```
import numpy as np
import pandas as pd
```

Differences in Differences

Luyi Huang and Joao Mansur

```
df = pd.read_csv(
    "https://github.com/nickeubank/MIDS_Data/blob/master/UDS_arrest_data.csv?raw=true")
```

```
df.head()
```

	YEAR	COUNTY	VIOLENT	F_DRUGOFF	total_population
0	1980	Alameda County	4504	3569	1105379.0
1	1981	Alameda County	4699	3926	1122759.3
2	1982	Alameda County	4389	4436	1140139.6
3	1983	Alameda County	4500	5086	1157519.9
4	1984	Alameda County	3714	5878	1174900.2

The unit of observation is the statistics for a year in a county including violent crime arrests, felony drug arrests, and total population. All CA counties are being tracked from 1980 to 2018.

```
print(min(df["YEAR"]))
```

```
1980
```

```
print(max(df["YEAR"]))
```

```
2018
```

Exercise 2

```
a = df[(df.YEAR==2007)|(df.YEAR==2008)|(df.YEAR==2009)]
```

```
a["average_rate"]=df.F_DRUGOFF/df.total_population
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: SettingWithCopyWarning:
```

```
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_
  """Entry point for launching an IPython kernel.
```

```
a.head()
```

	YEAR	COUNTY	VIOLENT	F_DRUGOFF	total_population	average_rate
27	2007	Alameda County	4443	6071	1490312.0	0.004074
28	2008	Alameda County	4336	5893	1496965.0	0.003937
29	2009	Alameda County	4318	5749	1503618.0	0.003823
66	2007	Alpine County	8	1	1184.9	0.000844
67	2008	Alpine County	4	4	1181.6	0.003385

```
median=pd.DataFrame(a.groupby("YEAR")['average_rate'].median())
median.columns=['Median_Rate']
median.reset_index()
```

	YEAR	Median_Rate
0	2007	0.003385
1	2008	0.003041
2	2009	0.002748

```
a=a.merge(median, left_on='YEAR', right_on='YEAR')
```

```
a.head()
```

	YEAR	COUNTY	VIOLENT	F_DRUGOFF	total_population	average_rate	Median_Rat
0	2007	Alameda County	4443	6071	1490312.0	0.004074	0.00338
1	2007	Alpine County	8	1	1184.9	0.000844	0.00338
2	2007	Amador County	93	111	37193.7	0.002984	0.00338
3	2007	Butte County	743	642	214951.3	0.002987	0.00338
4	2007	Calaveras County	193	157	44070.8	0.003562	0.00338

```
county=a.COUNTY.unique().tolist()
set=pd.DataFrame(county)
set.columns=['COUNTY']
```

```
set['treated']=0
for i in county:
  if any(a.loc[a['COUNTY'] == i]['average_rate']>a.loc[a['COUNTY'] ==i]['Median_Rate']) is Fa
    set['treated'][set.COUNTY==i]=0
  else:
    set['treated'][set.COUNTY==i]=1


a=a.merge(set, left_on='COUNTY', right_on='COUNTY')


a.loc[a['COUNTY'] == 'Butte County']
```

	YEAR	COUNTY	VIOLENT	F_DRUGOFF	total_population	average_rate	Median_Rate
9	2007	Butte County	743	642	214951.3	0.002987	0.003385
10	2008	Butte County	656	581	216634.2	0.002682	0.003041
11	2009	Butte County	641	542	218317.1	0.002483	0.002748

Exercise 3

```
a['vio_rate']=a.VIOLENT/a.total_population*100000
a.head()
```

	YEAR	COUNTY	VIOLENT	F_DRUGOFF	total_population	average_rate	Median_Rate	trea
0	2007	Alameda County	4443	6071	1490312.0	0.004074	0.003385	
1	2008	Alameda County	4336	5893	1496965.0	0.003937	0.003041	
2	2009	Alameda County	4318	5749	1503618.0	0.003823	0.002748	
		Alpine						

Exercise 4

```
post = df[(df.YEAR==2016)|(df.YEAR==2017)|(df.YEAR==2018)]
pre=a
post['vio_rate']=post.VIOLENT*100000/post.total_population
post['average_rate']=post.F_DRUGOFF/post.total_population
post=post.merge(set, left_on='COUNTY', right_on='COUNTY')
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
```

```
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_
  This is separate from the ipykernel package so we can avoid doing imports until
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_
  after removing the cwd from sys.path.
```

```
post.head()
```

