Visualizing Regression Results with dotwhisker: Groupwork

Yeiyoung Choo

Spring 2023

For group work, we will practice how to generate dot-and-whisker plots with results stored in regression objects. This is the first approach we learned in class.

Load Packages

```
library(tidyverse)
library(dotwhisker)
```

Load Data

Let's load the data we need for regression analysis. The data come from a study by Aslaksen (2010) on the relationship between oil income and democracy.

```
as2010 <- readRDS("aslaksen2010.rds")  # data abridged and prepped for exercise
```

1 Understanding Data and Variables

a) What are the variables related to oil? How many are there? (We will come back to working with these variables.)

b) What index is used to measure the country's level of democracy?

c) We have data on population and real GDP per capita. Using the variables you need, create two variables that capture logged values of these estimates, calling them `lpop` and `lrgdppc`. *Hint*: Use `mutate()`.

```
# a) oil_wb, oilproduction, oilprice. 3 variables
# (not 2!).  b) Political Rights Index.  c)
as2010 <- as2010 %>%
    mutate(lpop = log(population), lrgdppc = log(rgdppc))
```

2 Running Regression

Our outcome variable is `pr_lead`. This is a lead variable generated to capture the value of `pr` in the following year. Recall that `pr` measures democracy based on the Political Rights Index. Thus, our regression analysis evaluates the impact of some predictors on the level of democracy in the following year. (Alternatively, we can generate lag variables for predictors to capture their values in the previous year.)

As you may have guessed, our key predictor variable will capture the country's income from oil extraction. We will use our data to construct this variable. In our data, `oil_wb` is the volume of oil extraction and `oilprice` is the average world price of crude oil.

a) Create a variable called `oilshare` that measures oil income as a share (%) of the country's GDP. First, multiply the volume of oil extraction (`oil_wb`) by the price of oil (`oilprice`). Divide it by GDP. Multiply by 100 to obtain a percentage (%) estimate for oilshare. *Hint*: Use `mutate()`.

```
as2010 <- as2010 %>%
    mutate(oilshare = 100 * (oil_wb * oilprice/gdp))
```

Bonus: Replace NAs with 0 where the country's oil extraction according to `oil_wb` is 0. (Check the answer key right below. You need to finish this step before moving onto regression analysis.)

```
as2010$oilshare[as2010$oil_wb == 0] <- 0
```

b) Run a linear regression, using `pr` and `oilshare` as predictors. Remember our outcome variable is `pr_lead`. Store the regression model as an object called `n1`. This will be our Model 1. Use `summary()` to check your regression results. *Hint*: The basic syntax for linear regression is `lm(y~x,data=data)`.

```
n1 <- lm(pr_lead ~ pr + oilshare, data = as2010)
summary(n1)  #check results
```

```
##
## Call:
## lm(formula = pr_lead ~ pr + oilshare, data = as2010)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.98518 -0.01966 -0.00455  0.01476  0.81061
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.0265457  0.0027657   9.598  < 2e-16 ***
## pr           0.9586945  0.0042674 224.657  < 2e-16 ***
## oilshare    -0.0003052  0.0001051  -2.903  0.00371 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1006 on 4394 degrees of freedom
##   (1238 observations deleted due to missingness)
## Multiple R-squared:  0.9243, Adjusted R-squared:  0.9243
## F-statistic: 2.684e+04 on 2 and 4394 DF,  p-value: < 2.2e-16
```

