The Impact of Educational Expenses of New Coders*

The Factors Affecting Education Costs of Beginner Programmers to Learn to Code aside from University Tuition

Bongju Yoo

17 April 2021

1 Introduction

With an increasing interest in computer programming and the growth of coding related jobs, more and more people invest their time and money in taking programming courses (Hughes 2020). In this paper, I examine some of the factors that may affect educational expenses of new coders or entry-level programmers aside from university tuition, mainly focusing on the level of education, the size of city, commuting time, income, and learning time. The dataset used for this paper is obtained from freeCodeCamps GitHub page.[1] Approximately 20,000 beginner programmers had participated in the survey (freeCodeCamp 2017).

Firstly, I processed correlation analysis to identify the relationship between each independent variable and dependent variable (education costs) pair and found that all the pairs have a positive linear relationship except for commuting time. Secondly, I used multiple linear regression to determine the statistical significance of the factors. The result of the analysis shows that commuting time is not statistically significant. Interestingly, education background is also not statistically significant associated with educational expenses of beginner programmers. Other variables, such as the size of city, income, and learning time, have a statistical significance on the costs.

This analysis shows that beginner programmers living in small towns and/or low-income households are likely to spend less money on learning to code aside from university tuition than those who live in bigger cities and/or high-income households. It is assumed that there is still a lack of educational resources for new coders living in rural areas and/or low-income households. Free online learning platforms such as freeCodeCamp can be a great resource of education for the coders.

The paper is organized as follows. The Data section describes features of the original survey data and how the data is preprocessed. The Model section explains the multiple linear regression model used to assess the association between each factor and educational costs of beginner programmers, and evaluates the model using its residual standard error, multiple R-squared, and F-statistic. The Results section summarises the results of the regression model and model evaluation processes in the Model section. Lastly, the Discussion section discusses the findings and potential limitations of the paper, and suggests directions for future research related to this data.

2 Data

*https://github.com/bonjwow/new-coders

[1]https://github.com/freeCodeCamp/2017-new-coder-survey

Table 1: A partial view ofthe cleaned dataset

Gender	Age	City Population	Commute Time	Income	Months Programming	School Degree	Money For Learning
0	21	2	2	13000	5	0	1000
0	26	1	0	24000	5	1	0
1	29	1	3	40000	12	1	0
1	23	0	0	30000	29	1	700
0	20	0	1	20000	5	2	0
0	18	0	1	10000	3	0	0
0	32	0	3	20000	1	3	0
0	29	1	3	60000	12	3	200
1	46	2	0	76000	14	2	500
0	31	0	3	34000	28	1	500

2.1 Description of Dataset

The original dataset used for this paper is obtained from `freeCodeCamp`'s GitHub page. The format of the dataset is a comma-separated values (CSV) file, which contains 136 columns and 18,175 observations. I selected 8 variables and cleaned the data using the R programming language (R Core Team 2020) and the tidyverse package (Wickham et al. 2019) and the dplyr package (Wickham et al. 2021). The selected variables are: Age, CityPopulation, CommuteTime, Gender, Income, MoneyForLearning, MonthsProgramming, and SchoolDegree. The CityPopulation is the estimated number of city population of the recipient; the question asked was "About how many people live in your city?", and there are three options to choose: "less than 100,000", "between 100,000 and 1 million", and "more than 1 million". Since these answers were coded as strings, I converted them into numeric variables using the `recode` function of `dplyr` (Wickham et al. 2021). The answers for CommuteTime and SchoolDegree were also coded as strings in the original dataset, so I applied the same data cleaning process to the variables as I did with CityPopulation. Also, I omitted observations which have a missing value. Table 1 is a partial view of the cleaned dataset after the preprocessing.

