Multiple Regression: Understanding Statistical Control and Presenting Results

Andrew Zieffler

Prepare

```
# Load the data (homework-achievement.csv)
> city = read.csv(“~/epsy-8251/riverside_final.csv")

# Load libraries; Note: you may need to install them first
> library(sm)
> library(ggplot2)
```

Fit the Multiple Regression Model

```
> lm.1 = lm(income ~ 1 + edu + senior, data = city)
> summary(lm.1)


Coefficients:
            Estimate Std. Error t value   Pr(>|t|)
(Intercept)   6769.2     5372.9   1.260    0.21776
edu           2251.8      334.6   6.729 0.00000022 ***
senior         738.8      210.1   3.516    0.00146 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 7646 on 29 degrees of freedom
Multiple R-squared:  0.7418,   Adjusted R-squared:  0.724
F-statistic: 41.65 on 2 and 29 DF,  p-value: 0.000000002977
```

Predictions

$$\hat{\text{Income}} = 6769 + 2252(\text{Education Level}) + 739(\text{Seniority})$$

Let's predict the average income for employees who have differing education levels,

Education level = 10 years
Education level = 11 years
Education level = 12 years

Let's assume that these employees all have 10 years of seniority.

edu	senior	Predicted income
10	10	6769 + 2252(10) + 739(10) = 36,679
11	10	6769 + 2252(11) + 739(10) = 38,931
12	10	6769 + 2252(12) + 739(10) = 41,183

$$\hat{\text{Income}} = 6769 + 2252(\text{Education Level}) + 739(\text{Seniority})$$

edu	senior	Predicted income	
10 } +1	10	6769 + 2252(10) + 739(10) = 36,679 } +2252	
11 } +1	10	6769 + 2252(11) + 739(10) = 38,931 } +2252	
12	10	6769 + 2252(12) + 739(10) = 41,183	

A one-year difference in education level is associated with a $2252 difference in income…**controlling for** differences in seniority by **holding that value constant**.

edu	senior	Predicted income	
10 } +1	11	6769 + 2252(10) + 739(11) = 37,418 } +2252	
11 } +1	11	6769 + 2252(11) + 739(11) = 39,670 } +2252	
12	11	6769 + 2252(12) + 739(11) = 41,922	

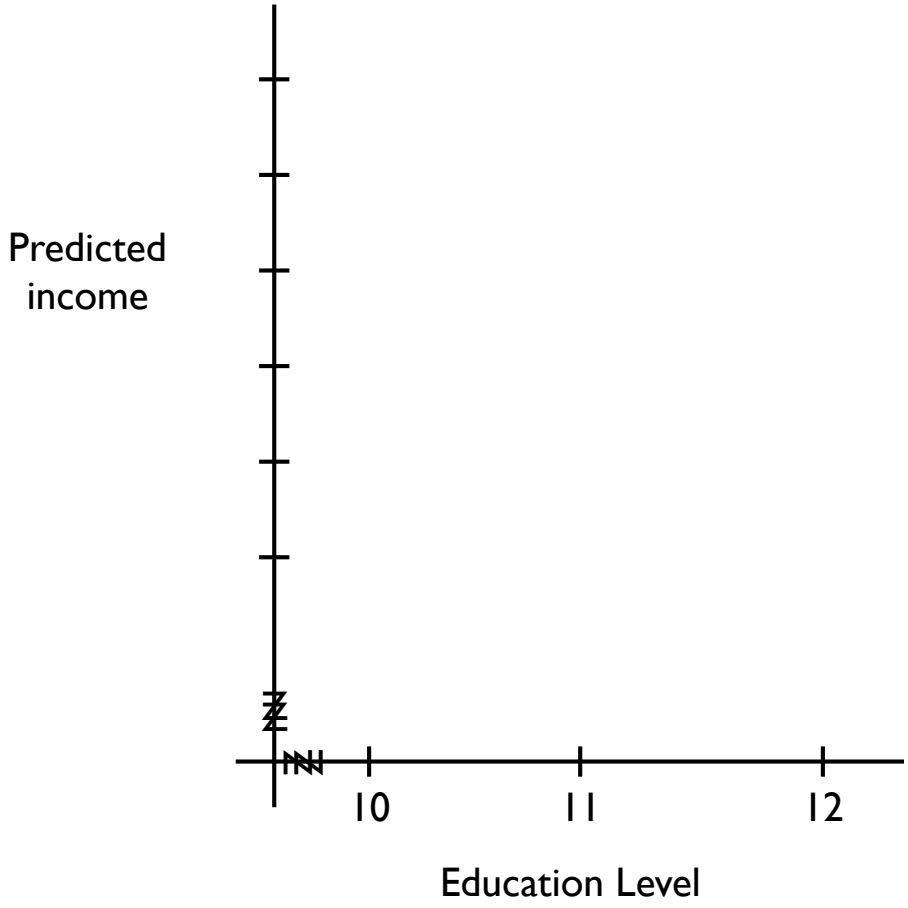
This will be true, regardless of which value we pick for `seniority`.

