```
In [36]: import numpy as np
         import pandas as pd
         from sklearn.feature_extraction.text import CountVectorizer
         from sklearn.feature_extraction.text import TfidfVectorizer
```

```
In [37]: df = pd.read_csv('https://raw.githubusercontent.com/carlson9/KocPython2020/mas
         ter/homework/immSurvey.csv')
```

In [38]:

```
df
```

Out[38]:

| | Unnamed: 0 | ids | X.1 | textToSend | Means | stanMeans | X | MetaID | treatment | pid_rep |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 6606 | 1 | problems caused by the influx of illegal immig... | 0.750000 | 2.409768 | 1 | 0 | 1 | 1.00000 |
| 1 | 2 | 6607 | 2 | if you mean illegal immigration, i'm afraid of... | 0.875000 | 3.710615 | 2 | 0 | 1 | 1.00000 |
| 2 | 3 | 6608 | 3 | that they should enter the same way my grandpa... | 0.416667 | -1.437706 | 3 | 0 | 0 | 0.33300 |
| 3 | 4 | 6609 | 4 | legally entering the usa meeting the requireme... | 0.458333 | 0.655503 | 4 | 0 | 0 | 0.50000 |
| 4 | 5 | 6610 | 5 | terror bombings killing us robbing america | 0.875000 | 5.337525 | 5 | 0 | 1 | 0.66667 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 335 | 337 | 6939 | 337 | those people that are here illegally should be... | 0.650000 | -0.326699 | 336 | 0 | 0 | 0.50000 |
| 336 | 338 | 6940 | 338 | racism & xenophobia make me worried! also, co... | 0.650000 | 0.789189 | 337 | 0 | 1 | 0.50000 |
| 337 | 339 | 6941 | 339 | that immigrants are the heart and soul and the... | 0.200000 | -2.239257 | 338 | 0 | 0 | 0.50000 |
| 338 | 340 | 6942 | 340 | job security for our citizens | 0.300000 | -1.716588 | 339 | 0 | 1 | 0.50000 |
| 339 | 341 | 6943 | 341 | public safety, effect on the economy, whether ... | 0.842105 | 4.268994 | 340 | 0 | 1 | 0.50000 |

340 rows × 38 columns

In [39]:
```
alphas = df.stanMeansNewSysPooled
sample = df.textToSend

v = CountVectorizer()
x = v.fit_transform(sample)
```

In [40]:
```
x
```

Out[40]:
```
<340x1475 sparse matrix of type '<class 'numpy.int64'>'
        with 6239 stored elements in Compressed Sparse Row format>
```

In [41]:
```
pd.DataFrame(x.toarray(), columns=v.get_feature_names())
```

Out[41]:

| | 11 | 12 | 125 | 18 | 1b | 600 | 95 | able | abolition | aboration | ... | wreckless | wrong | wrongly | x |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 | |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 | |
| 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 | |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 | |
| 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 | |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | |
| 335 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 | |
| 336 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 | |
| 337 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 1 | 0 | |
| 338 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 | |
| 339 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 | |

340 rows × 1475 columns

In [42]:
```
from sklearn.model_selection import train_test_split
xtrain, xtest, ytrain, ytest = train_test_split(x, alphas,
random_state=1)
```

In [43]:
```
from sklearn.gaussian_process import GaussianProcessRegressor
from sklearn.gaussian_process.kernels import ConstantKernel, RBF

rbf = ConstantKernel(1.0) * RBF(length_scale=1.0)
gpr = GaussianProcessRegressor(kernel=rbf, alpha=1e-8)
gpr.fit(xtrain.toarray(), ytrain)
```

Out[43]:
```
GaussianProcessRegressor(alpha=1e-08, copy_X_train=True,
                         kernel=1**2 * RBF(length_scale=1),
                         n_restarts_optimizer=0, normalize_y=False,
                         optimizer='fmin_l_bfgs_b', random_state=None)
```

