Lab 4: **Rmarkdown**, package **dplyr**, package **stargazer**, and regression plots

Ngoc Phan

September 28, 2018

Intro to RMarkdown

- Create Markdown Document
- Knit to HTML/PDF/Word
- Headers
- **Bold**
- *Italic*
- Bullet points
- Embedded link
- R code chunks: Labels, options

Manipulating/cleaning data with dplyr

As I showed you in the last lab, cleaning data with base R can be tricky and confusing. However, the beauty of R is that you can import packages that make these tasks much more straightforward.

```
library(readstata13)
happy <- read.dta13("happy_planet.dta")
colnames(happy)
```

```
##  [1] "code"             "country"          "region"
##  [4] "lifesat010"       "lifeexpyears"     "footprint"
##  [7] "hly"              "hpi"              "hpirank"
## [10] "gdppercapitappp" "hdi"              "population"
## [13] "reg1"             "West"             "MiddleEast"
## [16] "Africa"           "S_Asia"           "E_Asia"
## [19] "EEuropeUssr"      "LatinAmerica"
```

```
#install.packages("dplyr")
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

In the following subsections, we are going to contrast how we do certain tasks in base R and the way we do it in **dplyr**.

Select: Keeping and dropping variables

```r
#Base R
happy1 <- happy[, c("country", "lifesat010", "hdi", "gdppercapitappp")]
colnames(happy1)
```

```
## [1] "country"         "lifesat010"      "hdi"             "gdppercapitappp"
```

```r
#Dplyr
happy2 <- select(happy, country, lifesat010, hdi)
happy2 <- select(happy, country:population)

#Drop variables
happy3 <- select(happy, -region)
```

Filter: Returning rows with matching conditions

```r
#Base R
happy4 <- happy[happy$Africa == 1, ]


#Dplyr
happy4 <- filter(happy, Africa == 1)
happy5 <- filter(happy, population > 5)
```

Dropping missing values

```r
#summary(happy)
#is.na(happy$hdi)

#filter(happy, is.na(hdi))
#filter(happy, !is.na(hdi))

happy <- filter(happy, !is.na(hdi))
```

Arrange: Sorting data

```r
#Create a dataset of Western countries, keeping only four variables
west <- filter(happy, West == 1)
west <- select(west, country, lifesat010, hdi, population)

#Base R
#order(west$lifesat010)

west1 <- west[order(west$lifesat010), ]

#head(west)
west2 <- west[c(6, 4, 2) , ]
#west2
```

```
#Dplyr
#arrange(west, lifesat010)
#arrange(west, -lifesat010)
#arrange(west, desc(lifesat010))

#arrange(west, hdi, population)
```

Mutate: Creating new variables

```
#Base R
west$pop <- west$population*1000000
#west

#Dplyr
west <- select(west, -pop)
west <- mutate(west, pop = population*1000000)
```

Nice regression output with Stargazer

```
m1 <- lm(lifesat010 ~ hdi + lifeexpyears, data = happy)
summary(m1)
```

```
##
## Call:
## lm(formula = lifesat010 ~ hdi + lifeexpyears, data = happy)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.00395 -0.45875  0.01749  0.48453  1.61202
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.11969    0.46217  -0.259 0.796049
## hdi           3.59181    0.93163   3.855 0.000176 ***
## lifeexpyears  0.05038    0.01490   3.381 0.000938 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7303 on 138 degrees of freedom
## Multiple R-squared:  0.7234, Adjusted R-squared:  0.7194
## F-statistic: 180.4 on 2 and 138 DF,  p-value: < 2.2e-16
```

```
#install.packages("stargazer")
library(stargazer)
stargazer(m1, title = "Regression of Life Statisfaction on HDI")
```

% Table created by stargazer v.5.2.2 by Marek Hlavac, Harvard University. E-mail: hlavac at fas.harvard.edu
% Date and time: Thu, Oct 04, 2018 - 7:26:23 PM

Table 1: Regression of Life Statisfaction on HDI

	Dependent variable:
	lifesat010
hdi	3.592***
	(0.932)
lifeexpyears	0.050***
	(0.015)
Constant	−0.120
	(0.462)
Observations	141
R^2	0.723
Adjusted R^2	0.719
Residual Std. Error	0.730 (df = 138)
F Statistic	180.426*** (df = 2; 138)
Note:	*p<0.1; **p<0.05; ***p<0.01