edu	senior	Predicted income
10	10	36,679
11	10	38,931
12	10	41,183

edu	senior	Predicted income
10	11	37,418
11	11	39,670
12	11	41,922

Sketch the scatterplot to display the ordered pairs (*edu, predicted income*) for those employess whose seniority value is 12.

Add the ordered pairs (*edu, predicted income*) for those employees whose seniority value is 5 to the plot, but use a different symbol.



$$\hat{\text{Income}} = 6769 + 2252(\text{Education Level}) + 739(\text{Seniority})$$

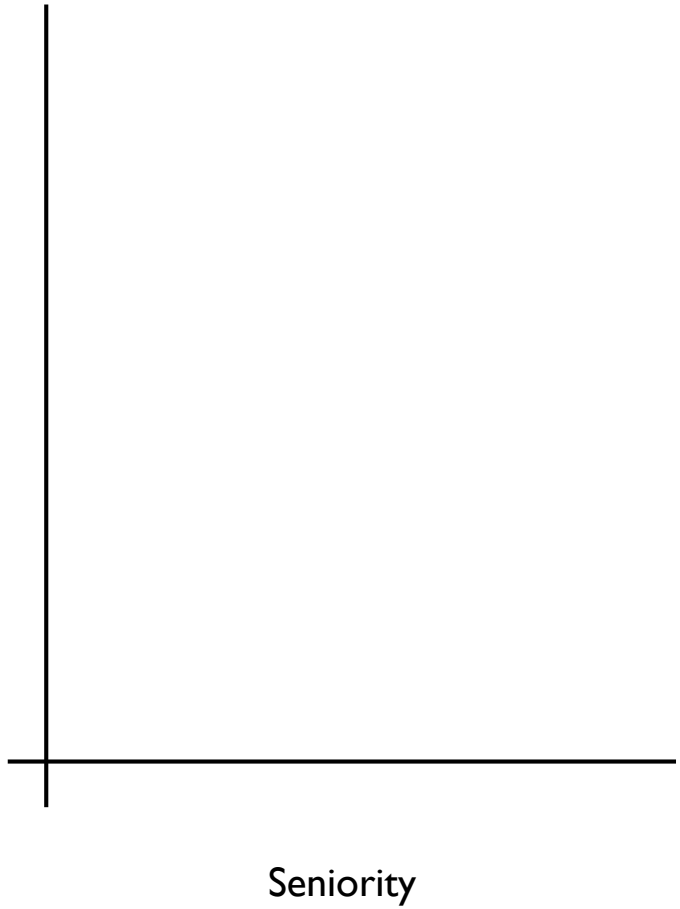
What happens if we pick a fixed education level and look at how the predicted income varies for different values of `seniority`?

`edu`	`senior`		Predicted `income`	
16	6	+1	6769 + 2252(16) + 739(6) = ?	?
16	7	+1	6769 + 2252(16) + 739(7) = ?	?
16	8		6769 + 2252(16) + 739(8) = ?	

Try it and convince yourself!

Sketch the predicted regression lines showing the effect of seniority on income for employees with 14 and 16 years of education.

Indicate what the slopes of the lines are and also indicate the vertical distance between the lines.