	YEAR	COUNTY	VIOLENT	F_DRUGOFF	total_population	vio_rate	average_rate
0	2016	Alameda County	3513	1762	1510271.0	232.607261	0.001167
1	2017	Alameda County	3965	1279	1510271.0	262.535664	0.000847
2	2018	Alameda County	4132	1062	1510271.0	273.593282	0.000703
3	2016	Alpine County	5	0	1175.0	425.531915	0.000000
4	2017	Alpine County	3	2	1175.0	255.319149	0.001702

```
post.groupby('treated')['vio_rate'].mean()
```

```
treated
0    281.606474
1    391.966602
Name: vio_rate, dtype: float64
```

```
pre.groupby('treated')['vio_rate'].mean()
```

```
treated
0    299.336802
1    418.332955
Name: vio_rate, dtype: float64
```

```
#Difference for treated
(391.966602-418.332955)
```

```
-26.366353000000004
```

```
#Diff in Diff
391.96-418.33-(281.61-299.34)
```

```
-8.640000000000043
```

The change in violent arrest rates for our treated group was a decrease of 26.37 crimes per 100k population. Our DinD estimator was only -8.63. [Please ignore the weird decimals after the ones provided]

If we only do the pre-post comparison, we will omit the fact that the violent rate in the untreated group actually increases after 2010, showing a simple comparison would have overestimated the effect.

Exercise 5

```
all = df[(df.YEAR==2016)|(df.YEAR==2017)|(df.YEAR==2018)|(df.YEAR==2007)|(df.YEAR==2008)|(df.
all=all.merge(set, left_on='COUNTY', right_on='COUNTY')
all['vio_rate']=all.VIOLENT*100000/all.total_population
all['average_rate']=all.F_DRUGOFF/all.total_population
```

```
all.head()
```

	YEAR	COUNTY	VIOLENT	F_DRUGOFF	total_population	treated	vio_rate	aver
0	2007	Alameda County	4443	6071	1490312.0	1	298.125493	
1	2008	Alameda County	4336	5893	1496965.0	1	289.652731	
2	2009	Alameda County	4318	5749	1503618.0	1	287.174003	
3	2016	Alameda County	3513	1762	1510271.0	1	232.607261	
4	2017	Alameda County	3965	1279	1510271.0	1	262.535664	

```
all['indict_year']=all["YEAR"].apply(lambda x : 0 if x<2010 else 1)
all['interaction']=all['indict_year']*all['treated']
all.head()
```

	YEAR	COUNTY	VIOLENT	F_DRUGOFF	total_population	treated	vio_rate	average_rat
0	2007	Alameda County	4443	6071	1490312.0	1	298.125493	0.00407
1	2008	Alameda County	4336	5893	1496965.0	1	289.652731	0.00393
2	2009	Alameda County	4318	5749	1503618.0	1	287.174003	0.00382
		Alameda						

```
import statsmodels.api as sm
import statsmodels.formula.api as smf
```

```
model = smf.ols('vio_rate ~ indict_year+treated+interaction', all,missing="drop").fit()
fe_groups = all.copy()
for i in ['indict_year', 'treated', 'vio_rate','interaction']:
    fe_groups = fe_groups[pd.notnull(fe_groups[i])]
model.get_robustcov_results(cov_type="cluster", groups=fe_groups["COUNTY"],missing="drop").su
```

OLS Regression Results

Dep. Variable:	vio_rate	**R-squared:**	0.255
Model:	OLS	**Adj. R-squared:**	0.249
Method:	Least Squares	**F-statistic:**	15.50
Date:	Thu, 04 Mar 2021	**Prob (F-statistic):**	1.73e-07
Time:	03:42:45	**Log-Likelihood:**	-2086.1
No. Observations:	348	**AIC:**	4180.
Df Residuals:	344	**BIC:**	4196.
Df Model:	3		
Covariance Type:	cluster		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	299.3368	15.635	19.145	0.000	268.028	330.646
indict_year	-17.7303	8.903	-1.992	0.051	-35.557	0.097
treated	118.9962	22.044	5.398	0.000	74.853	163.139
interaction	-8.6360	16.749	-0.516	0.608	-42.175	24.903

Omnibus:	14.883	**Durbin-Watson:**	0.779
Prob(Omnibus):	0.001	**Jarque-Bera (JB):**	15.638
Skew:	0.510	**Prob(JB):**	0.000402
Kurtosis:	3.192	**Cond. No.**	7.87

Notes:
[1] Standard Errors are robust to cluster correlation (cluster)

The result we get for the interaction term is equal to the Difference in Difference value we calculated manually in the previous exercise.

This means that interaction terms with two indicator variables function like a Difference in Difference comparison. It expresses the change in condition when both indicators are active.

