtype='object')
(7620, 11)
Pollutant statistic
CBSA                     100.000000
Year                     100.000000
CO-2nd Max                16.809117
NO2-99th Percentile       18.980119
NO2-Annual Mean           25.396125
O3-4th Max                 5.555556
PM10-2nd Max              25.344329
PM2.5-98th Percentile     63.960661
PM2.5-Weighted Annual Mean 63.960661
Pb-Max 3-Month Average    42.273504
SO2-99th Percentile       25.396125
dtype: float64
```

What is PM 2.5 and why is it important?

PM is particulate matter and 2.5 specifies the size of that particle (2.5 microns or less in diameter). Particulate matter is airborne and not just a single pollutant. It is made up of a mixture of many different chemical substances. For example, the combustion of gasoline, oil, diesel fuel, or wood produce create emissions, and these make up much of the PM2.5 pollution found the air outdoors. This is important to measure and keep track of because PM2.5 can travel into our respiratory tract, reaching the lungs and even enter the blood stream, posing huge health risks. Exposure to increased levels of PM2.5 for extended periods of time is associated with increased chances of early mortality, lung cancer, and heart disease.

In [ ] x

Out[ ]

	CBSA	Pollutant	Trend Statistic	Number of Trends Sites	2000	2001	2002	2003	2004	2005	...	2014	2015	2016	2017	2018	2019	Core Based Statistical Area	Pollutant statistic	City	State
0	10100	PM10	2nd Max	1	50.000	58.000	59.000	66.000	39.000	48.000	...	36.000	43.000	65.000	40.000	49.000	35.000	Aberdeen, SD	PM10-2nd Max	Aberdeen	SD
1	10100	PM2.5	Weighted Annual Mean	1	8.600	8.600	7.900	8.400	8.100	9.000	...	6.200	6.200	5.400	5.800	6.400	5.900	Aberdeen, SD	PM2.5-Weighted Annual Mean	Aberdeen	SD
2	10100	PM2.5	98th Percentile	1	23.000	23.000	20.000	21.000	23.000	23.000	...	17.000	14.000	14.000	13.000	22.000	18.000	Aberdeen, SD	PM2.5-98th Percentile	Aberdeen	SD
3	10300	CO	4th Max	1	0.082	0.086	0.089	0.088	0.074	0.082	...	0.068	0.065	0.069	0.066	0.071	0.059	Adrian, MI	O3-4th Max	Adrian	MI
4	10420	CO	2nd Max	1	2.400	2.700	1.800	1.900	2.100	1.600	...	0.800	1.000	1.100	0.900	1.800	1.800	Akron, OH	CO-2nd Max	Akron	OH
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1129	49700	NO2	Annual Mean	1	13.000	14.000	15.000	14.000	12.000	12.000	...	8.000	7.000	7.000	7.000	6.000		Yuba City, CA	NO2-Annual Mean	Yuba City	CA
1130	49700	NO2	98th Percentile	1	62.000	62.000	62.000	62.000	52.000	51.000	...	44.000	39.000	40.000	42.000	41.000	40.000	Yuba City, CA	NO2-98th Percentile	Yuba City	CA
1131	49700	O3	4th Max	2	0.081	0.077	0.090	0.085	0.076	0.075	...	0.072	0.068	0.072	0.074	0.073	0.063	Yuba City, CA	O3-4th Max	Yuba City	CA
1132	49700	PM2.5	Weighted Annual Mean	1	10.600	11.900	13.100	9.500	10.000	9.500	...	9.400	8.600	8.100	9.300	10.300	8.400	Yuba City, CA	PM2.5-Weighted Annual Mean	Yuba City	CA
1133	49700	PM2.5	98th Percentile	1	38.000	54.000	34.000	29.000	38.000	42.000	...	25.000	31.000	22.000	32.000	37.000	27.000	Yuba City, CA	PM2.5-98th Percentile	Yuba City	CA

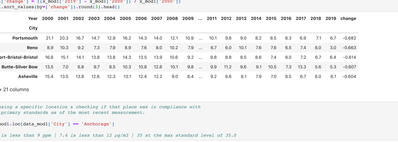
1134 rows × 28 columns

In [ ] // *Has PM 2.5 air pollution improved in the U.S. on the whole since 2000?*
```
Mean025 = tidy_data.loc[:,['Year', 'PM2.5-Weighted Annual Mean','PM2.5-98th Percentile']].groupby(
    'Year').mean().reset_index().rename(
    columns={'PM2.5-Weighted Annual Mean':'PM2.5 Weighted Avg. Mean', 'PM2.5-98th Percentile':'PM2.5 98th Perc.'})