c) Run another linear regression, using `pr`, `oilshare`, `lpop`, and `lrgdppc` as predictors. Store the regression object as `n2`. This is our Model 2. Check your regression results.

```
n2 <- lm(pr_lead ~ pr + oilshare + lpop + lrgdppc,
    data = as2010)
summary(n2)  #check results
```

```
##
## Call:
## lm(formula = pr_lead ~ pr + oilshare + lpop + lrgdppc, data = as2010)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.98118 -0.02211 -0.00128  0.01230  0.80286
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.0847748  0.0206169  -4.112 4.00e-05 ***
## pr           0.9318744  0.0056244 165.684  < 2e-16 ***
## oilshare    -0.0006740  0.0001210  -5.568 2.73e-08 ***
## lpop         0.0012616  0.0009902   1.274    0.203    
## lrgdppc      0.0129660  0.0017616   7.360 2.19e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.09944 on 4238 degrees of freedom
##   (1392 observations deleted due to missingness)
## Multiple R-squared:  0.9262, Adjusted R-squared:  0.9261 
## F-statistic: 1.329e+04 on 4 and 4238 DF,  p-value: < 2.2e-16
```

d) Lastly, adding `educ` and `open` to the list of predictors used for Model 2, run a linear regression and store it as an object called `n3`. This is our Model 3. Check your regression results.

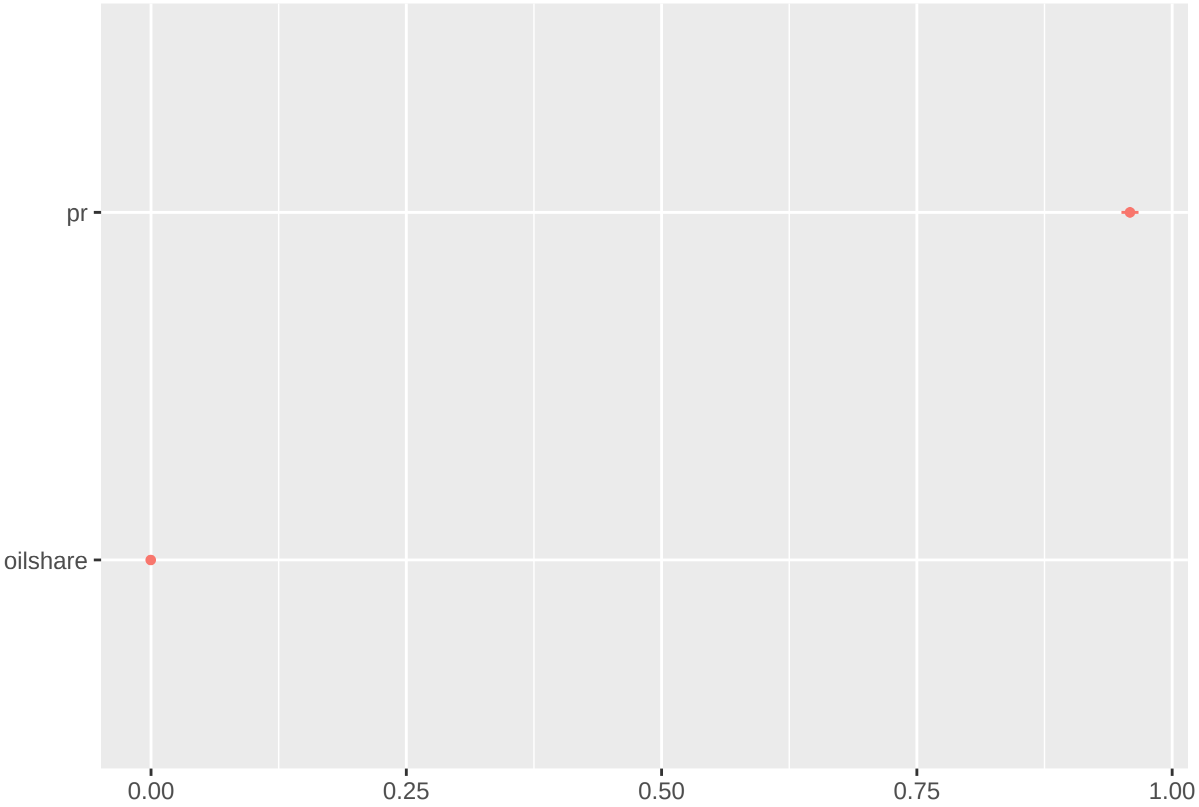
```
n3 <- lm(pr_lead ~ pr + oilshare + lpop + lrgdppc +
    educ + open, data = as2010)
summary(n3)  #check results
```

```
## 
## Call:
## lm(formula = pr_lead ~ pr + oilshare + lpop + lrgdppc + educ + 
##     open, data = as2010)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -0.97980 -0.02469 -0.00333  0.01592  0.78150 
## 
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  3.290e-03  3.348e-02   0.098 0.921727    
## pr           9.076e-01  7.223e-03 125.648  < 2e-16 ***
## oilshare    -5.974e-04  1.684e-04  -3.547 0.000395 ***
## lpop        -1.754e-03  1.429e-03  -1.228 0.219599    
## lrgdppc      8.368e-03  3.143e-03   2.662 0.007804 ** 
## educ         4.818e-03  1.130e-03   4.263 2.08e-05 ***
## open        -1.119e-04  4.727e-05  -2.366 0.018030 *  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.1024 on 3143 degrees of freedom
##   (2485 observations deleted due to missingness)
## Multiple R-squared:  0.9188, Adjusted R-squared:  0.9186 
## F-statistic:  5928 on 6 and 3143 DF,  p-value: < 2.2e-16
```