Exercise 6

```
import matplotlib.pyplot as plt
plt.style.use('seaborn-whitegrid')
```

```
df = pd.read_csv(
    "https://github.com/nickeubank/MIDS_Data/blob/master/UDS_arrest_data.csv?raw=true")
check=df[((df.YEAR<=2009)&(df.YEAR >=2000))|((df.YEAR<=2018)&(df.YEAR >=2016))]
```

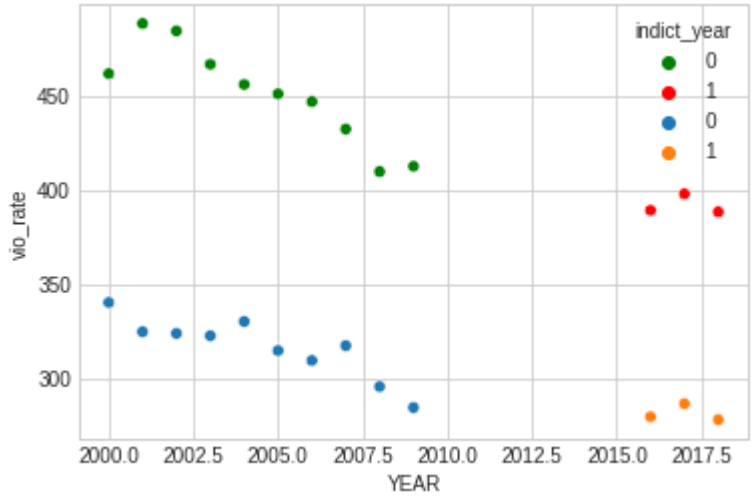
```
check=check.merge(set, left_on='COUNTY', right_on='COUNTY')
check['indict_year']=check["YEAR"].apply(lambda x : 0 if x<=2009 else 1)
check["vio_rate"]=check.VIOLENT*100000/check.total_population


treat=check[check.treated==1]
control=check[check.treated==0]
tre=pd.DataFrame(treat.groupby("YEAR")['vio_rate'].mean())
tre.columns=['vio_rate']
tre=tre.reset_index()
tre['indict_year']=tre["YEAR"].apply(lambda x : 0 if x<=2009 else 1)
ctr=pd.DataFrame(control.groupby("YEAR")['vio_rate'].mean())
ctr.columns=['vio_rate']
ctr=ctr.reset_index()
ctr['indict_year']=ctr["YEAR"].apply(lambda x : 0 if x<=2009 else 1)


import seaborn as sns


sns.scatterplot(x="YEAR", y="vio_rate", data=tre, hue="indict_year", palette=['green','red'])
sns.scatterplot(x="YEAR", y="vio_rate", data=ctr, hue="indict_year")
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fc69ef66dd0>
```



We get more confident about the same parallel assumption. We can see that the tread between control group and treated group is almost the same.

Exercise 7

```
all=all.set_index(['COUNTY','YEAR'])


all
```

		VIOLENT	F_DRUGOFF	total_population	treated	vio_rate	average_rate
COUNTY	YEAR						
Alameda County	**2007**	4443	6071	1490312.0	1	298.125493	0.004074
	2008	4336	5893	1496965.0	1	289.652731	0.003937
	2009	4318	5749	1503618.0	1	287.174003	0.003823
	2016	3513	1762	1510271.0	1	232.607261	0.001167
	2017	3965	1279	1510271.0	1	262.535664	0.000847
...	**...**	...	...	...	...	...	...
Yuba County	**2008**	375	214	69767.8	1	537.497241	0.003067
	2009	354	211	70961.4	1	498.862762	0.002973
	2016	491	154	72155.0	1	680.479523	0.002134
	2017	464	121	72155.0	1	643.060079	0.001677
	2018	391	164	72155.0	1	541.888989	0.002273

348 rows × 8 columns

```
from linearmodels import PanelOLS
mod = PanelOLS.from_formula('vio_rate ~EntityEffects+TimeEffects+indict_year', data=all
)
mod.fit(cov_type='clustered', cluster_entity=True)
```

PanelOLS Estimation Summary

Dep. Variable:	vio_rate	**R-squared:**	0.0000
Estimator:	PanelOLS	**R-squared (Between):**	-0.0769
No. Observations:	348	**R-squared (Within):**	0.0448
Date:	Thu, Mar 04 2021	**R-squared (Overall):**	-0.0745
Time:	03:48:43	**Log-likelihood**	-1858.9
Cov. Estimator:	Clustered		

From the result, since the same trend assumption is satisfied and coeffcients in each regression(DID calculation, fixed effects regression) are negative, we can say that the marijuana legalization reduced violent crime.

Max Obs:	6.0000	**F-statistic (robust):**	1.465e-28
Time periods:	6	**Distribution:**	F(1,284)
Avg Obs:	58.000		
Min Obs:	58.000		
Max Obs:	58.000		

Parameter Estimates

	Parameter	Std. Err.	T-stat	P-value	Lower CI	Upper CI
indict_year	-29.304	2.421e+15	-1.21e-14	1.0000	-4.766e+15	4.766e+15

F-test for Poolability: 17.884
P-value: 0.0000
Distribution: F(62,284)

Included effects: Entity, Time
id: 0x7f6406f23050