IDS 690: Unifying Data Science Traffic Fatalities Problem Set

Derek Wales and Prajwal Vijendra

In [1]:
```
import pandas as pd
import numpy as np
import statsmodels.api as sm
import statsmodels.formula.api as smf
from statsmodels.formula.api import ols
import patsy
from plotnine import *
from scipy.stats import ttest_ind
import matplotlib.pyplot as plt
from matplotlib.colors import ListedColormap
from plotnine import *
import seaborn as sns; sns.set(color_codes=True)
```

Exercise One:

In [2]:
```
# Load in data
traffic_df = pd.read_csv('us_driving_fatalities.csv')

traffic_df.sample(3)
```

Out[2]:

	Unnamed: 0	state	year	spirits	unemp	income	emppop	beertax	baptist	morm
137	138	mi	1986	1.75	8.8	15278.637695	58.407467	0.471886	0.67755	0.2000
95	96	ks	1986	1.14	5.4	14977.295898	64.407715	0.418576	3.39910	0.4845
250	251	pa	1987	1.23	5.7	15200.000000	57.356422	0.240000	0.10000	0.3429

3 rows × 35 columns

In [3]:
```
# Counting the number of states
len(traffic_df['state'].value_counts())
```

Out[3]: 48

In [4]:
```
# Determining the year range
traffic_df.describe()
```

Out[4]:

	Unnamed: 0	year	spirits	unemp	income	emppop	beertax
count	336.000000	336.000000	336.000000	336.000000	336.000000	336.000000	336.000000
mean	168.500000	1985.000000	1.753690	7.346726	13880.184533	60.805676	0.513256
std	97.139076	2.002983	0.683575	2.533405	2253.046291	4.721656	0.477844
min	1.000000	1982.000000	0.790000	2.400000	9513.761719	42.993198	0.043311
25%	84.750000	1983.000000	1.300000	5.475000	12085.849854	57.691426	0.208849
50%	168.500000	1985.000000	1.670000	7.000000	13763.128906	61.364660	0.352589
75%	252.250000	1987.000000	2.012500	8.900000	15175.124268	64.412504	0.651573
max	336.000000	1988.000000	4.900000	18.000000	22193.455078	71.268654	2.720764

8 rows × 31 columns

This data contains information on traffic deaths from 48 states from 1982 to 1988. This dataset is a by year by state representation of the demographics and traffic fatalities.

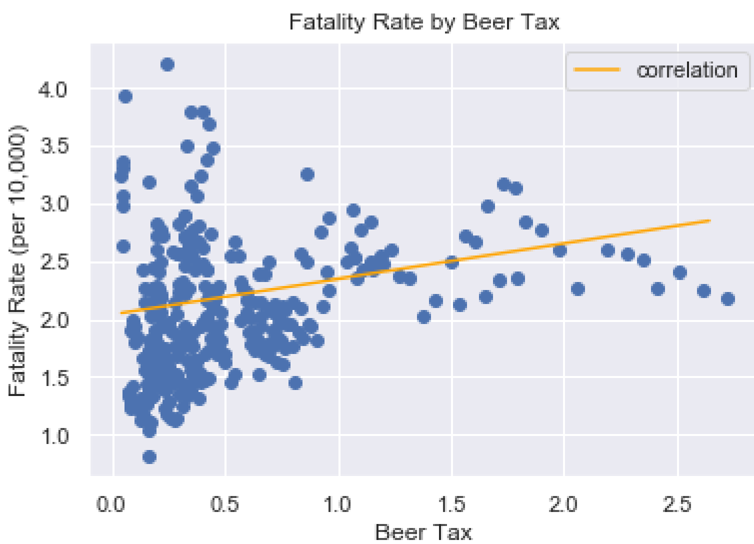
Exercise Two: Calculating Fatality Rate

In [5]:
```
traffic_df["fat_rate"] =  (traffic_df['fatal']/traffic_df['pop'])*10000
```

Exercise Three: Draw a Scatterplot

In [6]:
```
beertax_range = np.arange(traffic_df.beertax.min(), traffic_df.beertax.max(),
.1) # range of beertax values to plot
correl = traffic_df[['fat_rate', 'beertax']].corr().iloc[1,0] # correlation be
tween beer tax and fatality rate
correl_line = beertax_range * correl + traffic_df.fat_rate.mean() # predicted
 fatality rate based on correlation

# Plot
plt.scatter(x=traffic_df.beertax, y=traffic_df.fat_rate)
plt.plot(beertax_range, correl_line, color = 'orange', label = 'correlation')
plt.xlabel('Beer Tax')
plt.ylabel('Fatality Rate (per 10,000)')
plt.title('Fatality Rate by Beer Tax')
plt.legend()
plt.show()
```



Exercise Four: Fitting a Simple OLS

In [7]:
```
smf.ols('fat_rate ~ beertax', data = traffic_df).fit().summary()
```

Out[7]:

OLS Regression Results

Dep. Variable:	fat_rate	**R-squared:**	0.093
Model:	OLS	**Adj. R-squared:**	0.091
Method:	Least Squares	**F-statistic:**	34.39
Date:	Mon, 24 Feb 2020	**Prob (F-statistic):**	1.08e-08
Time:	17:16:06	**Log-Likelihood:**	-271.04
No. Observations:	336	**AIC:**	546.1
Df Residuals:	334	**BIC:**	553.7
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	1.8533	0.044	42.539	0.000	1.768	1.939
beertax	0.3646	0.062	5.865	0.000	0.242	0.487

Omnibus:	66.653	**Durbin-Watson:**	0.465
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	112.734
Skew:	1.134	**Prob(JB):**	3.31e-25
Kurtosis:	4.707	**Cond. No.**	2.76

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

A simple OLS seems to suggest that as the beer tax increases so does the driving fatalities.