3 Generating a Dot-and-Whisker Plot

a) Besides presenting a regression table, you want to visualize your regression results with a coefficient plot. Generate a dot-and-whisker plot for Model 1, using the regression object you saved above.

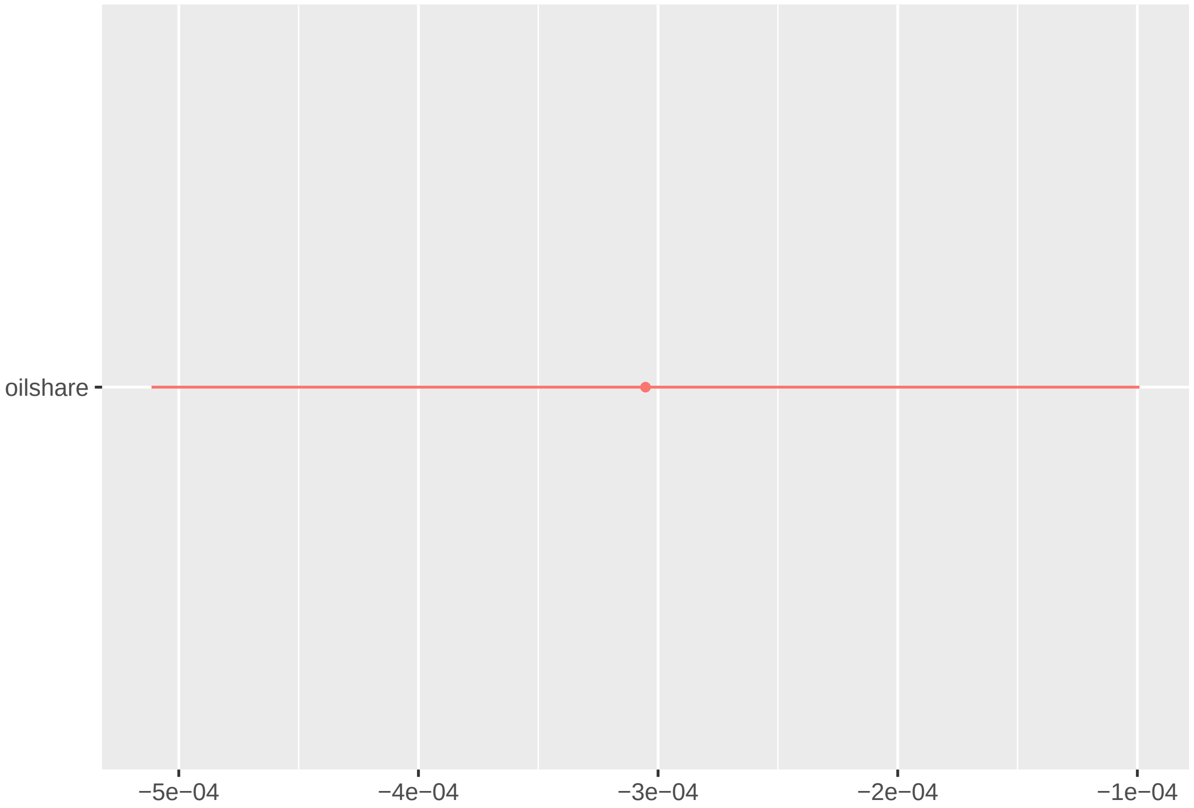
```
dwplot(n1)
```



As we include `pr` as a predictor for our outcome `pr_lead`, the coefficient estimates on other