```r
stargazer(m1, title = "Regression of Life Statisfaction on HDI",
          dep.var.labels = "Life Satisfaction",
          covariate.labels = c("Human Development Index", "GDP per capita PPP"))
```

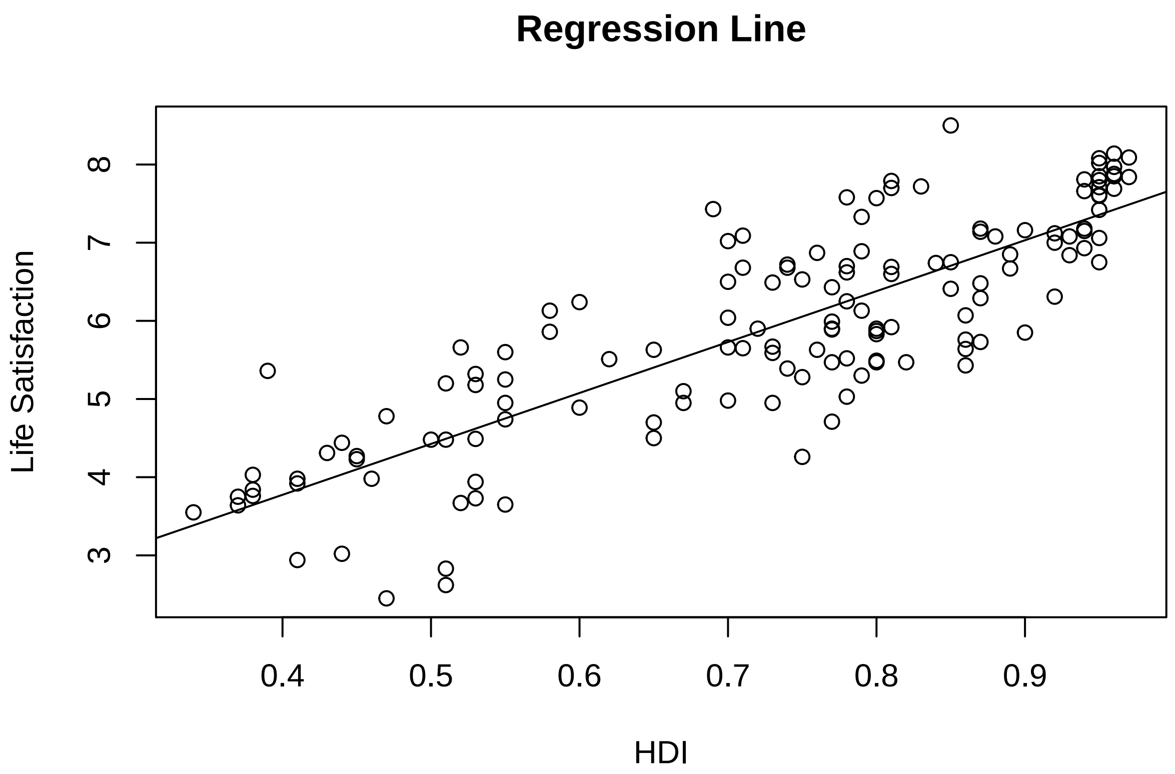
% Table created by stargazer v.5.2.2 by Marek Hlavac, Harvard University. E-mail: hlavac at fas.harvard.edu
% Date and time: Thu, Oct 04, 2018 - 7:26:23 PM