Exercise Five: OLS w/Fixed Effects for States

In [8]:
```
smf.ols('fat_rate ~ beertax + C(state)', data = traffic_df).fit().summary()
```

Out[8]:

OLS Regression Results

Dep. Variable:	fat_rate	**R-squared:**	0.905
Model:	OLS	**Adj. R-squared:**	0.889
Method:	Least Squares	**F-statistic:**	56.97
Date:	Mon, 24 Feb 2020	**Prob (F-statistic):**	1.96e-120
Time:	17:16:07	**Log-Likelihood:**	107.97
No. Observations:	336	**AIC:**	-117.9
Df Residuals:	287	**BIC:**	69.09
Df Model:	48		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	3.4776	0.313	11.098	0.000	2.861	4.094
C(state)[T.ar]	-0.6550	0.219	-2.990	0.003	-1.086	-0.224
C(state)[T.az]	-0.5677	0.267	-2.129	0.034	-1.093	-0.043
C(state)[T.ca]	-1.5095	0.304	-4.960	0.000	-2.109	-0.910
C(state)[T.co]	-1.4843	0.287	-5.165	0.000	-2.050	-0.919
C(state)[T.ct]	-1.8623	0.281	-6.638	0.000	-2.414	-1.310
C(state)[T.de]	-1.3076	0.294	-4.448	0.000	-1.886	-0.729
C(state)[T.fl]	-0.2681	0.139	-1.924	0.055	-0.542	0.006
C(state)[T.ga]	0.5246	0.184	2.852	0.005	0.163	0.887
C(state)[T.ia]	-1.5439	0.253	-6.092	0.000	-2.043	-1.045
C(state)[T.id]	-0.6690	0.258	-2.593	0.010	-1.177	-0.161
C(state)[T.il]	-1.9616	0.291	-6.730	0.000	-2.535	-1.388
C(state)[T.in]	-1.4615	0.273	-5.363	0.000	-1.998	-0.925
C(state)[T.ks]	-1.2232	0.245	-4.984	0.000	-1.706	-0.740
C(state)[T.ky]	-1.2175	0.287	-4.240	0.000	-1.783	-0.652
C(state)[T.la]	-0.8471	0.189	-4.490	0.000	-1.218	-0.476
C(state)[T.ma]	-2.1097	0.276	-7.641	0.000	-2.653	-1.566
C(state)[T.md]	-1.7064	0.283	-6.025	0.000	-2.264	-1.149
C(state)[T.me]	-1.1079	0.191	-5.797	0.000	-1.484	-0.732
C(state)[T.mi]	-1.4845	0.236	-6.290	0.000	-1.949	-1.020
C(state)[T.mn]	-1.8972	0.265	-7.157	0.000	-2.419	-1.375
C(state)[T.mo]	-1.2963	0.267	-4.861	0.000	-1.821	-0.771
C(state)[T.ms]	-0.0291	0.148	-0.196	0.845	-0.321	0.263
C(state)[T.mt]	-0.3604	0.264	-1.365	0.173	-0.880	0.159
C(state)[T.nc]	-0.2905	0.120	-2.424	0.016	-0.526	-0.055

C(state)[T.nd]	-1.6234	0.254	-6.396	0.000	-2.123	-1.124
C(state)[T.ne]	-1.5222	0.249	-6.106	0.000	-2.013	-1.032
C(state)[T.nh]	-1.2545	0.210	-5.983	0.000	-1.667	-0.842
C(state)[T.nj]	-2.1057	0.307	-6.855	0.000	-2.710	-1.501
C(state)[T.nm]	0.4264	0.254	1.677	0.095	-0.074	0.927
C(state)[T.nv]	-0.6008	0.286	-2.101	0.037	-1.164	-0.038
C(state)[T.ny]	-2.1867	0.299	-7.316	0.000	-2.775	-1.598
C(state)[T.oh]	-1.6744	0.254	-6.597	0.000	-2.174	-1.175
C(state)[T.ok]	-0.5451	0.169	-3.223	0.001	-0.878	-0.212
C(state)[T.or]	-1.1680	0.286	-4.088	0.000	-1.730	-0.606
C(state)[T.pa]	-1.7675	0.276	-6.402	0.000	-2.311	-1.224
C(state)[T.ri]	-2.2651	0.294	-7.711	0.000	-2.843	-1.687
C(state)[T.sc]	0.5572	0.110	5.065	0.000	0.341	0.774
C(state)[T.sd]	-1.0037	0.210	-4.788	0.000	-1.416	-0.591
C(state)[T.tn]	-0.8757	0.268	-3.267	0.001	-1.403	-0.348
C(state)[T.tx]	-0.9175	0.246	-3.736	0.000	-1.401	-0.434
C(state)[T.ut]	-1.1640	0.196	-5.926	0.000	-1.551	-0.777
C(state)[T.va]	-1.2902	0.204	-6.320	0.000	-1.692	-0.888
C(state)[T.vt]	-0.9660	0.211	-4.576	0.000	-1.382	-0.550
C(state)[T.wa]	-1.6595	0.283	-5.854	0.000	-2.217	-1.102
C(state)[T.wi]	-1.7593	0.294	-5.985	0.000	-2.338	-1.181
C(state)[T.wv]	-0.8968	0.247	-3.636	0.000	-1.382	-0.411
C(state)[T.wy]	-0.2285	0.313	-0.730	0.466	-0.844	0.387
beertax	-0.6559	0.188	-3.491	0.001	-1.026	-0.286