predictors are quite small relative to the estimate on `pr`. (This is understandable, as we expect the level of democracy in a certain year to be a strong predictor of the level of democracy in the following year.)

See what happens when you generate the plot again with the command: ‘dwplot(n1,vars_order=“oilshare).

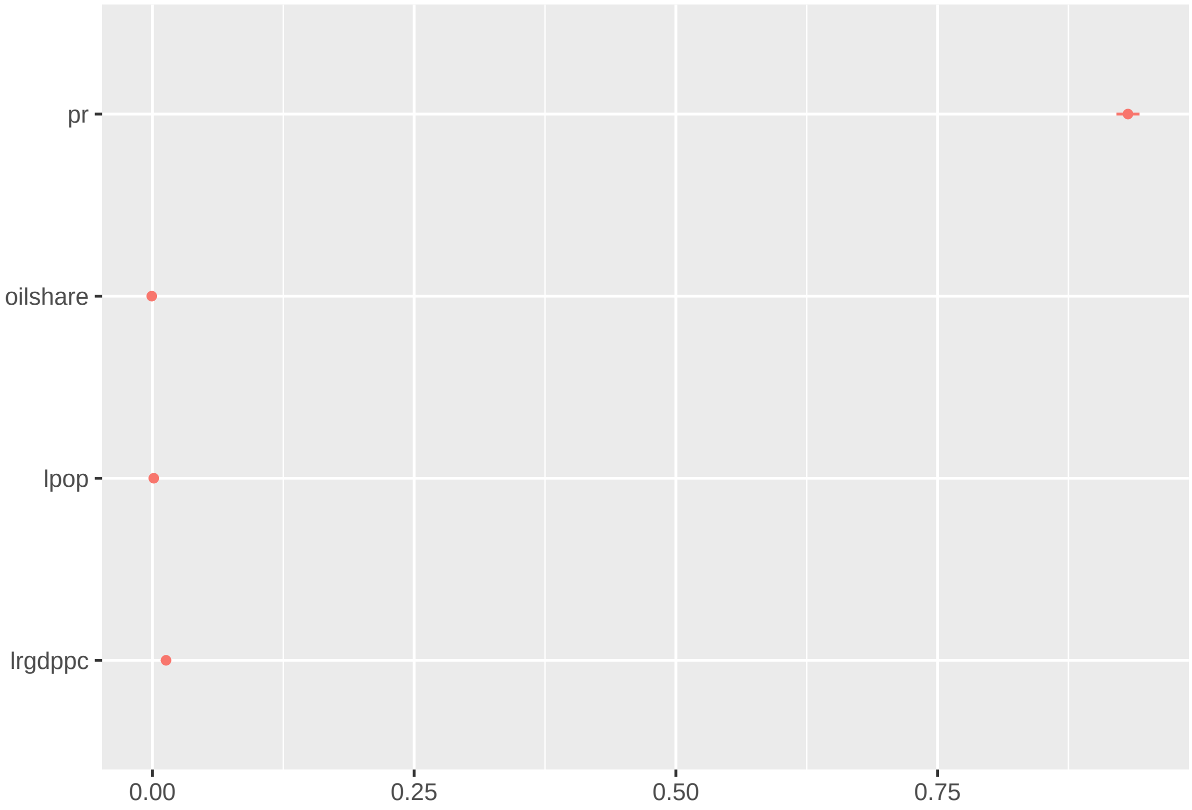
```
dwplot(n1, vars_order = "oilshare")
```