Table 2: Regression of Life Statisfaction on HDI

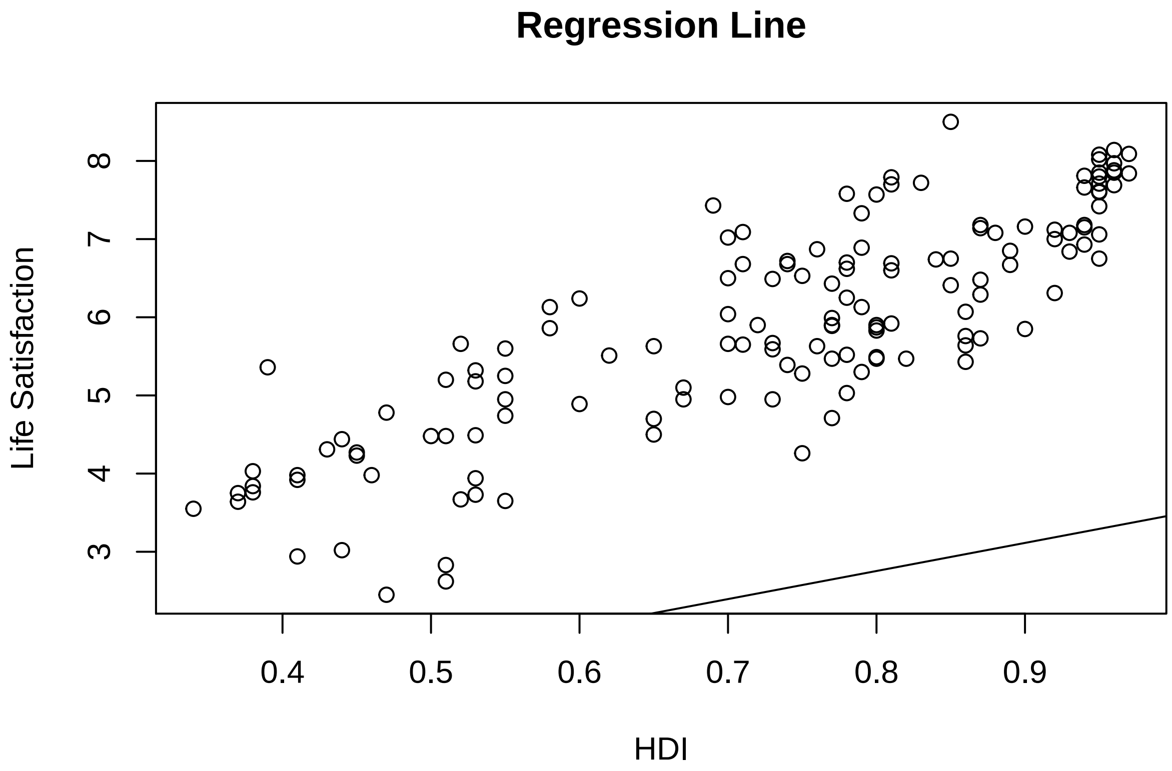
	Dependent variable:
	Life Satisfaction
Human Development Index	3.592***
	(0.932)
GDP per capita PPP	0.050***
	(0.015)
Constant	−0.120
	(0.462)
Observations	141
R^2	0.723
Adjusted R^2	0.719
Residual Std. Error	0.730 (df = 138)
F Statistic	180.426*** (df = 2; 138)
Note:	*p<0.1; **p<0.05; ***p<0.01

Plot regression line

