An Interpretable Linear Regression Demo

Hendra Bunyamin

4/8/2021

We set the **working directory**.

```
setwd("/home/hbunyamin/Projects/2021-nuni-it-online-seminar/tahun-2021/codes/demo-interpretable-ml")
```

Let's load the dataset (Molnar 2019).

```
load("bike.RData")
```

We view the first five rows.

```
head(bike)
```

```
##   season   yr mnth    holiday weekday     workingday weathersit     temp
## 1 SPRING 2011  JAN NO HOLIDAY     SAT NO WORKING DAY      MISTY 8.175849
## 2 SPRING 2011  JAN NO HOLIDAY     SUN NO WORKING DAY      MISTY 9.083466
## 3 SPRING 2011  JAN NO HOLIDAY     MON    WORKING DAY       GOOD 1.229108
## 4 SPRING 2011  JAN NO HOLIDAY     TUE    WORKING DAY       GOOD 1.400000
## 5 SPRING 2011  JAN NO HOLIDAY     WED    WORKING DAY       GOOD 2.666979
## 6 SPRING 2011  JAN NO HOLIDAY     THU    WORKING DAY       GOOD 1.604356
##       hum windspeed  cnt days_since_2011
## 1 80.5833 10.749882  985               0
## 2 69.6087 16.652113  801               1
## 3 43.7273 16.636703 1349               2
## 4 59.0435 10.739832 1562               3
## 5 43.6957 12.522300 1600               4
## 6 51.8261  6.000868 1606               5
```

We summarize the `bike` dataset as follows:

```
summary(bike)
```

```
##     season       yr           mnth           holiday      weekday
##  SPRING:181   2011:365   JAN    : 62   NO HOLIDAY:710   SUN:105
##  SUMMER:184   2012:366   MAR    : 62   HOLIDAY   : 21   MON:105
##  FALL  :188              MAY    : 62                    TUE:104
##  WINTER:178              JUL    : 62                    WED:104
##                          AUG    : 62                    THU:104
##                          OKT    : 62                    FRI:104
##                          (Other):359                    SAT:105
##           workingday            weathersit       temp            hum
##  NO WORKING DAY:231   GOOD           :463   Min.   :-5.221   Min.   : 0.00
##  WORKING DAY   :500   MISTY          :247   1st Qu.: 7.843   1st Qu.:52.00
##                       RAIN/SNOW/STORM: 21   Median :15.422   Median :62.67
##                                             Mean   :15.283   Mean   :62.79
##                                             3rd Qu.:22.805   3rd Qu.:73.02
##                                             Max.   :32.498   Max.   :97.25
```

```
##
##    windspeed           cnt       days_since_2011
##  Min.   : 1.500   Min.   :  22   Min.   :  0.0
##  1st Qu.: 9.042   1st Qu.:3152   1st Qu.:182.5
##  Median :12.125   Median :4548   Median :365.0
##  Mean   :12.763   Mean   :4504   Mean   :365.0
##  3rd Qu.:15.625   3rd Qu.:5956   3rd Qu.:547.5
##  Max.   :34.000   Max.   :8714   Max.   :730.0
##
```

We extract the features columns just like in the slides as follows:

```r
bike_to_interpreted <- bike[c("cnt", "season", "holiday", "days_since_2011",
                         "workingday", "weathersit", "temp", "hum", "windspeed")]
```

Let's see the first five rows in `bike_to_interpreted`.

```r
head(bike_to_interpreted)
```

```
##    cnt season    holiday days_since_2011     workingday weathersit     temp
## 1  985 SPRING NO HOLIDAY               0 NO WORKING DAY      MISTY 8.175849
## 2  801 SPRING NO HOLIDAY               1 NO WORKING DAY      MISTY 9.083466
## 3 1349 SPRING NO HOLIDAY               2    WORKING DAY       GOOD 1.229108
## 4 1562 SPRING NO HOLIDAY               3    WORKING DAY       GOOD 1.400000
## 5 1600 SPRING NO HOLIDAY               4    WORKING DAY       GOOD 2.666979
## 6 1606 SPRING NO HOLIDAY               5    WORKING DAY       GOOD 1.604356
##       hum windspeed
## 1 80.5833 10.749882
## 2 69.6087 16.652113
## 3 43.7273 16.636703
## 4 59.0435 10.739832
## 5 43.6957 12.522300
## 6 51.8261  6.000868
```

