Import all modules

In [63]:

```
import os
import pydotplus
import numpy as np
import pandas as pd
import seaborn as sns
import statsmodels.api as sm
import matplotlib.pyplot as plt
from sklearn import tree
from IPython.display import Image
from sklearn.datasets import load_iris
from sklearn.naive_bayes import GaussianNB
from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import OneHotEncoder
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier, plot_tree
from sklearn.tree.export import export_text
from sklearn.tree import export_graphviz
from sklearn.metrics import accuracy_score
from sklearn.externals.six import StringIO
```

Read training

Read data from training set.

In [27]:

```
training_set = pd.read_csv("adult.data")
training_set = training_set[~(training_set.eq(' ?')).any(1)]
training_set
```

```
C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\ops\array_ops.py:253: Future
Warning: elementwise comparison failed; returning scalar instead, but in the future will per
form elementwise comparison
  res_values = method(rvalues)
```

Out[27]:

	age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race
0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White
1	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White
2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White
3	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black
4	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black
...	...	...	...	...	...	...	...	...	..
32556	27	Private	257302	Assoc-acdm	12	Married-civ-spouse	Tech-support	Wife	White
32557	40	Private	154374	HS-grad	9	Married-civ-spouse	Machine-op-inspct	Husband	White
32558	58	Private	151910	HS-grad	9	Widowed	Adm-clerical	Unmarried	White
32559	22	Private	201490	HS-grad	9	Never-married	Adm-clerical	Own-child	White
32560	52	Self-emp-inc	287927	HS-grad	9	Married-civ-spouse	Exec-managerial	Wife	White

30162 rows × 15 columns

Read test

Read data from test set, and get rid of unknown('?') values.

In [28]:

```
test_set = pd.read_csv("adult.test")
test_set = test_set[~(test_set.eq(' ?')).any(1)]
test_set
```

```
C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\ops\array_ops.py:253: Future
Warning: elementwise comparison failed; returning scalar instead, but in the future will per
form elementwise comparison
  res_values = method(rvalues)
```

Out[28]:

	age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	ra
0	25	Private	226802	11th	7	Never-married	Machine-op-inspct	Own-child	Bla
1	38	Private	89814	HS-grad	9	Married-civ-spouse	Farming-fishing	Husband	Wh
2	28	Local-gov	336951	Assoc-acdm	12	Married-civ-spouse	Protective-serv	Husband	Wh
3	44	Private	160323	Some-college	10	Married-civ-spouse	Machine-op-inspct	Husband	Bla
5	34	Private	198693	10th	6	Never-married	Other-service	Not-in-family	Wh
...	...	...	...	...	...	...	...	...	
16275	33	Private	245211	Bachelors	13	Never-married	Prof-specialty	Own-child	Wh
16276	39	Private	215419	Bachelors	13	Divorced	Prof-specialty	Not-in-family	Wh
16278	38	Private	374983	Bachelors	13	Married-civ-spouse	Prof-specialty	Husband	Wh
16279	44	Private	83891	Bachelors	13	Divorced	Adm-clerical	Own-child	Asia Pa Islanc
16280	35	Self-emp-inc	182148	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	Wh

15060 rows × 15 columns

Preprocess the training set data and test set data

Change string to number using label encoder.