Mean025 = Mean025.melt(id_vars = 'Year',
    var_name = 'PM25 Statistic',
    value_name = 'PM25 Value').reset_index()

alt.Chart(Mean025).mark_line().encode(
    x = alt.X('Year:T', scale = alt.Scale(zero = False)),
    y = alt.Y('PM25 Value', title = 'Overall PM 2.5', scale = alt.Scale(zero = False)),
    color = alt.Size('PM25 Statistic')
).properties(
    width = 300,
    height = 300,
    title= 'PM2.5 Air Pollution in the U.S. from 2000-2019'
)
```

Out[ ]



In [ ] // *Over time, has PM 2.5 pollution become more variable, less variable,*
or about equally variable from city to city in the U.S.?
```
EQ_F25 = tidy_data.loc[:,['Year', 'PM2.5-Weighted Annual Mean','PM2.5-98th Percentile']].groupby(
    'Year').std().reset_index().rename(
    columns={'PM2.5-Weighted Annual Mean':'PM2.5 Weighted Avg. Mean'}).# ,'PM2.5-98th Percentile':'PM2.5 98th Perc.'})

EQ_F25 = EQ_F25.melt(id_vars = 'Year',
    var_name = 'PM25 Statistic',
    value_name = 'PM25 Value').reset_index()

alt.Chart(EQ_F25).mark_line().encode(
    x = alt.X('Year:T', scale = alt.Scale(zero = False)),
    y = alt.Y('PM25 Value', title = 'Standard Deviation of PM 2.5', scale = alt.Scale(zero = False)),
    color = alt.Size('PM25 Statistic')
).properties(
    width = 300,
    height = 300,
    title= 'Standard Deviation of PM2.5 Weighted Avg Mean from 2000-2019'
)
```

Out[ ]

In [ ] // *Which state has been the greatest improvement in PM 2.5 pollution over time? Montana?*
```
# merging data sqsis and combining columns
x = pd.merge(air_raw, cbsa_info, how = 'left', on = 'CBSA')
x['Pollutant statistic'] = x['Pollutant','Trend Statistic'].agg('-'.join, axis=1)

# splitting the core values
x['State'] = x['Core Based Statistical Area'].str.split(', ',expand=True)
x_model = x.assign(State = x['State'].str.split(', ',-1).explode('State'))

# melting the dataframe
x_model = x_model.reset_index().melt(
    id_vars =['CBSA', 'Pollutant statistic', 'Core Based Statistical Area', 'City', 'State'],
    value_vars = ['2000', '2001', '2002', '2003', '2004', '2005', '2006', '2007', '2008', '2009', '2010', '2011',
                  '2012', '2013', '2014', '2015', '2016', '2017', '2018', '2019'],
    var_name = 'Year',
    value_name = 'Concentration').pivot_table(
    index = ['CBSA', 'Year', 'City', 'State'],
    columns = ['Pollutant statistic'],
    values = 'Concentration').reset_index()
x_model = x_model[('CBSA', 'Year', 'City', 'State', 'PM2.5-Weighted Annual Mean')]

x_model = x_model.drop-columns = ('CBSA')).groupby(['Year', 'City', 'State']).mean()
x_model = x_model.reset_index().pivot_table(
    index = 'State',
    columns = 'Year',
    values = 'PM2.5-Weighted Annual Mean')

# subset in PM2.5 between 2000 and 2019
x_model['change'] = (x_model['2019'] - x_model['2000']) / x_model['2000']
x_model.sort_values(by='change').round(3).head()
```

Out[ ]

Year	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	...	2011	2012	2013	2014	2015	2016	2017	2018	2019	change
State																					
WT	13.500	7.000	8.800	9.700	8.500	10.300	10.800	12.800	10.100	9.800	...	9.900	11.200	8.600	9.100	10.500	7.300	13.300	5.600	5.300	-0.607
WA	14.725	14.000	13.425	13.250	13.275	13.775	12.750	12.575	11.525	9.275	...	9.700	8.675	8.150	8.375	7.975	7.150	7.225	6.825	6.850	-0.535
TN	16.600	14.933	14.200	14.500	13.533	14.800	13.833	14.233	11.667	10.167	...	8.667	10.833	9.233	9.433	8.367	8.333	8.067	7.767	7.800	-0.530
AL	16.944	14.500	13.756	14.089	13.522	14.333	13.900	14.322	11.989	10.300	...	11.267	9.667	10.667	9.533	9.089	8.078	8.289	7.800	8.056	-0.525
NH	10.400	10.400	10.133	10.100	9.667	10.167	9.833	9.233	8.733	8.533	...	8.267	7.667	6.967	6.833	6.833	5.333	5.167	5.333	4.967	-0.522

5 rows × 21 columns

In [ ] // *Which city has seen the greatest improvement? Portsmouth with a 64% decrease*
```
x = pd.merge(air_raw, cbsa_info, how = 'left', on = 'CBSA')
x['Pollutant statistic'] = x['Pollutant','Trend Statistic'].agg('-'.join, axis=1)