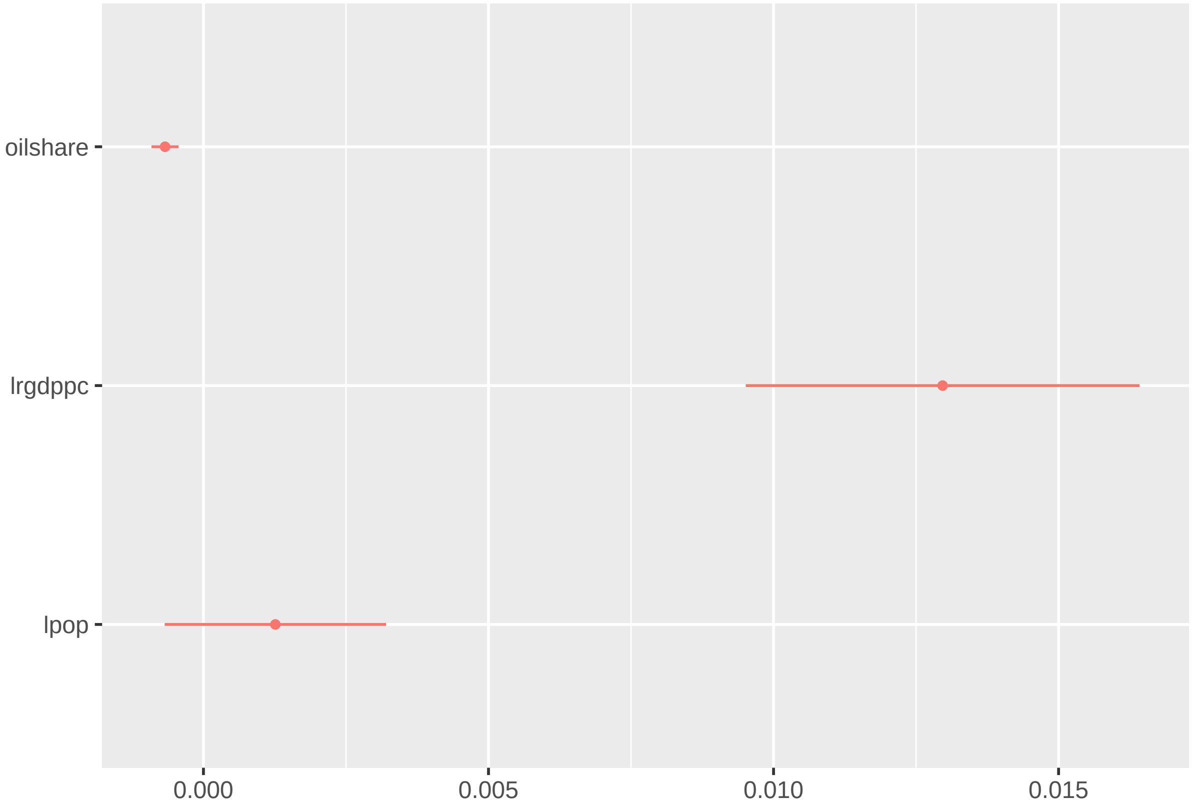
b) Generate a dot-and-whisker plot for Model 2. And another plot, omitting `pr` as you did above.

c) Recall that our outcome variable is `pr_lead`, which is the level of democracy in the following year. According to the plot you just created, what are the predictors used in this regression model? (Identify these variables briefly, no need to fully describe them.) Aside from `pr`, what is the predictor variable commonly used across Model 1 and Model 2?

```
dwplot(n2)
```



```
dwplot(n2, vars_order = c("oilshare", "lrgdppc", "lpop"))
```