We summarize the `bike_to_interpreted` dataset.

```r
summary(bike_to_interpreted)
```

```
##       cnt           season          holiday    days_since_2011
##  Min.   :  22   SPRING:181   NO HOLIDAY:710   Min.   :  0.0
##  1st Qu.:3152   SUMMER:184   HOLIDAY   : 21   1st Qu.:182.5
##  Median :4548   FALL  :188                    Median :365.0
##  Mean   :4504   WINTER:178                    Mean   :365.0
##  3rd Qu.:5956                                 3rd Qu.:547.5
##  Max.   :8714                                 Max.   :730.0
##           workingday              weathersit       temp             hum
##  NO WORKING DAY:231   GOOD            :463   Min.   :-5.221   Min.   : 0.00
##  WORKING DAY   :500   MISTY           :247   1st Qu.: 7.843   1st Qu.:52.00
##                       RAIN/SNOW/STORM: 21   Median :15.422   Median :62.67
##                                              Mean   :15.283   Mean   :62.79
##                                              3rd Qu.:22.805   3rd Qu.:73.02
##                                              Max.   :32.498   Max.   :97.25
##    windspeed
##  Min.   : 1.500
##  1st Qu.: 9.042
##  Median :12.125
##  Mean   :12.763
```

```
##   3rd Qu.:15.625  
##   Max.   :34.000  
```

At last, we create the *interpretable linear model* that is **linear regression** as follows:

```
lm_bike <- lm(cnt ~ ., data=bike_to_interpreted)
```

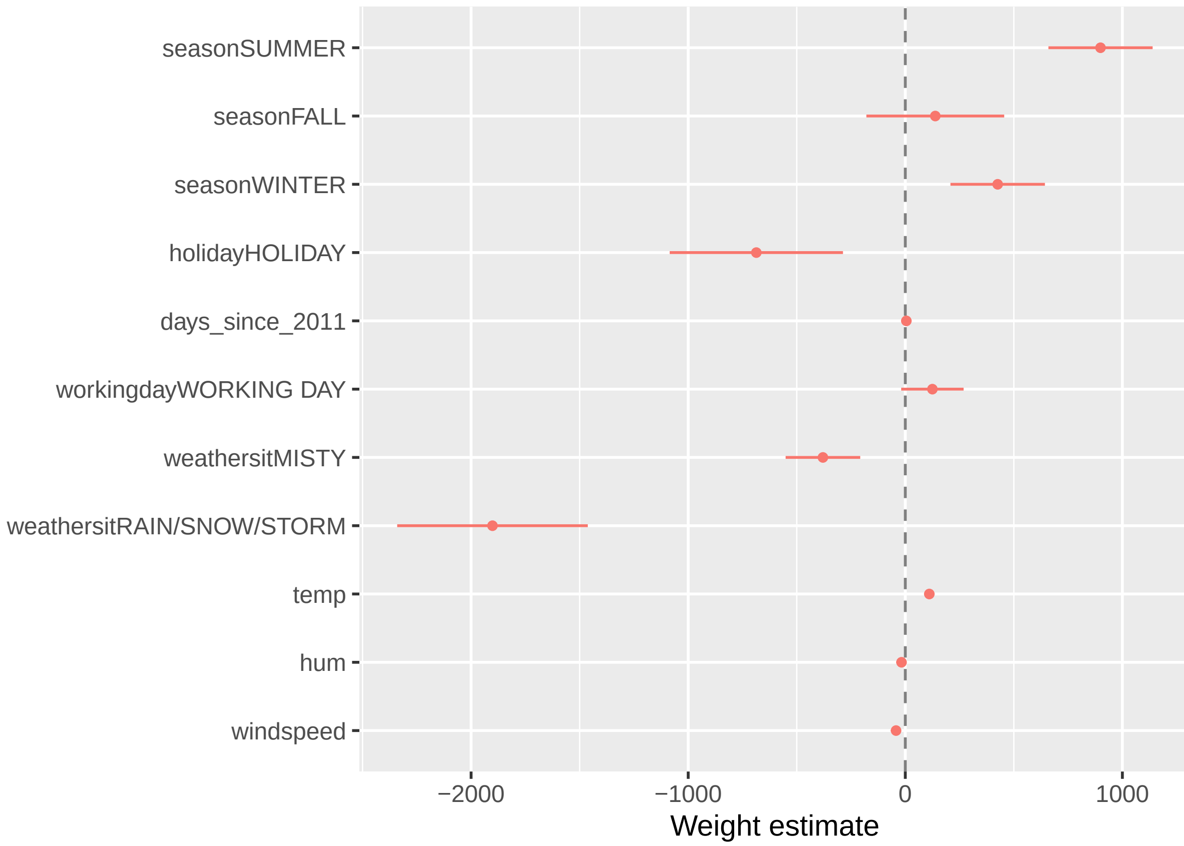
We show the details of the linear models as follows:

```
summary(lm_bike)
```

```
## 
## Call:
## lm(formula = cnt ~ ., data = bike_to_interpreted)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -3509.6  -397.9    78.7   534.1  3482.4 
## 
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)    
## (Intercept)               2399.4422   238.3066  10.069  < 2e-16 ***
## seasonSUMMER               899.3182   122.2833   7.354 5.24e-13 ***
## seasonFALL                 138.2154   161.7037   0.855 0.392977    
## seasonWINTER               425.6029   110.8199   3.840 0.000134 ***
## holidayHOLIDAY            -686.1154   203.3015  -3.375 0.000778 ***
## days_since_2011              4.9264     0.1728  28.507  < 2e-16 ***
## workingdayWORKING DAY      124.9209    73.2666   1.705 0.088623 .  
## weathersitMISTY           -379.3985    87.5532  -4.333 1.68e-05 ***
## weathersitRAIN/SNOW/STORM -1901.5399   223.6400  -8.503  < 2e-16 ***
## temp                       110.7096     7.0433  15.718  < 2e-16 ***
## hum                        -17.3772     3.1694  -5.483 5.80e-08 ***
## windspeed                  -42.5135     6.8917  -6.169 1.15e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 886.9 on 719 degrees of freedom
## Multiple R-squared:  0.7936, Adjusted R-squared:  0.7904 
## F-statistic: 251.2 on 11 and 719 DF,  p-value: < 2.2e-16
```