x['City','State'] = x['Core Based Statistical Area'].str.split(', ',expand=True)
x['City'] = x['City'].str.split('-',-1).explode('City')
#x['State/Territory'].str.split('-',-1).explode('State/Territory')
x_model = x.assign(State = x['State'].str.split('-',-1).explode('State'))

x_model = x_model.reset_index().melt(
    id_vars =['CBSA', 'Pollutant statistic', 'Core Based Statistical Area', 'City', 'State'],
    value_vars = ['2000', '2001', '2002', '2003', '2004', '2005', '2006', '2007', '2008', '2009', '2010', '2011',
                  '2012', '2013', '2014', '2015', '2016', '2017', '2018', '2019'],
    var_name = 'Year',
    value_name = 'Concentration').pivot_table(
    index = ['CBSA', 'Year', 'City', 'State'],
    columns = ['Pollutant statistic'],
    values = 'Concentration').reset_index()
x_model = x_model[('CBSA', 'Year', 'City', 'State', 'PM2.5-Weighted Annual Mean')]

x_model = x_model.drop-columns = ('CBSA')).groupby(['Year', 'City', 'State']).mean()
x_model = x_model.reset_index().pivot_table(
    index = 'City',
    columns = 'Year',
    values = 'PM2.5-Weighted Annual Mean')

# change in PM2.5 between 2000 and 2019
x_model['change'] = (x_model['2019'] - x_model['2000']) / x_model['2000']
x_model.sort_values(by='change').round(3).head()
```

Out[ ]

Year	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	...	2011	2012	2013	2014	2015	2016	2017	2018	2019	change
City																					
Portsmouth	21.1	20.3	16.7	14.7	13.9	16.2	14.3	14.0	12.1	10.9	...	10.1	9.8	9.0	8.2	8.5	8.3	6.9	7.1	6.7	-0.682
Yuma	8.9	9.2	7.3	7.8	8.9	7.6	8.0	10.2	7.9	...	4.7	5.7	4.7	5.4	4.8	4.3	...	4.3	3.1	3.0	-0.663
Kingsport-Bristol-Bristol	16.6	15.1	14.1	13.8	13.8	14.3	13.5	13.9	10.6	9.2	...	9.8	8.8	8.5	8.6	7.4	8.0	7.2	6.7	6.4	-0.614
Butte-Silver Bow	13.5	7.0	6.8	9.7	8.5	10.3	10.8	12.8	10.1	9.8	...	9.9	11.2	9.8	9.1	10.5	7.3	13.3	5.6	5.3	-0.607
Asheville	15.4	13.5	13.8	12.6	12.3	13.1	12.4	12.2	9.0	8.4	...	9.2	8.6	8.1	7.9	7.0	6.5	7.0	6.0	6.1	-0.604

5 rows × 21 columns

In [ ] // *Choosing a specific location a checking if that place was in compliance with*
 # EPA primary standards as of the most recent measurement.
```
data_model.loc[data_model['City'] == 'Anchorage']

# 2.4 is less than 9 ppm / 1.4 is less than 12 ug/m3 / 35 at the max standard level of 35.0
```

In [ ]

	CBSA	Core Based Statistical Area	Pollutant statistic	2000	2001	2002	2003	2004	2005	2006	...	2012	2013	2014	2015	2016	2017	2018	2019	City	State/Territory
33	11260	Anchorage, AK	CO-2nd Max	5.4	5.7	4.7	5.7	6.4	4.8	4.3	...	4.3	3.1	3.0	3.0	3.5	2.7	2.4	Anchorage	AK
34	11260	Anchorage, AK	PM2.5-Weighted Annual Mean	5.8	6.2	5.8	6.6	6.9	6.7	6.9	...	6.2	5.6	7.0	6.5	6.1	5.6	5.0	7.4	Anchorage	AK
35	11260	Anchorage, AK	PM2.5-98th Percentile	20.0	25.0	25.0	23.0	30.0	22.0	33.0	...	31.0	22.0	29.0	28.0	23.0	28.0	18.0	35.0	Anchorage	AK

3 rows × 25 columns