2.2 Methodology and Data Collection

```
##
## ================================================================================
## Statistic            N      Mean     St. Dev.   Min  Pctl(25) Pctl(75)    Max
## --------------------------------------------------------------------------------
## Gender            7,022   0.174      0.379      0      0        0         1
## Age               7,022   29.774     7.684     13      25       33       71
## CityPopulation    7,022   1.221      0.777      0      1        2         2
## CommuteTime       7,022   2.199      1.451      0      1        3         5
## Income            7,022 42,966.890 59,162.290 6,000  17,000   55,000  1,000,000
## MonthsProgramming 7,022   23.986     46.496     0      3        26       744
## SchoolDegree      7,022   1.676      0.968      0      1        2         4
## MoneyForLearning  7,022 1,032.273  4,030.722    0      0       399     170,000
## --------------------------------------------------------------------------------
```

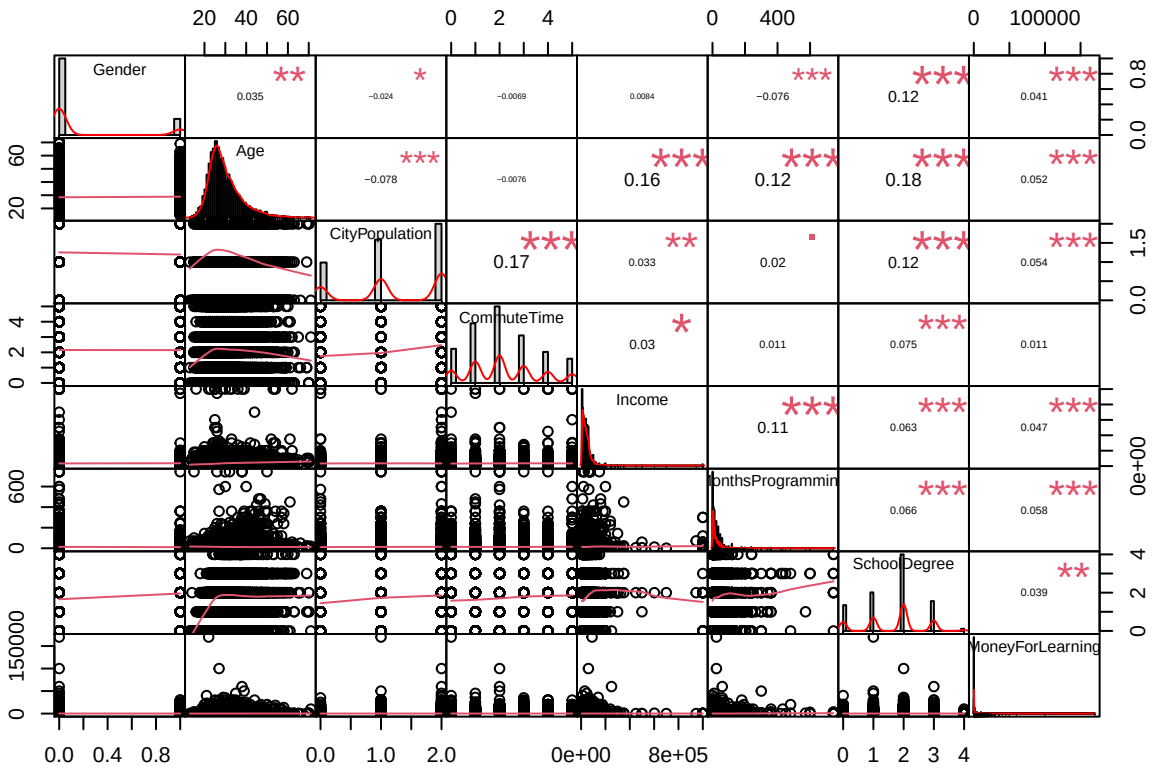
```
### Print correlation matrix
PerformanceAnalytics::chart.Correlation(dfNewCoders,
```

Table 2: Correlation matrix for variables

	1	2	3	4	5	6	7	8
1. Gender	-							
2. Age	0.03**	-						
3. CityPopulation	-0.02*	-0.08***	-					
4. CommuteTime	-0.01	-0.01	0.17***	-				
5. Income	0.01	0.16***	0.03**	0.03*	-			
6. MonthsProgramming	-0.08***	0.12***	0.02	0.01	0.11***	-		
7. SchoolDegree	0.12***	0.18***	0.12***	0.07***	0.06***	0.07***	-	
8. MoneyForLearning	0.04***	0.05***	0.05***	0.01	0.05***	0.06***	0.04**	-