We probably need to install the `dotwhisker` and `dply` packages. Specifically, with `dotwhisker` we can view the weights of our linear model. Alternatively, we can call `coef-plot.R` function.

```
library(dotwhisker)
library(dplyr)
dwplot(lm_bike,
       vline = geom_vline(xintercept = 0, colour = "grey50", linetype = 2)) + xlab("Weight estimate")
```

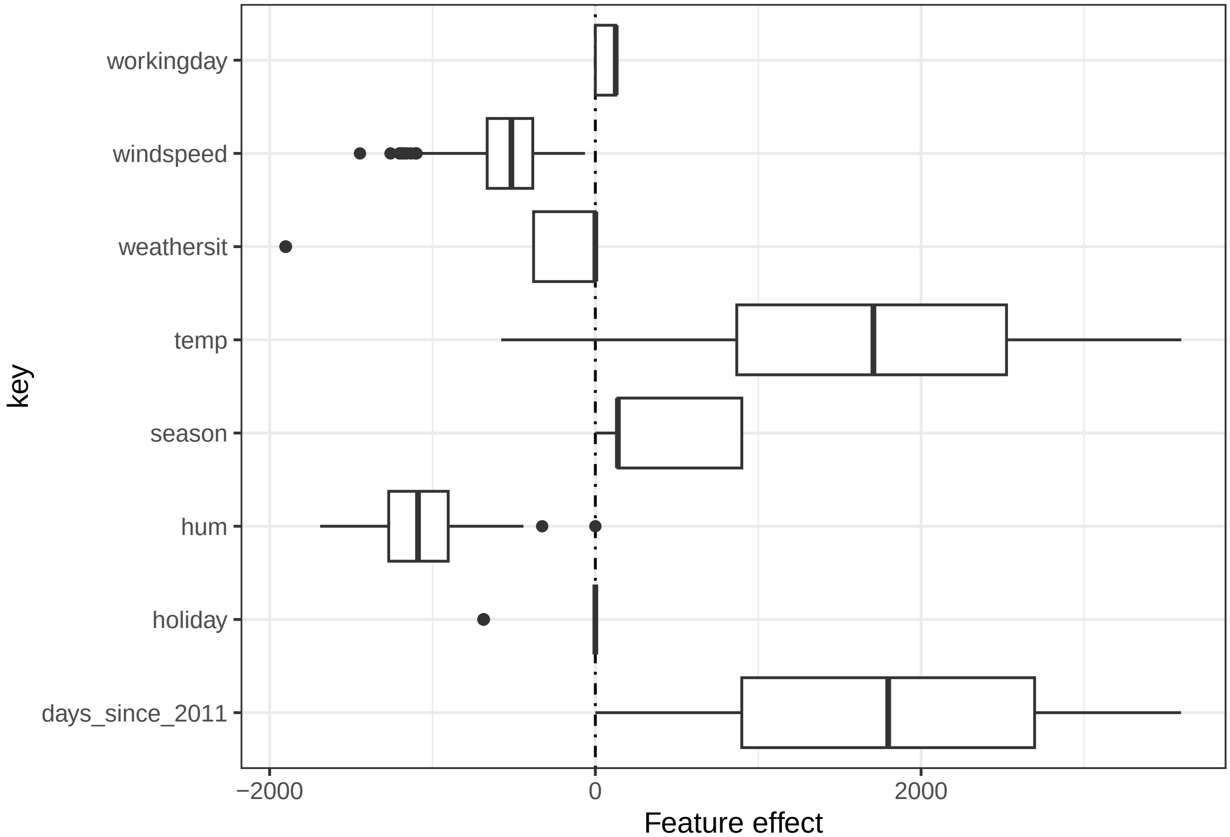