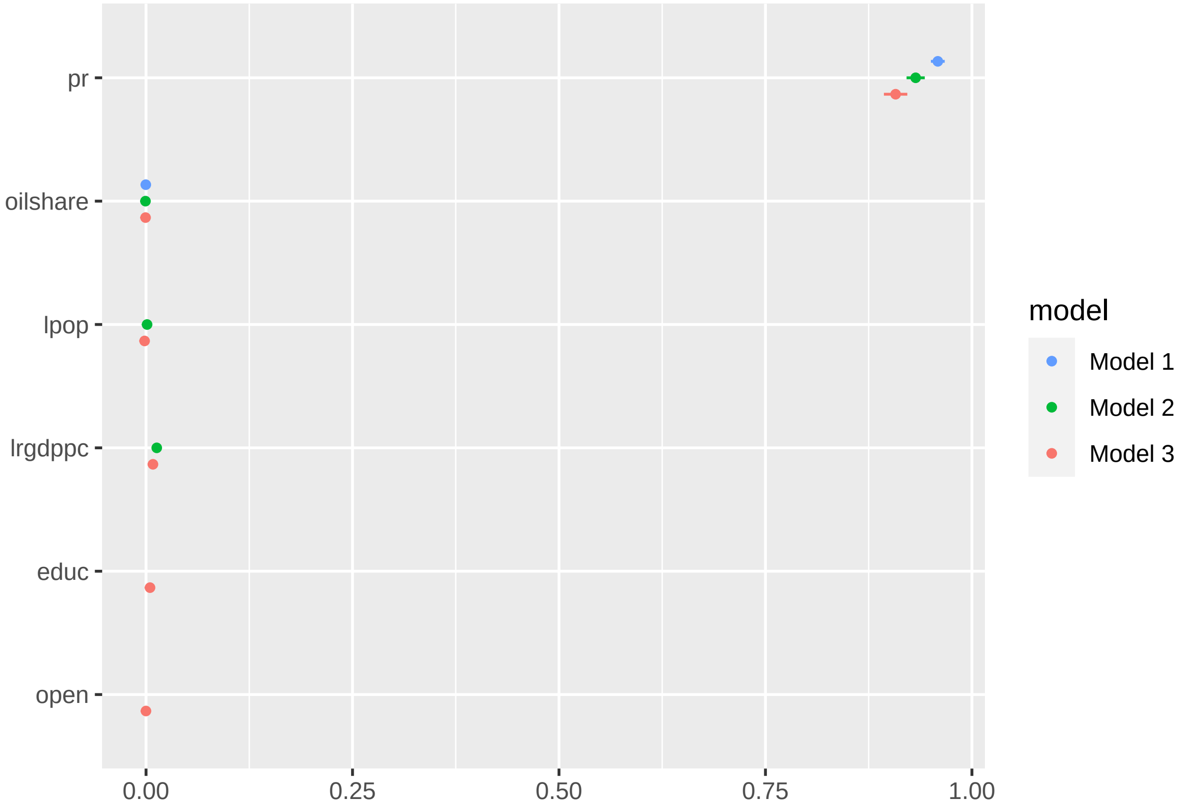


```
# predictors used in Model 2: pr, oilshare,
# lrgdppc, lpop.  besides pr, both models used:
# oilshare.
```