In [44]:
```
m, c = gpr.predict(xtest.toarray(), return_cov=True)
```

In [45]:
```
print (m,c)
```

```
[-4.70368302e-01 -1.88451167e-01  1.79316802e-01 -1.08671571e+00
  7.72459599e-02  4.54517233e-03 -4.21716023e-02  5.49369072e-09
  2.98879892e-03  5.09086166e-02 -1.50141289e-01 -1.66857615e-02
  5.43721214e-01  1.23403909e-11 -6.22090133e-01 -4.13130805e-01
 -7.64168129e-01  3.52373448e-01 -4.52882609e-01 -1.29262377e+00
 -8.32969768e-02 -7.29553404e-01 -9.95216088e-01  3.68688097e-03
  8.54655530e-14  2.87784204e-02 -1.16016714e+00 -2.84864588e-01
 -8.39988585e-02 -1.07503417e+00 -2.16794271e-04  4.15471760e-06
 -1.09992427e-01  1.08702424e-04 -1.42257897e+00 -1.34403006e-02
 -7.06904402e-02 -1.65489446e-01 -1.22983445e-03 -1.61471917e-03
 -7.37694006e-01 -1.62698645e-01 -3.74296032e-01  3.02417965e-01
 -8.35432977e-01  1.24294052e-01 -9.68070785e-01  5.50835571e-02
  5.53143826e-09 -1.87067406e-02  2.11023464e-02 -1.81546821e-01
 -3.76305514e-02 -7.19630101e-02  9.31731214e-04 -1.26322946e+00
  1.22225891e+00 -5.69459234e-03 -2.12699263e-02 -2.32891069e-01
  2.21443661e-02  7.82059053e-06 -2.06860989e-01  1.88364332e-01
 -1.88297873e-01 -1.03254425e-01 -8.24605644e-01  3.65161751e-04
 -9.47848804e-02 -3.19408726e-01 -3.10947531e-03 -1.86026687e-02
 -1.52503957e+00 -1.39136038e-01 -3.28390340e-03 -3.30650003e-01
 -1.19234770e+00  7.33661635e-02 -3.47838643e-01 -5.62486050e-02
  2.90508862e-01 -4.67463932e-03  7.33291588e-04 -1.19234770e+00
  7.86244634e-02] [[ 4.79211237e-01 -5.96123032e-03 -2.26081337e-03 ... -1.55
211484e-05
   2.80687473e-10 -1.27233189e-04]
 [-5.96123032e-03  7.12749388e-01 -4.93387333e-03 ... -4.63993380e-06
   2.06233752e-10  1.84411957e-04]
 [-2.26081337e-03 -4.93387333e-03  3.80176056e-01 ...  3.69791304e-05
   7.16987580e-11  6.38047600e-04]
 ...
 [-1.55211484e-05 -4.63993380e-06  3.69791304e-05 ...  9.43235175e-01
  -1.60513584e-12  5.52921394e-04]
 [ 2.80687251e-10  2.06233919e-10  7.16988691e-11 ... -1.60513578e-12
   9.99999950e-09 -6.18173568e-11]
 [-1.27233189e-04  1.84411957e-04  6.38047600e-04 ...  5.52921394e-04
  -6.18173464e-11  9.38018398e-01]]
```

In [54]:
```
np.corrcoef(ytest,m)
```

Out[54]:
```
array([[1.        , 0.63286061],
       [0.63286061, 1.        ]])
```

In [55]:
```
#CORRELATION COEFFFICIENT IS 0.6328
```

In [56]:
```
bigram_vectorizer =  CountVectorizer(ngram_range=(2, 2), token_pattern=r'\b\w+
\b', min_df=1)
```

In [57]:
```
anl = bigram_vectorizer.build_analyzer()
```

In [58]:
```
new_x = bigram_vectorizer.fit_transform(sample)
pd.DataFrame(new_x.toarray(), columns=bigram_vectorizer.get_feature_names())
```

Out[58]:

| | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 11 | 12 | ... | you | yo |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 3 | became | boarders | difficult | immigrants | is | language | not | style | million | | come | c |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | |
| 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | |
| 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | |
| 335 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | |
| 336 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | |
| 337 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | |
| 338 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | |
| 339 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | |

340 rows × 5209 columns

In [64]:
```
xtrain, xtest, ytrain, ytest = train_test_split(new_x, alphas,
random_state=1)

rbf = ConstantKernel(1.0) * RBF(length_scale=1.0)
gpr = GaussianProcessRegressor(kernel=rbf, alpha=1e-8)
gpr.fit(xtrain.toarray(), ytrain)
```

Out[64]:
```
GaussianProcessRegressor(alpha=1e-08, copy_X_train=True,
                         kernel=1**2 * RBF(length_scale=1),
                         n_restarts_optimizer=0, normalize_y=False,
                         optimizer='fmin_l_bfgs_b', random_state=None)
```

In [65]:
```
m, c = gpr.predict(xtest.toarray(), return_cov=True)
```

In [66]:
```
np.corrcoef(ytest, m)
```

Out[66]:
```
array([[1.        , 0.45842178],
       [0.45842178, 1.        ]])
```

In [67]:
```
#BIAGRAM VECTORIZER REDUCED CORRELATION COEFFICIENT FROM 0.6328 to 0.4555
```

In [ ]:
```

```