We can load the following R codes consisting several functions for showing the *interpretability of our linear regression* model.

```
source("utils.R", encoding = "UTF-8")
source("ggplot-theme.R", encoding = "UTF-8")
source("effect-plot.R", encoding = "UTF-8")
source("coef-plot.R", encoding = "UTF-8")
```

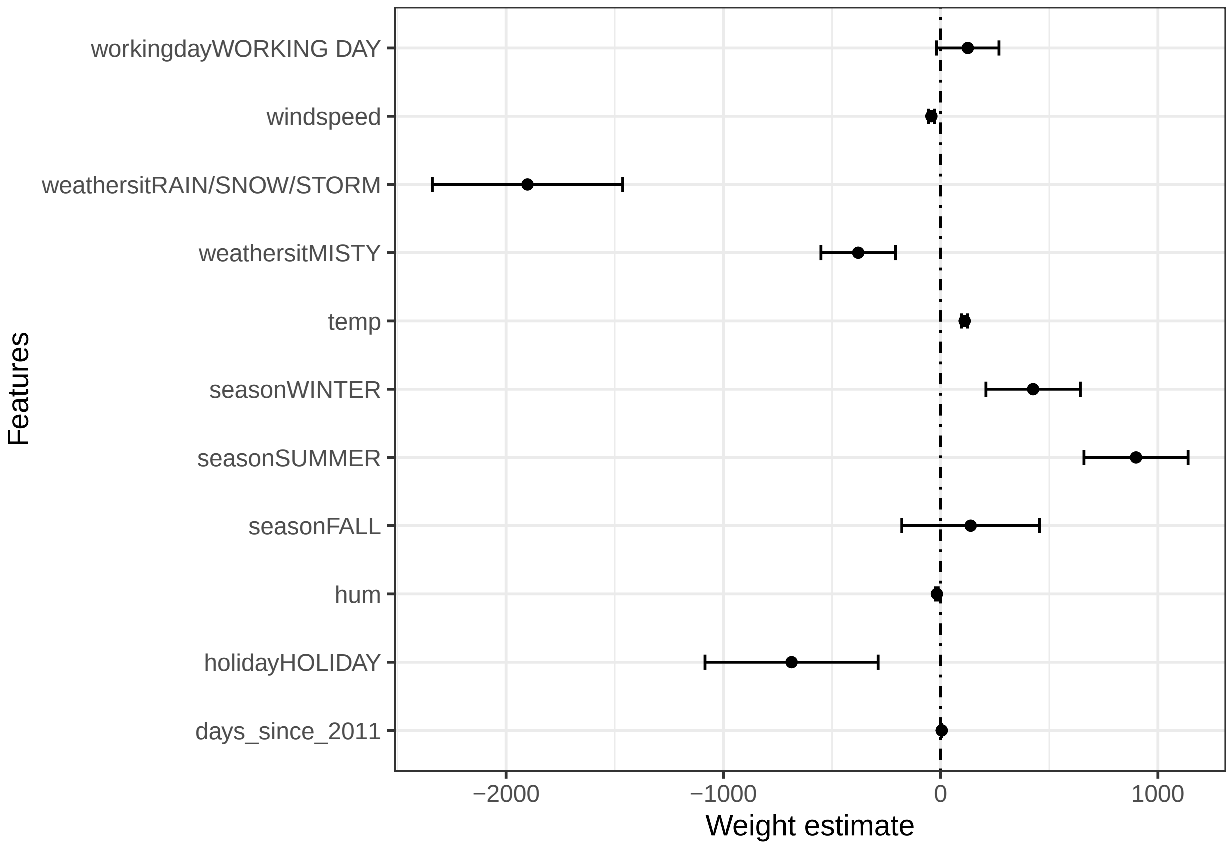
Let's display the **effect plot** of our model.

```
effect_plot(lm_bike, bike_to_interpreted)
```