4 Plotting Results from Multiple Regression Models

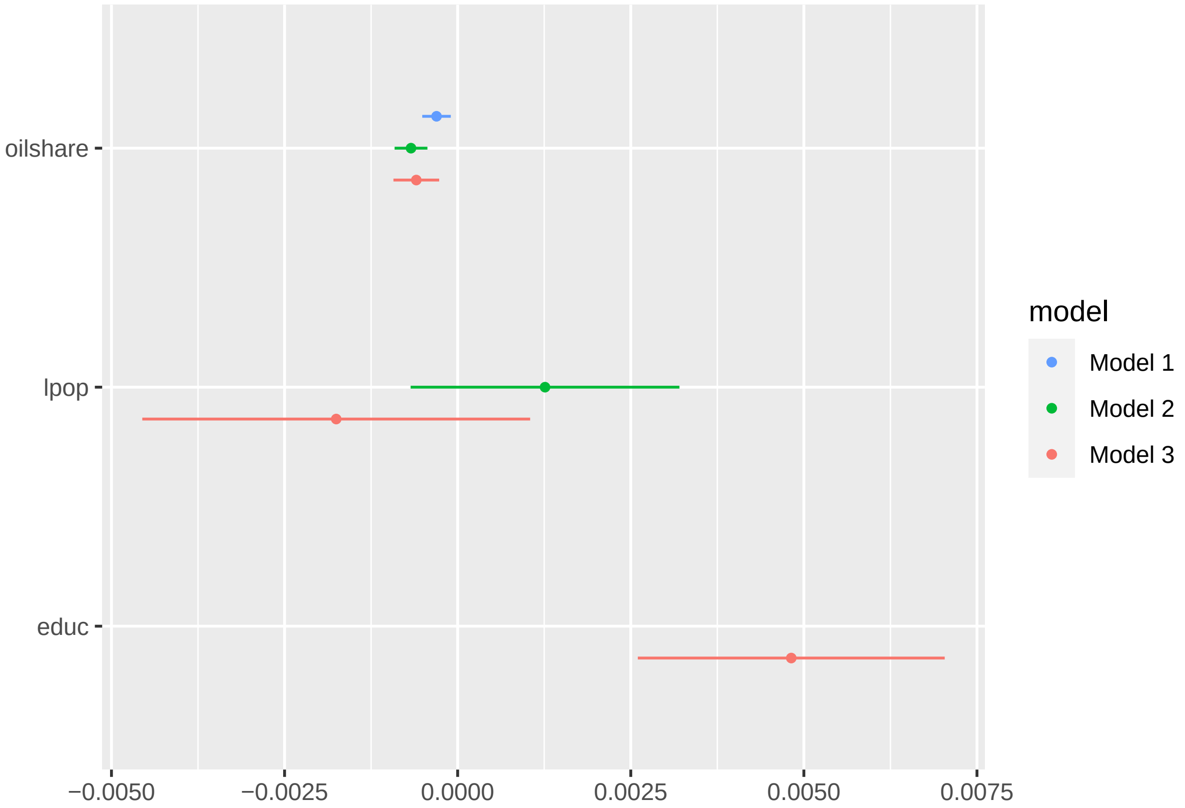
a) You want to display results from all three regression models in one plot. Using the regression objects `n1`, `n2`, `n3`, generate a dot-and-whisker plot for all models.

```
dwplot(list(n1, n2, n3))
```



b) The plot has all variables, but you only want to display results for several key variables. Modify your plot by displaying results for the following variables only: `oilshare`, `lpop`, and `educ`.

```
dwplot(list(n1, n2, n3), vars_order = c("oilshare",
    "lpop", "educ"))
```



c) Based on the plot, what does our analysis suggest about the impact of oil income on democracy? Is there a negative or positive effect? Why would you say so?

5 Customizing the Plot

Start with `plot3 <-` to store your plot as an object. We will further customize the plot above by doing the following.

a) Set the model order: Model 1, Model 2, Model 3.