Using R to get Predictions

```
> myData = data.frame(
   edu =  c(10, 11, 12),
   senior =  c(10, 10, 10)
   )

> myData


  homework parentEd
1        1       12
2        2       12
3        3       12
```

We create a new data frame from which we are going to predict GPAs. The variables in this data frame need to have the exact same names as the predictors in your `lm()` model.

```
lm(income ~ edu + senior)
```

Use the `predict()` function to obtain predictions. This function takes the name of the fitted model and the argument `newdata=` which gives the name of the data frame from which we are predicting.

```
> predict(lm.1, newdata = myData)

       1        2        3
74.66235 75.65019 76.63804
```

Here we append the predictions to the original data frame to make it more readable.

```
> myPreds = predict(lm.a, newdata = myData)

> cbind(myData, myPreds)

  homework parentEd  myPreds
1        1       12 74.66235
2        2       12 75.65019
3        3       12 76.63804
```

Try using R to compute the predicted values for income for the following students.

edu	senior
16	6
16	7
16	8

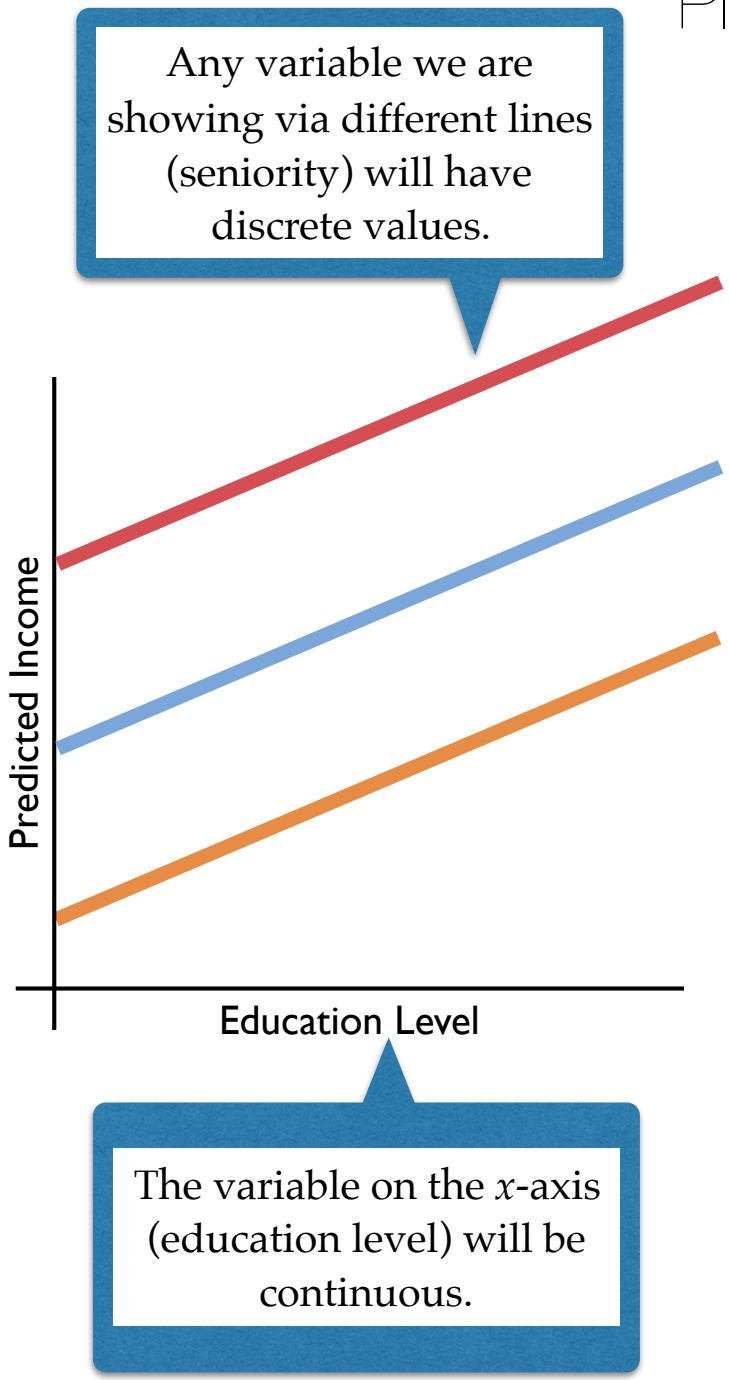
Considering a Plot of the Results

$$\hat{\text{Income}} = 6769 + 2252(\text{Education Level}) + 739(\text{Seniority})$$

With two (or more) effects in the model we have multiple displays of the fitted lines that are possible.