Omnibus:	53.045	**Durbin-Watson:**	1.517
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	219.863
Skew:	0.585	**Prob(JB):**	1.81e-48
Kurtosis:	6.786	**Cond. No.**	187.

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

When controls for state are added, the coefficient of the beer tax goes from positive (suggesting that it increases the fatality rate) to negative. Showing that a beer tax drops fatality rate.

Exercise Six A: Explain the Results from the Models with and without Fixed Effects

Each state has drastically different driving patterns so when you control for those states it reduces the effect of the beer tax. This implies that states with high beer taxes can reduce fatalities.

Exercise Six B: Demeaning

In [9]:
```
# Mean Centering by state.
traffic_df['fat_rate_state_c'] = traffic_df['fat_rate'] - traffic_df.groupby([
'state']).fat_rate.transform(np.mean)
traffic_df['beertax_state_c'] = traffic_df['beertax'] - traffic_df.groupby(['s
tate']).beertax.transform(np.mean)
```

In [10]:
```
# Linear Regression
smf.ols('fat_rate_state_c ~ beertax_state_c', data = traffic_df).fit().summary
()
```

Out[10]:

OLS Regression Results

Dep. Variable:	fat_rate_state_c	**R-squared:**	0.041
Model:	OLS	**Adj. R-squared:**	0.038
Method:	Least Squares	**F-statistic:**	14.19
Date:	Mon, 24 Feb 2020	**Prob (F-statistic):**	0.000196
Time:	17:16:10	**Log-Likelihood:**	107.97
No. Observations:	336	**AIC:**	-211.9
Df Residuals:	334	**BIC:**	-204.3
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	-1.648e-17	0.010	-1.72e-15	1.000	-0.019	0.019
beertax_state_c	-0.6559	0.174	-3.767	0.000	-0.998	-0.313

Omnibus:	53.045	**Durbin-Watson:**	1.517
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	219.863
Skew:	0.585	**Prob(JB):**	1.81e-48
Kurtosis:	6.786	**Cond. No.**	18.1

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Exercise Seven: Fitting the Model with fixed effects

In [11]:
```
# Create multi-index
df_multiindex = traffic_df.set_index(['state', traffic_df.index])
#df_multiindex.head(15)
```

In [12]:
```
from statsmodels.formula.api import ols
```

In [13]:
```
from linearmodels import PanelOLS
```

In [14]:
```
# Run fixed effects model
# Entity fixed effects
(
    PanelOLS.from_formula('fat_rate ~ beertax + EntityEffects', data = df_mult
iindex)
    .fit(cov_type='clustered', cluster_entity=True)
)
```

Out[14]:

PanelOLS Estimation Summary

Dep. Variable:	fat_rate	**R-squared:**	0.0407
Estimator:	PanelOLS	**R-squared (Between):**	-0.3805
No. Observations:	336	**R-squared (Within):**	0.0407
Date:	Mon, Feb 24 2020	**R-squared (Overall):**	-0.3775
Time:	17:16:15	**Log-likelihood**	107.97
Cov. Estimator:	Clustered		
		F-statistic:	12.190
Entities:	48	**P-value**	0.0006
Avg Obs:	7.0000	**Distribution:**	F(1,287)
Min Obs:	7.0000		
Max Obs:	7.0000	**F-statistic (robust):**	5.1576
		P-value	0.0239
Time periods:	336	**Distribution:**	F(1,287)
Avg Obs:	1.0000		
Min Obs:	1.0000		
Max Obs:	1.0000		

Parameter Estimates

	Parameter	Std. Err.	T-stat	P-value	Lower CI	Upper CI
beertax	-0.6559	0.2888	-2.2710	0.0239	-1.2243	-0.0874

F-test for Poolability: 52.179
P-value: 0.0000
Distribution: F(47,287)

Included effects: Entity
id: 0x2e8ae4e7388

The beertax estimations between exercise 6 and 7 are the same of a -.656 decrease in fatality rate.

Exercise Eight:

In [ ]:

```

```