* p < 0.05, ** p < 0.01, *** p < 0.001

```
                                    histogram = TRUE,
                                    pch = 19)
```



3 Model

```
#### Multiple regression analysis ####
multiReg1 <- lm(formula =
```

```
                MoneyForLearning ~
                CityPopulation +
                CommuteTime +
                Income +
                MonthsProgramming +
                SchoolDegree,
                data = dfNewCoders)

multiReg2 <- lm(formula =
                MoneyForLearning ~
                Age +
                CityPopulation +
                CommuteTime +
                Income +
                MonthsProgramming +
                SchoolDegree,
                data = dfNewCoders)

multiReg3 <- lm(formula =
                MoneyForLearning ~
                Gender +
                Age +
                CityPopulation +
                CommuteTime +
                Income +
                MonthsProgramming +
                SchoolDegree,
                data = dfNewCoders)


# summary(multiReg3)
stargazer::stargazer(multiReg1, multiReg2, multiReg3,
          title = "Result of regression model",
          header = FALSE,
          type = "latex")
```

```
#### Beta ####
QuantPsyc::lm.beta(multiReg3)
```

```
##            Gender               Age    CityPopulation       CommuteTime
##      0.0424994985      0.0407495701      0.0539486694      0.0001163827
##            Income MonthsProgramming      SchoolDegree
##      0.0318718375      0.0510716380      0.0150135277
```

```
#### Check multicollinearity with Variance Inflation Factor (VIF) & Tolerance ####

### VIF
car::vif(multiReg3)
```

```
##            Gender               Age    CityPopulation       CommuteTime
##          1.022726          1.080787          1.055941          1.034138
##            Income MonthsProgramming      SchoolDegree
##          1.036287          1.033784          1.073855
```

Table 3: Result of regression model

	Dependent variable:		
	MoneyForLearning		
	(1)	(2)	(3)
Gender			452.239***
			(127.733)
Age		21.818***	21.377***
		(6.477)	(6.473)
CityPopulation	249.902***	272.396***	279.752***
	(62.959)	(63.266)	(63.248)
CommuteTime	−1.737	−0.782	0.323
	(33.574)	(33.550)	(33.524)
Income	0.003***	0.002***	0.002***
	(0.001)	(0.001)	(0.001)
MonthsProgramming	4.469***	4.107***	4.427***
	(1.038)	(1.043)	(1.046)
SchoolDegree	114.923**	84.034*	62.536
	(50.119)	(50.915)	(51.234)
Constant	320.017**	−281.955	−329.379
	(132.956)	(222.693)	(222.913)
Observations	7,022	7,022	7,022
R^2	0.008	0.010	0.012
Adjusted R^2	0.008	0.009	0.011
Residual Std. Error	4,015.106 (df = 7016)	4,012.149 (df = 7015)	4,008.854 (df = 7014)
F Statistic	11.944*** (df = 5; 7016)	11.859*** (df = 6; 7015)	11.972*** (df = 7; 7014)
Note:			*p<0.1; **p<0.05; ***p<0.01

```r
### Tolerance
1/car::vif(multiReg3)
```

```
##            Gender               Age    CityPopulation       CommuteTime
##         0.9777785         0.9252514         0.9470228         0.9669891
##            Income MonthsProgramming      SchoolDegree
##         0.9649839         0.9673203         0.9312248
```

```r
# TODO: RMSE
```

4 Results

5 Discussion

Appendix

References

freeCodeCamp. 2017. *2017 New Coder Survey.* https://github.com/freeCodeCamp/2017-new-coder-survey.

Hughes, Owen. 2020. *Developer Training Sees Spike in Demand as More People Learn to Code.* https://www.techrepublic.com/article/the-economic-outlook-is-uncertain-so-more-people-want-to-become-developers/.

R Core Team. 2020. *R: A Language and Environment for Statistical Computing.* Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Wickham, Hadley, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Grolemund, et al. 2019. "Welcome to the tidyverse." *Journal of Open Source Software* 4 (43): 1686. https://doi.org/10.21105/joss.01686.

Wickham, Hadley, Romain François, Lionel Henry, and Kirill Müller. 2021. *Dplyr: A Grammar of Data Manipulation.* https://CRAN.R-project.org/package=dplyr.