```
m2 <- lm(lifesat010 ~ hdi, data = happy)
plot(happy$hdi, happy$lifesat010, main = "Regression Line",
     xlab = "HDI", ylab = "Life Satisfaction")
abline(m2)
```



```
#?abline
#m1$coefficients

plot(happy$hdi, happy$lifesat010, main = "Regression Line",
     xlab = "HDI", ylab = "Life Satisfaction")
abline(m1$coefficients["(Intercept)"], m1$coefficients["hdi"])
```



Plot coefficients and confidence intervals

```
happy <- mutate(happy, hdi100 = hdi*100)
m3 <- lm(lifesat010 ~ hdi100 + gdppercapitappp + lifeexpyears, data = happy)
summary(m3)
```

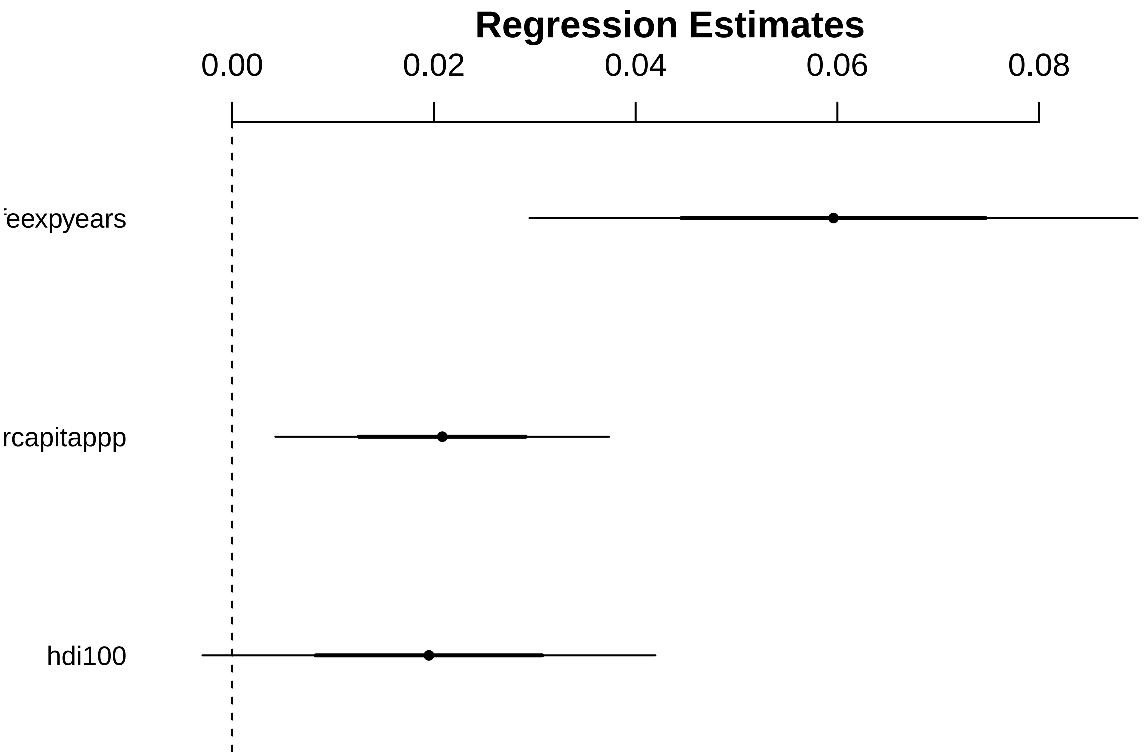
```