Let's display the **coefficient plot** of our model.

```
coef_plot(lm_bike)
```



Let's load the `cervical` dataset (Molnar 2019).

```
load("cervical.RData")
```

Let us view the first five rows.

```
head(cervical)
```

```
##   Age Number.of.sexual.partners First.sexual.intercourse Num.of.pregnancies
## 1  18                         4                       15                  1
## 2  15                         1                       14                  1
## 3  34                         1                       15                  1
## 4  52                         5                       16                  4
## 5  46                         3                       21                  4
## 6  42                         3                       23                  2
##   Smokes Smokes..years. Hormonal.Contraceptives Hormonal.Contraceptives..years.
## 1      0              0                       0                               0
## 2      0              0                       0                               0
## 3      0              0                       0                               0
## 4      1             37                       1                               3
## 5      0              0                       1                              15
## 6      0              0                       0                               0
##   IUD IUD..years. STDs STDs..number. STDs..Number.of.diagnosis
## 1   0           0    0             0                         0
## 2   0           0    0             0                         0
## 3   0           0    0             0                         0
## 4   0           0    0             0                         0
## 5   0           0    0             0                         0
## 6   0           0    0             0                         0
```

```
##   STDs..Time.since.first.diagnosis STDs..Time.since.last.diagnosis  Biopsy
## 1                                1                               1 Healthy
## 2                                1                               1 Healthy
## 3                                1                               1 Healthy
## 4                                1                               1 Healthy
## 5                                1                               1 Healthy
## 6                                1                               1 Healthy
```

We convert the `Biopsy` column into binary values (`1 = Cancer` and `0 = Healthy`).

```r
cervical$Biopsy <- ifelse( cervical$Biopsy == "Healthy", 0, 1  )
```

```r
cervical_to_interpreted <- cervical[c("Hormonal.Contraceptives", "Smokes", "Num.of.pregnancies", "STDs.
                         "IUD", "Biopsy")]
```

Let us model the `cervical_to_interpreted` dataset by using logistic regression.

```r
lr_cervical <- glm(Biopsy ~ ., data=cervical_to_interpreted, family = binomial(link = "logit"))
```

Let us examine the model's summary.

```r
summary(lr_cervical)
```

```
##
## Call:
## glm(formula = Biopsy ~ ., family = binomial(link = "logit"),
##     data = cervical_to_interpreted)
##
## Coefficients:
##                           Estimate Std. Error z value Pr(>|z|)
## (Intercept)               -2.91015    0.32259  -9.021   <2e-16 ***
## Hormonal.Contraceptives   -0.11666    0.29896  -0.390   0.6964
## Smokes                     0.25578    0.37193   0.688   0.4916
## Num.of.pregnancies         0.03680    0.09653   0.381   0.7030
## STDs..Number.of.diagnosis  0.81549    0.32601   2.501   0.0124 *
## IUD                        0.61630    0.39959   1.542   0.1230
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 408.60  on 857  degrees of freedom
## Residual deviance: 399.47  on 852  degrees of freedom
## AIC: 411.47
##
## Number of Fisher Scoring iterations: 5
```

References

Molnar, Christoph. 2019. *Interpretable Machine Learning: A Guide for Making Black Box Models Explainable.*