Prediction of Europe’s “Wellbeing” Using Economic Data Analysis

Ryan Gibertini

Caralyn Clark

1. Introduction

In the 1964 supreme court case Jacobellis v Ohio, then justice Potter Stewart coined a famous phrase when trying to pin down what made something obscene. “I know it when I see it,” has since become a colloquial phrase used to describe the more slippery words in the English language. I also believe it is a perfect fit when trying to define happiness, we know it when we see it. The following report will explore a few socioeconomic and sociopolitical factors that may have an influence in certain European country’s self-reported happiness scores, as measured on a scale of one to ten.

1. Data

RESPONSE VARIABLE:

Wellbeing (WELL) – Each person who completed the survey in 2016.

EXPLANATORY VARIABLES:

GDP per capita (GDPK) – The total GDP of a country divided by its population given in USD.

Life expectancy (LIFE) – The average period, in year, someone is expected to live.

Democratic Dummy (DD) – If a country is not democratic, value is 0. If a country is democratic, value is 1.

Unemployment (UMEPY) – The amount of people who are unemployed divided by the total labor force.

DATA SOURCES:

1. [www.statista.com](http://www.statista.com)
2. [www.europeansocialsurvery.org](http://www.europeansocialsurvery.org)
3. [www.imf.org](http://www.imf.org)
4. www.worldlifeexpectancy.com
5. www.eiu.com
6. Plots of Dependent Variables & Independent Variables

A picture containing screenshot

Description automatically generated

1. Model specification & Regression Estimation Results

Unrestricted Model Results:

==============================================================================

Dep. Variable: y No. Observations: 310307

Model: GLM Df Residuals: 310302

Model Family: Poisson Df Model: 4

Link Function: log Scale: 1.0000

Method: IRLS Log-Likelihood: -6.7885e+05

Date: Fri, 05 Apr 2019 Deviance: 1.8094e+05

Time: 20:21:03 Pearson chi2: 1.52e+05

No. Iterations: 4 Covariance Type: nonrobust

==============================================================================

coef std err z P>|z| [0.025 0.975]

------------------------------------------------------------------------------

const 1.8852 0.004 516.265 0.000 1.878 1.892

x1 2.774e-06 5.31e-08 52.216 0.000 2.67e-06 2.88e-06

x2 -0.0006 4.95e-05 -12.118 0.000 -0.001 -0.001

x3 0.0320 0.002 16.396 0.000 0.028 0.036

x4 0.0045 0.000 16.616 0.000 0.004 0.005

==============================================================================

Restricted Model Results (GDPK and UMEPY covariates only):

==============================================================================

Dep. Variable: y No. Observations: 310307

Model: GLM Df Residuals: 310304

Model Family: Poisson Df Model: 2

Link Function: log Scale: 1.0000

Method: IRLS Log-Likelihood: -6.7906e+05

Date: Fri, 05 Apr 2019 Deviance: 1.8136e+05

Time: 20:27:24 Pearson chi2: 1.53e+05

No. Iterations: 4 Covariance Type: nonrobust

==============================================================================

coef std err z P>|z| [0.025 0.975]

------------------------------------------------------------------------------

const 1.8386 0.002 841.412 0.000 1.834 1.843

x1 3.181e-06 3.44e-08 92.552 0.000 3.11e-06 3.25e-06

x2 0.0046 0.000 18.145 0.000 0.004 0.005

==============================================================================

1. Checking the Adequacy of the Model

Mean of Residuals (Unnormalized): -0.03873873375067684

Standard Deviation of Residuals (Unnormalized): 0.7635176686625287

Vs.

Standardized Residual Mean: 5.861902578884319e-18

Standardized Residual Standard Deviation: 1.0

1. Inference & Interpretations of Every Significant Coefficients

Given our unrestricted model, all our covariates are statistically significant at the 5% level. When X1 (GDPK) changes by one unit, Y (WELL) rises by 2.774e-06 units. When X2 (LIFE) changes by one unit, Y decreases by -0.0006 units. When X3 (DD) equals one, Y rises by 0.0320 unit. When X4 (UMEPY) changes by one unit, Y rises by 0.0045 units. These results prove that variables such as GDP per capita, life expectancy, unemployment and democratic society do positively and negatively affect the wellbeing of a country’s citizens.

1. Summary

From the above multi-variate GLM regression, we were able to determine that all of our covariates are statistically significant to the 95% level and satisfy the reason for doing so laid out in the introduction. We were able to explore a few different social, economic, and political reasons that, we know now, do have an effect on the well-being, or happiness, of a given European country. From this, we can conclude that the typical intuition of an economists would be correct in this scenario. That is, better off countries with democracies tend to be happier, on average, than worse off countries with flawed regimes.

1. Appendix of Code in R and Python

R Code for the Plots:

data1 <- read.csv('C:/ECO4422/data/wellbeingEU.csv', header=TRUE)

par(mfrow=c(2,2))

plot(data1$GDPK,data1$WELL,

type = "p",

main = "GDP per capita versus Wellbeing",

ylab = "Wellbeing",

xlab = "GDP per capita",

col = "blue")

plot(data1$LIFE,data1$WELL,

type = "p",

main = "Life Expectancy versus Wellbeing",

xlab = "Life Expectancy",

ylab = "Wellbeing",

col = "red")

plot(data1$DD,data1$WELL,

type = "p",

main = "Democracy versus Wellbeing",

xlab = "Democracy",

ylab = "Wellbeing",

col = "purple")

plot(data1$UMEPY,data1$WELL,

type = "p",

main = "Unemployment versus Wellbeing",

xlab = "Unemployment",

ylab = "Wellbeing",

col = "green")

Python Code for the Models:

from pandas import read\_csv

import numpy as np

from pandas import DataFrame

from numpy import array

import statsmodels.api as sm

data1 = read\_csv('C:/ECO4422/data/WellbeingEU.csv', delimiter=",")

Y = np.transpose(np.array([data1.WELL]))

X = np.transpose(np.array([data1.GDPK, data1.LIFE, data1.DD, data1.UMEPY]))

X = sm.add\_constant(X)

model1 = sm.GLM(endog=Y, exog=X, family=sm.families.Poisson())

results = model1.fit()

beta\_hats = results.params

print(results.summary())

Y = np.transpose(np.array([data1.WELL]))

X = np.transpose(np.array([data1.GDPK, data1.UMEPY]))

X = sm.add\_constant(X)

model1 = sm.GLM(endog=Y, exog=X, family=sm.families.Poisson())

results = model1.fit()

beta\_hats = results.params

print(results.summary())

print(np.mean(results.resid\_deviance))

print(np.std(results.resid\_deviance))

mean = np.mean(results.resid\_deviance)

std = np.std(results.resid\_deviance)

resid = results.resid\_deviance

stand\_resid = (((resid) - (mean))/std)

np.mean(stand\_resid)

np.std(stand\_resid)

print(np.mean(stand\_resid))

print(np.std(stand\_resid))