##
## Call:
## lm(formula = lifesat010 ~ hdi100 + gdppercapitappp + lifeexpyears,
##     data = happy)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.06713 -0.46186  0.00688  0.41733  1.73494
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     0.215091   0.472590   0.455 0.649734
## hdi100          0.019506   0.011228   1.737 0.084598 .
## gdppercapitappp 0.020826   0.008274   2.517 0.012987 *
## lifeexpyears    0.059617   0.015072   3.955 0.000122 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.7166 on 137 degrees of freedom
## Multiple R-squared:  0.7356, Adjusted R-squared:  0.7298
## F-statistic:   127 on 3 and 137 DF,  p-value: < 2.2e-16
```

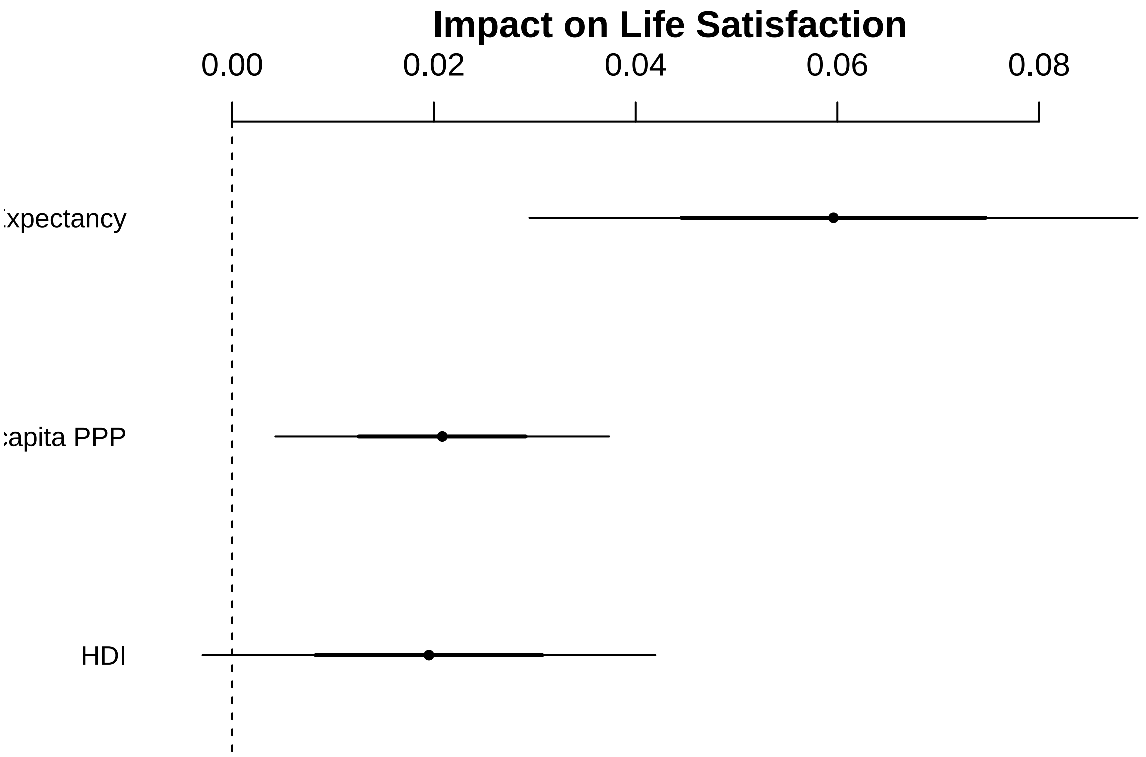
```
#install.packages("arm")
library(arm)
```