- Which predictor do you want to display on the x-axis?
- How many levels of the remaining predictors do you want to display?

Planning the Plot

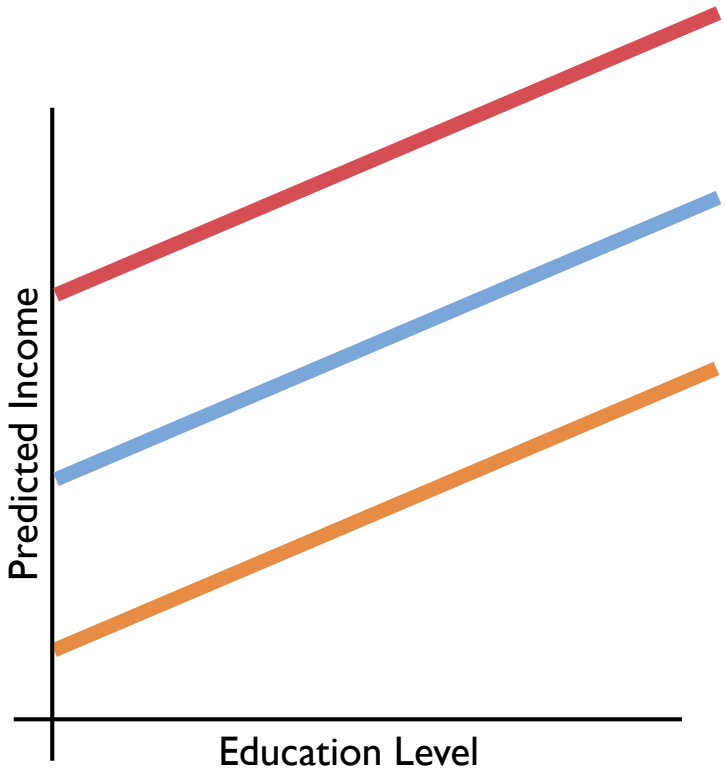


- Which predictor do you want to display on the *x*-axis?

  Education Level

- How many levels of the remaining predictors do you want to display?

  Let's show 3 different values of seniority




```
> summary(city)

      edu             senior
 Min.   : 8   Min.   : 1.00
 1st Qu.:12   1st Qu.: 9.75
 Median :16   Median :15.00
 Mean   :16   Mean   :14.81
 3rd Qu.:20   3rd Qu.:20.25
 Max.   :24   Max.   :27.00
```

Range of education levels is from 8–24 (variable on x-axis)

Values picked should be between 1 and 27, but should be interpretable.

- What **range of values** should we use for the predictor on the x-axis?

  8–24

- Which discrete values should we choose for the the remaining predictors?

  1 (little seniority), 10 (some seniority), 25 (a lot of seniority)

```
> plotData = expand.grid(
    edu =  seq(from = 1, to = 24, by = 1),
    senior =  c(1, 10, 25)
    )

> plotData
```

Range of education levels (variable on *x*-axis)

Discrete values of 1, 10, and 25.

	edu	senior
1	1	1
2	2	1
3	3	1
⋮	⋮	⋮
23	23	1
24	24	1
25	1	10
26	2	10
27	3	10
⋮	⋮	⋮
47	23	10
48	24	10
49	1	25
50	2	25
51	3	25
⋮	⋮	⋮
72	24	25

The `expand.grid()` function crosses all values of edu with all levels of senior. This sets up several pairs of values from which we can predict income.

```
> yhat =  predict(lm.1, newdata = plotData)

> plotData =  cbind(plotData, yhat)

> plotData
```

	edu	senior	yhat
1	1	1	9759.814
2	2	1	12011.660
3	3	1	14263.505
4	4	1	16515.351
5	5	1	18767.197
6	6	1	21019.042
⋮	⋮	⋮	⋮
67	19	25	68024.15
68	20	25	70276.00
69	21	25	72527.84
70	22	25	74779.69
71	23	25	77031.53
72	24	25	79283.38

Predict using the fitted model and the newly created data

Bind the data and the predictions together

After predicting, we can coerce any variable with discrete values (`seniority`) into a factor. This will help auto-create a legend when we plot it later on.