In [29]:

```
lba = LabelEncoder()
workclass = training_set['workclass']
lba.fit(workclass)
training_set['workclass'] = lba.transform(training_set['workclass'])

lbatest = LabelEncoder()
workclass = test_set['workclass']
lbatest.fit(workclass)
test_set['workclass'] = lbatest.transform(test_set['workclass'])

lbb = LabelEncoder()
education = training_set['education']
lbb.fit(education)
training_set['education'] = lbb.transform(training_set['education'])

lbbtest = LabelEncoder()
workclass = test_set['education']
lbbtest.fit(education)
test_set['education'] = lbbtest.transform(test_set['education'])

lbc = LabelEncoder()
marital_status = training_set['marital-status']
lbc.fit(marital_status)
training_set['marital-status'] = lbc.transform(training_set['marital-status'])

lbctest = LabelEncoder()
marital_status = test_set['marital-status']
lbctest.fit(marital_status)
test_set['marital-status'] = lbctest.transform(test_set['marital-status'])

lbd = LabelEncoder()
occupation = training_set['occupation']
lbd.fit(occupation)
training_set['occupation'] = lbd.transform(training_set['occupation'])

lbdtest = LabelEncoder()
occupation = test_set['occupation']
lbdtest.fit(occupation)
test_set['occupation'] = lbdtest.transform(test_set['occupation'])

lbe = LabelEncoder()
relationship = training_set['relationship']
lbe.fit(relationship)
training_set['relationship'] = lbe.transform(training_set['relationship'])

lbetest = LabelEncoder()
relationship = test_set['relationship']
lbetest.fit(relationship)
test_set['relationship'] = lbetest.transform(test_set['relationship'])

lbf = LabelEncoder()
race = training_set['race']
lbf.fit(race)
training_set['race'] = lbf.transform(training_set['race'])

lbftest = LabelEncoder()
race = test_set['race']
lbftest.fit(race)
test_set['race'] = lbftest.transform(test_set['race'])
```

```
lbg = LabelEncoder()
sex = training_set['sex']
lbg.fit(sex)
training_set['sex'] = lbg.transform(training_set['sex'])

lbgtest = LabelEncoder()
sex = test_set['sex']
lbgtest.fit(sex)
test_set['sex'] = lbgtest.transform(test_set['sex'])

lbh = LabelEncoder()
nativecountry = training_set['native-country']
lbh.fit(nativecountry)
training_set['native-country'] = lbh.transform(training_set['native-country'])

lbhtest = LabelEncoder()
nativecountry = test_set['native-country']
lbhtest.fit(nativecountry)
test_set['native-country'] = lbhtest.transform(test_set['native-country'])

lbi = LabelEncoder()
income = training_set['income']
lbi.fit(income)
training_set['income'] = lbi.transform(training_set['income'])

lbitest = LabelEncoder()
income = test_set['income']
lbitest.fit(income)
test_set['income'] = lbitest.transform(test_set['income'])
```

Load training data and test data

In [36]:

```
X_train = training_set.iloc[:,:-1]
X_train
```

Out[36]:

	age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race
0	39	5	77516	9	13	4	0	1	4
1	50	4	83311	9	13	2	3	0	4
2	38	2	215646	11	9	0	5	1	4
3	53	2	234721	1	7	2	5	0	2
4	28	2	338409	9	13	2	9	5	2
...	...	...	...	...	...	...	...	...	...
32556	27	2	257302	7	12	2	12	5	4
32557	40	2	154374	11	9	2	6	0	4
32558	58	2	151910	11	9	6	0	4	4
32559	22	2	201490	11	9	4	0	3	4
32560	52	3	287927	11	9	2	3	5	4

30162 rows × 14 columns

Y_train

In [33]:

```
Y_train = training_set.iloc[:,-1:]
Y_train
```

Out[33]:

	income
0	0
1	0
2	0
3	0
4	0
...	...
32556	0
32557	1
32558	0
32559	0
32560	1

30162 rows × 1 columns

X_test

In [34]:

```
X_test = test_set.iloc[:,:-1]
X_test
```

Out[34]:

	age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race
0	25	2	226802	1	7	4	6	3	2
1	38	2	89814	11	9	2	4	0	4
2	28	1	336951	7	12	2	10	0	4
3	44	2	160323	15	10	2	6	0	2
5	34	2	198693	0	6	4	7	1	4
...	...	...	...	...	...	...	...	...	...
16275	33	2	245211	9	13	4	9	3	4
16276	39	2	215419	9	13	0	9	1	4
16278	38	2	374983	9	13	2	9	0	4
16279	44	2	83891	9	13	0	0	3	1
16280	35	3	182148	9	13	2	3	0	4

15060 rows × 14 columns

Y_test

In [35]:

```
Y_test = test_set.iloc[:,-1:]
Y_test
```

Out[35]:

	income
0	0
1	0
2	1
3	1
5	0
...	...
16275	0
16276	0
16278	0
16279	0
16280	1

15060 rows × 1 columns

Naive Bayes

In [43]:

```
gnb = GaussianNB()
Y_pred = gnb.fit(X_train, Y_train.values.ravel()).predict(X_test)
print("Total points : %d    Mislabeled points : %d"
      % (X_test.shape[0], (Y_test.values.ravel() != Y_pred).sum()))
```

Total points : 15060   Mislabeled points : 3184

Gini

In [67]:

```
clf = tree.DecisionTreeClassifier(criterion='gini')
clf.fit(X_train,Y_train)
Y_predict = clf.predict(X_test)
accuracy_score(Y_test,Y_predict)
```

Out[67]:

0.8027888446215139

Entropy

In [68]:

```
clf1 = tree.DecisionTreeClassifier(criterion='entropy')
clf1.fit(X_train,Y_train)
Y_predict1 = clf1.predict(X_test)
accuracy_score(Y_test,Y_predict1)
```