```
## Loading required package: MASS

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select

## Loading required package: Matrix

## Loading required package: lme4

##
## arm (Version 1.10-1, built: 2018-4-12)

## Working directory is D:/Dropbox/WORK/Courses (now)/POLI630 Intro to Empirics/ps630 public material/Du
```

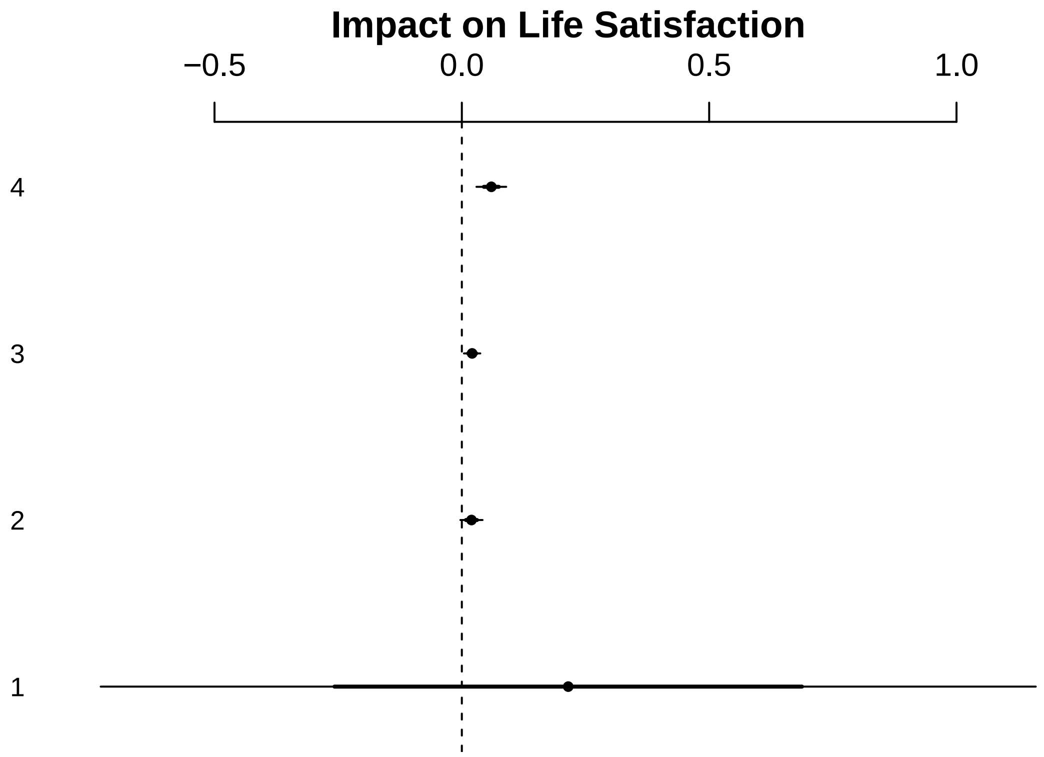
```
coefplot(m3)
```



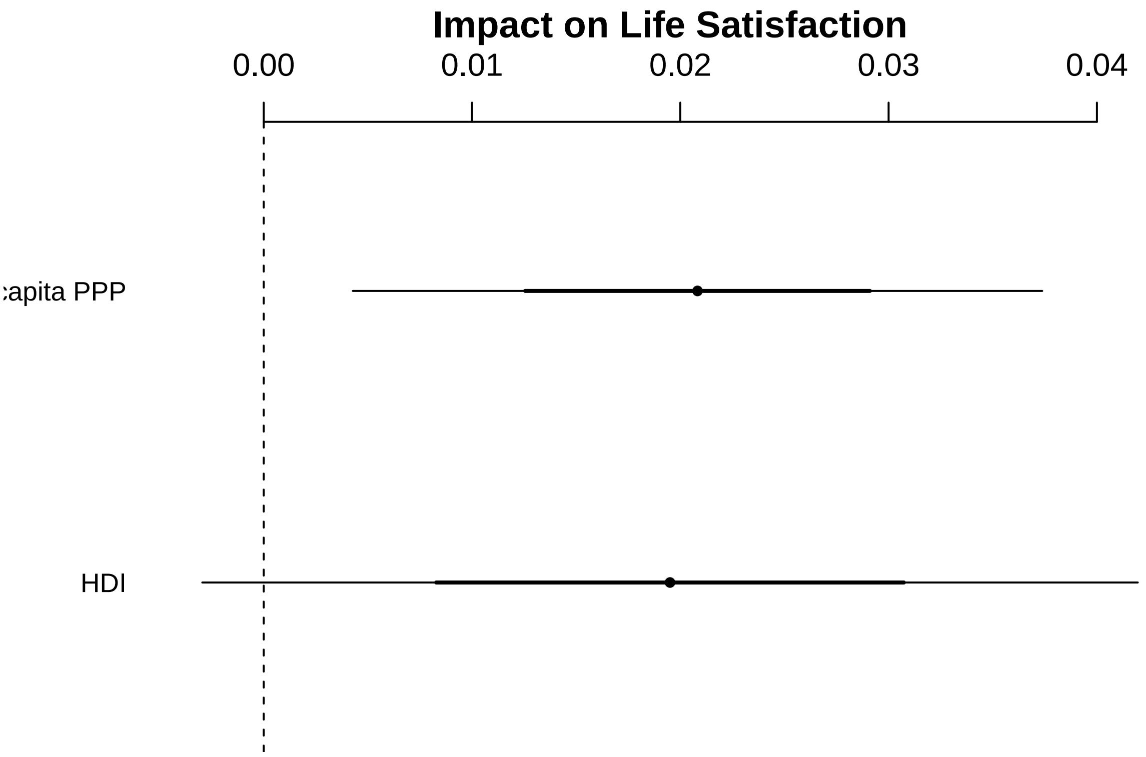
```
coefplot(m3, main = "Impact on Life Satisfaction",
         varnames = c("Intercept", "HDI", "GDP per capita PPP", "Life Expectancy"))
```



```
#summary(m3)$coefficients[, 1]
#summary(m3)$coefficients[, 2]
#summary(m3)$coefficients

coefplot(summary(m3)$coefficients[, 1],
         summary(m3)$coefficients[, 2],
         main = "Impact on Life Satisfaction")
```



```
coefplot(summary(m3)$coefficients[c("hdi100", "gdppercapitappp"), 1],
         summary(m3)$coefficients[c("hdi100", "gdppercapitappp"), 2],
         varnames = c("HDI", "GDP per capita PPP"),
         main = "Impact on Life Satisfaction")
```


Impact on Life Satisfaction
0.00
0.01
0.02
0.03
0.04
:apita PPP
HDI