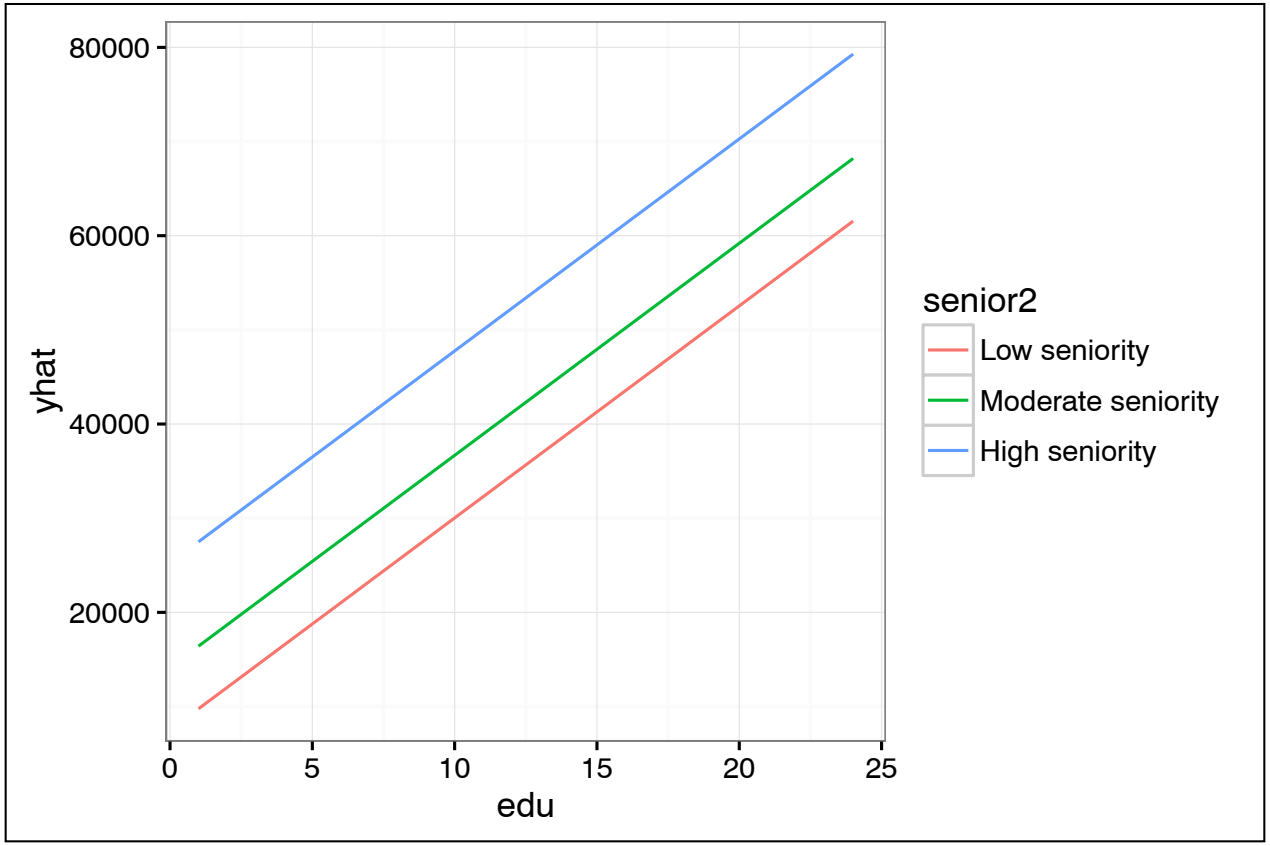
```
> plotData$senior2 =  factor(plotData$senior,
    levels = c(1, 10, 25),
    labels = c("Low seniority", "Moderate seniority", "High Seniority")
    )

> head(plotData)


  edu senior      yhat       senior2
1   1      1  9759.814 Low seniority
2   2      1 12011.660 Low seniority
3   3      1 14263.505 Low seniority
4   4      1 16515.351 Low seniority
5   5      1 18767.197 Low seniority
6   6      1 21019.042 Low seniority
```