Out[68]:

```
0.8067065073041169
```

The Correlation

In [66]:

```
stats = sm.add_constant(X_train)
model = sm.OLS(Y_train, stats).fit()
prediction = model.predict(stats)
summary_model = model.summary()
DF = pd.DataFrame(X_train.describe())
print(DF,"\n")
print(prediction,"\n")
print(training_set.corr())
summary_model
```

```
                age    workclass       fnlwgt    education  education-num  \
count  30162.000000  30162.000000  3.016200e+04  30162.000000  30162.000000
mean      38.437902      2.199324  1.897938e+05     10.333764     10.121312
std       13.134665      0.953925  1.056530e+05      3.812292      2.549995
min       17.000000      0.000000  1.376900e+04      0.000000      1.000000
25%       28.000000      2.000000  1.176272e+05      9.000000      9.000000
50%       37.000000      2.000000  1.784250e+05     11.000000     10.000000
75%       47.000000      2.000000  2.376285e+05     12.000000     13.000000
max       90.000000      6.000000  1.484705e+06     15.000000     16.000000

       marital-status    occupation  relationship          race           sex  \
count    30162.000000  30162.000000  30162.000000  30162.000000  30162.000000
mean         2.580134      5.959850      1.418341      3.678602      0.675685
std          1.498016      4.029566      1.601338      0.834709      0.468126
min          0.000000      0.000000      0.000000      0.000000      0.000000
25%          2.000000      2.000000      0.000000      4.000000      0.000000
50%          2.000000      6.000000      1.000000      4.000000      1.000000
75%          4.000000      9.000000      3.000000      4.000000      1.000000
max          6.000000     13.000000      5.000000      4.000000      1.000000

       capital-gain  capital-loss  hours-per-week  native-country
count  30162.000000  30162.000000    30162.000000    30162.000000
mean    1092.007858     88.372489       40.931238       36.382567
std     7406.346497    404.298370       11.979984        6.105372
min        0.000000      0.000000        1.000000        0.000000
25%        0.000000      0.000000       40.000000       38.000000
50%        0.000000      0.000000       40.000000       38.000000
75%        0.000000      0.000000       45.000000       38.000000
max    99999.000000   4356.000000       99.000000       40.000000

0        0.350397
1        0.375028
2        0.276567
3        0.233115
4        0.207117
           ...
32556    0.161495
32557    0.253743
32558    0.072231
32559   -0.006412
32560    0.253604
Length: 30162, dtype: float64

                     age  workclass    fnlwgt  education  education-num  \
age             1.000000   0.080540 -0.076511  -0.001111       0.043526
workclass       0.080540   1.000000 -0.032493   0.017855       0.037833
fnlwgt         -0.076511  -0.032493  1.000000  -0.027102      -0.044992
education      -0.001111   0.017855 -0.027102   1.000000       0.345410
education-num   0.043526   0.037833 -0.044992   0.345410       1.000000
```

```
marital-status -0.276373  -0.034241  0.032163  -0.040664      -0.063419
occupation     -0.005682   0.015572  0.000204  -0.038212       0.087717
relationship   -0.246456  -0.067417  0.009298  -0.012717      -0.091935
race            0.023374   0.044731 -0.023895   0.011154       0.032805
sex             0.081993   0.074973  0.025362  -0.027888       0.006157
capital-gain    0.080154   0.035350  0.000422   0.030575       0.124416
capital-loss    0.060165   0.007204 -0.009750   0.015028       0.079646
hours-per-week  0.101599   0.050724 -0.022886   0.059887       0.152522
native-country -0.001905   0.007668 -0.066717   0.078790       0.091555
income          0.241998   0.018044 -0.008957   0.078987       0.335286

                marital-status  occupation  relationship      race       sex  \
age                  -0.276373   -0.005682     -0.246456  0.023374  0.081993
workclass            -0.034241    0.015572     -0.067417  0.044731  0.074973
fnlwgt                0.032163    0.000204      0.009298 -0.023895  0.025362
education            -0.040664   -0.038212     -0.012717  0.011154 -0.027888
education-num        -0.063419    0.087717     -0.091935  0.032805  0.006157
marital-status        1.000000    0.022655      0.177964 -0.068627 -0.119813
occupation            0.022655    1.000000     -0.053727  0.000717  0.062313
relationship          0.177964   -0.053727      1.000000 -0.117143 -0.584876
race                 -0.068627    0.000717     -0.117143  1.000000  0.089186
sex                  -0.119813    0.062313     -0.584876  0.089186  1.000000
capital-gain         -0.042418    0.022162     -0.058259  0.014353  0.048814
capital-loss         -0.035203    0.014607     -0.063567  0.023517  0.047011
hours-per-week       -0.189003    0.018365     -0.257850  0.048532  0.231268
native-country       -0.025902   -0.003483     -0.010809  0.124514  0.000618
income               -0.193518    0.051577     -0.251003  0.071658  0.216699

                capital-gain  capital-loss  hours-per-week  native-country  \
age                 0.080154      0.060165        0.101599       -0.001905
workclass           0.035350      0.007204        0.050724        0.007668
fnlwgt              0.000422     -0.009750       -0.022886       -0.066717
education           0.030575      0.015028        0.059887        0.078790
education-num       0.124416      0.079646        0.152522        0.091555
marital-status     -0.042418     -0.035203       -0.189003       -0.025902
occupation          0.022162      0.014607        0.018365       -0.003483
relationship       -0.058259     -0.063567       -0.257850       -0.010809
race                0.014353      0.023517        0.048532        0.124514
sex                 0.048814      0.047011        0.231268        0.000618
capital-gain        1.000000     -0.032229        0.080432        0.008530
capital-loss       -0.032229      1.000000        0.052417        0.009386
hours-per-week      0.080432      0.052417        1.000000        0.008408
native-country      0.008530      0.009386        0.008408        1.000000
income              0.221196      0.150053        0.229480        0.023268

                  income
age             0.241998
workclass       0.018044
fnlwgt         -0.008957
education       0.078987
education-num   0.335286
marital-status -0.193518
occupation      0.051577
relationship   -0.251003
race            0.071658
sex             0.216699
capital-gain    0.221196
capital-loss    0.150053
hours-per-week  0.229480
native-country  0.023268
income          1.000000
```