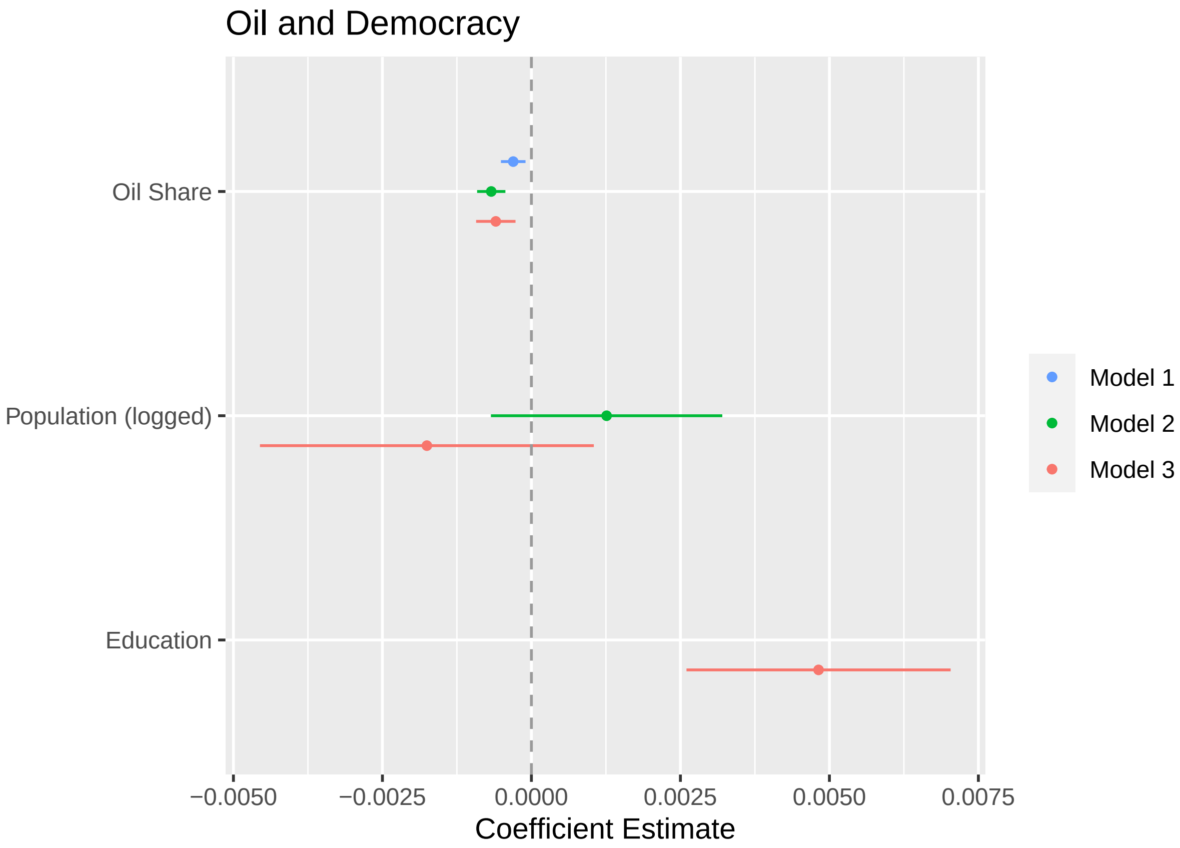
b) Keeping the same order, rename predictor variables: Oil Share, Population (logged), Education.

c) Label x-axis "Coefficient Estimate" and keep y-axis blank.

d) Add a title: "Oil and Democracy" *Bonus*: Add a dotted vertical line where the x-intercept is 0.

```
plot3 <- dwplot(list(n1, n2, n3), model_order = c("Model 1",
    "Model 2", "Model 3"), vars_order = c("oilshare",
    "lpop", "educ")) %>%
    relabel_predictors(c(oilshare = "Oil Share", lpop = "Population (logged)",
        educ = "Education")) + xlab("Coefficient Estimate") +
    ylab("") + geom_vline(xintercept = 0, colour = "grey60",
    linetype = 2) + ggtitle("Oil and Democracy") +
    theme(legend.title = element_blank())  #optional - remove legend title

plot3
```



6 Saving Your Works

a) Save your plot as a PNG file using `ggsave()`.

```
ggsave("dwplot1_oildem.png", plot3, width = 6.5, height = 4.5,
    dpi = 400)
# ggsave('your filepath/your filename.png'...)
```

b) After loading data, we have created additional variables to conduct regression analysis. Save the final version of your dataset as a RDS file called `reg_aslaksen2010.rds`. We will use it for homework!