Here we use the group= aesthetic to draw different lines for each value of senior2. We then use color= to color the lines different colors.

```
> ggplot(data = plotData, aes(x = edu, y = yhat, group = senior2)) +
    geom_line(aes(color = senior2)) +
    theme_bw()
```



```
> ggplot(data = plotData, aes(x = edu, y = yhat, group = senior2)) +
    geom_line(aes(color = senior2), lwd = 1.5) +
    theme_bw() +
    xlab("Education level") +
    ylab("Predicted income") +
    scale_color_brewer(name = "Seniority level", palette = "Set2")
```

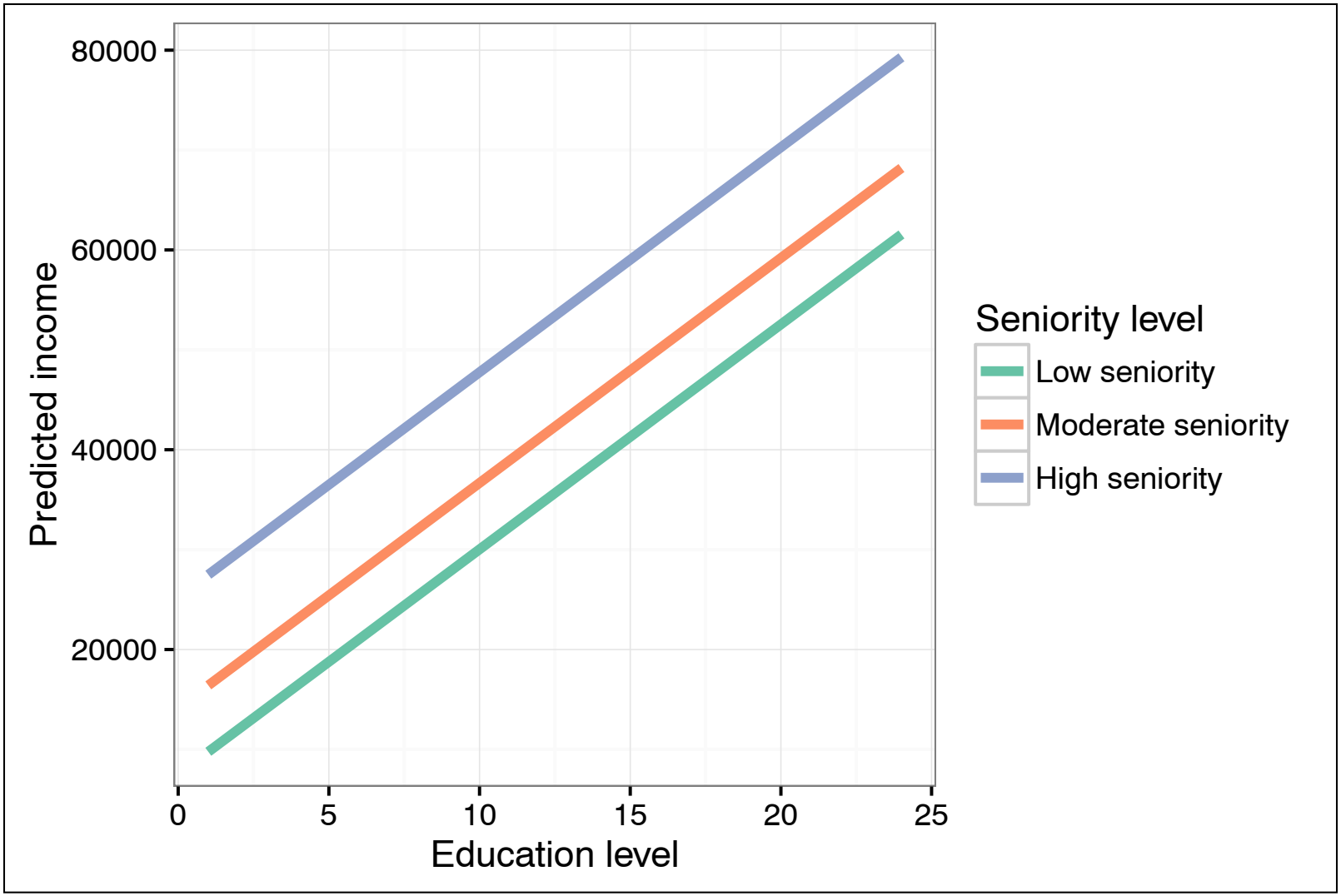


Figure 1. *Predicted income as a function of education level for employees with 1 (low), 10 (moderate), and 25 (high) years of seniority.*