Out[66]:

OLS Regression Results

Dep. Variable:	income	**R-squared:**	0.262
Model:	OLS	**Adj. R-squared:**	0.262
Method:	Least Squares	**F-statistic:**	766.4
Date:	Thu, 09 Apr 2020	**Prob (F-statistic):**	0.00
Time:	03:08:49	**Log-Likelihood:**	-12918.
No. Observations:	30162	**AIC:**	2.587e+04
Df Residuals:	30147	**BIC:**	2.599e+04
Df Model:	14		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
const	-0.5826	0.023	-25.267	0.000	-0.628	-0.537
age	0.0052	0.000	29.600	0.000	0.005	0.006
workclass	-0.0148	0.002	-6.568	0.000	-0.019	-0.010
fnlwgt	7.129e-08	2.04e-08	3.497	0.000	3.13e-08	1.11e-07
education	-0.0035	0.001	-5.772	0.000	-0.005	-0.002
education-num	0.0485	0.001	52.600	0.000	0.047	0.050
marital-status	-0.0230	0.002	-15.169	0.000	-0.026	-0.020
occupation	0.0012	0.001	2.198	0.028	0.000	0.002
relationship	-0.0160	0.002	-9.254	0.000	-0.019	-0.013
race	0.0152	0.003	5.815	0.000	0.010	0.020
sex	0.1117	0.006	19.524	0.000	0.100	0.123
capital-gain	9.192e-06	2.93e-07	31.378	0.000	8.62e-06	9.77e-06
capital-loss	0.0001	5.33e-06	21.203	0.000	0.000	0.000
hours-per-week	0.0035	0.000	18.179	0.000	0.003	0.004
native-country	-0.0006	0.000	-1.657	0.098	-0.001	0.000

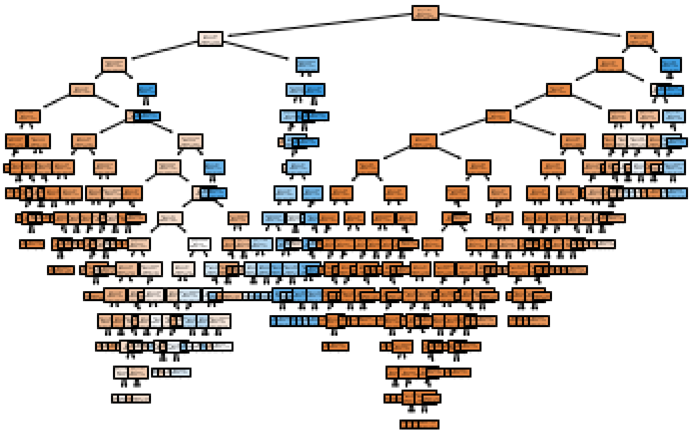
Omnibus:	2688.738	**Durbin-Watson:**	2.003
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	3229.701
Skew:	0.779	**Prob(JB):**	0.00
Kurtosis:	2.624	**Cond. No.**	2.35e+06

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 2.35e+06. This might indicate that there are
strong multicollinearity or other numerical problems.

Tree diagram

In [64]:

```
plt.figure()
plot_tree(clf, filled=True)
plt.show()
```



In [ ]:

```
```