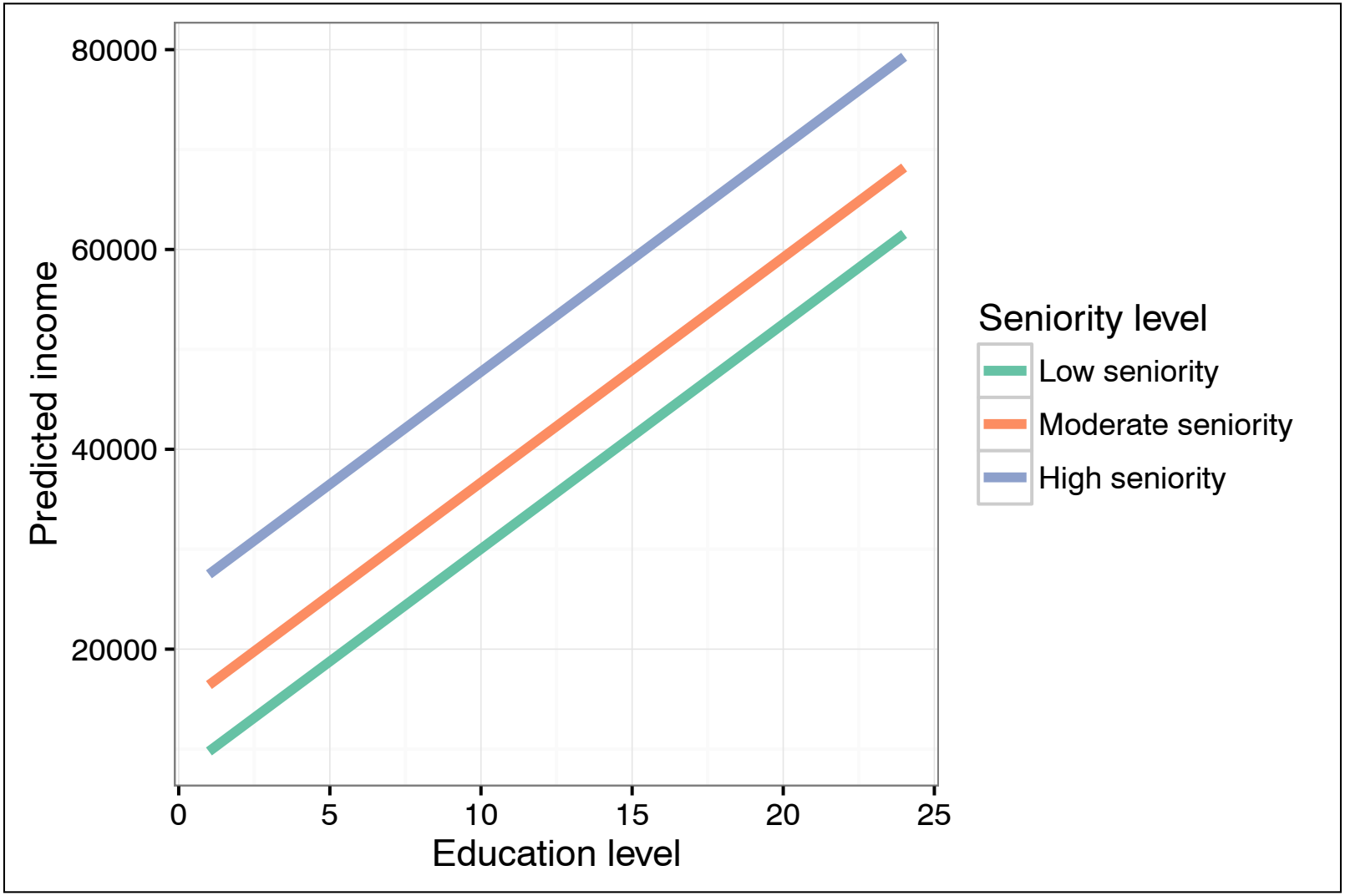


Figure 1. *Predicted income as a function of education level. Seniority is patrolled out of the model by fixing this variable to its mean value of 14.81.*