R Exercise 1: INTA 6450 - Data Analytics and Security

2024-10-15

R Exercise 1

Prompt

1. Download `exercise_1.r`
2. Open the file in RStudio and run the commands in the exercise and look at the output.
3. Read the comments to try to understand how each command is working, and look at the output. Make some changes to the code to explore how the output differs.
4. Then, submit the output for running these commands for this assignment. You may copy/paste output from your terminal into a word or text document. You can convert the notebook to PDF submission:File->Compile Report->Report output format as HTML/PDF.
5. Write and submit 100-150 words describing the extension you made to the code. You may post the change description as a comment at the beginning of your code, or submit the write-up in a separate document. Make sure to clearly mark where in your code the changes were made either by commenting, highlighting, or noting the command lines where the changes occur. **NOTE:** Exported code does not retain line numbers. Please refrain from phrases such as "code change begins on line 62".

Submissions

You will submit the following:

1. Original code output (R file code and output)
2. Your code change and output from that code change (comment where your code changes start and end)
3. Your code change description (this should be between 100 to 150 words)

An example of what the change description should look like:

```
############################
# CHANGE DESCRIPTION
# 100-150 words on the changes I've made, including a reference to which
# initial command I've changed >
# END OF CHANGE DESCRIPTION
############################
```

Files must be in one of the following formats: pdf, doc, html

Please note some of these exercises have graphs/charts - be sure when you export the original output to also check that the file includes graphs/charts that may have been present. If it does not, you will need to re-export or manually include the graphs in your final pdf, doc or html file

Solution Summary

The solution was created by evaluating a linear regression model that included all predictors. It can be concluded that not all predictors are necessary, but only including `wage` and `educ` lacks a full definition of predictors that are significant. A plot is created to evaluate the relationship for each predictor, which requires a function to create each plot and then map them to form a full single image of all predictors.

This is important because the demonstration that there will be several important predictors. Predictors are important, yet all predictors should not be included. The data could be fitted better by adding in k-fold cross-validation. We could also create a model with only the significant predictors.

An overview of using the data and creating the figures are discussed further below. It is important to note that k-fold cross-validation would allow for determining if the original data high accuracy or R-squared values are due to overfitting.

Original code is attached in submission for reference or to run, as it is fully functional.

Wages

To observe the relationship between `wage` and each of the 17 predictors, a matrix scatter plots is developed with `ggplot` and `gridExtra`. A function is created to create a single scatter plot so that mapping can be utilized to create all 17 plots at once and be placed in a matrix for a single image. We could:

- Change `lm` model to 17 predictors
- Change `lm` model to significant number of predictors
- Add cross-validation using k-fold cross-validation

Wages Overview

The data obtained from `wage2.csv` contains information on monthly earnings, education, several demographic variables, and IQ scores for 935 men in 1980 [1].

[2]

- `wage`
- `hours`
- `IQ`
- `KWW`
- `educ`
- `exper`
- `tenure`
- `age`
- `married`
- `black`
- `south`
- `urban`
- `sibs`
- `brthord`
- `meduc`
- `feduc`
- `lwage`

Load and Inspect the Data

```
# Import libraries
# lm and glm are in the stats package loaded by default
library(ggplot2) # Plot functions
library(reshape)

# Wages ------------------------------------------------------------------------
# Load wages data from AWS .csv
wages <- read.csv('http://inta.gatech.s3.amazonaws.com/wage2.csv')
# Write the wages data to a .csv in local directory
# write.csv(wages, "exercises/M5_regression/data/wage2.csv", row.names = FALSE)

# Summary of wages statistics from data frame
summary(wages)
```

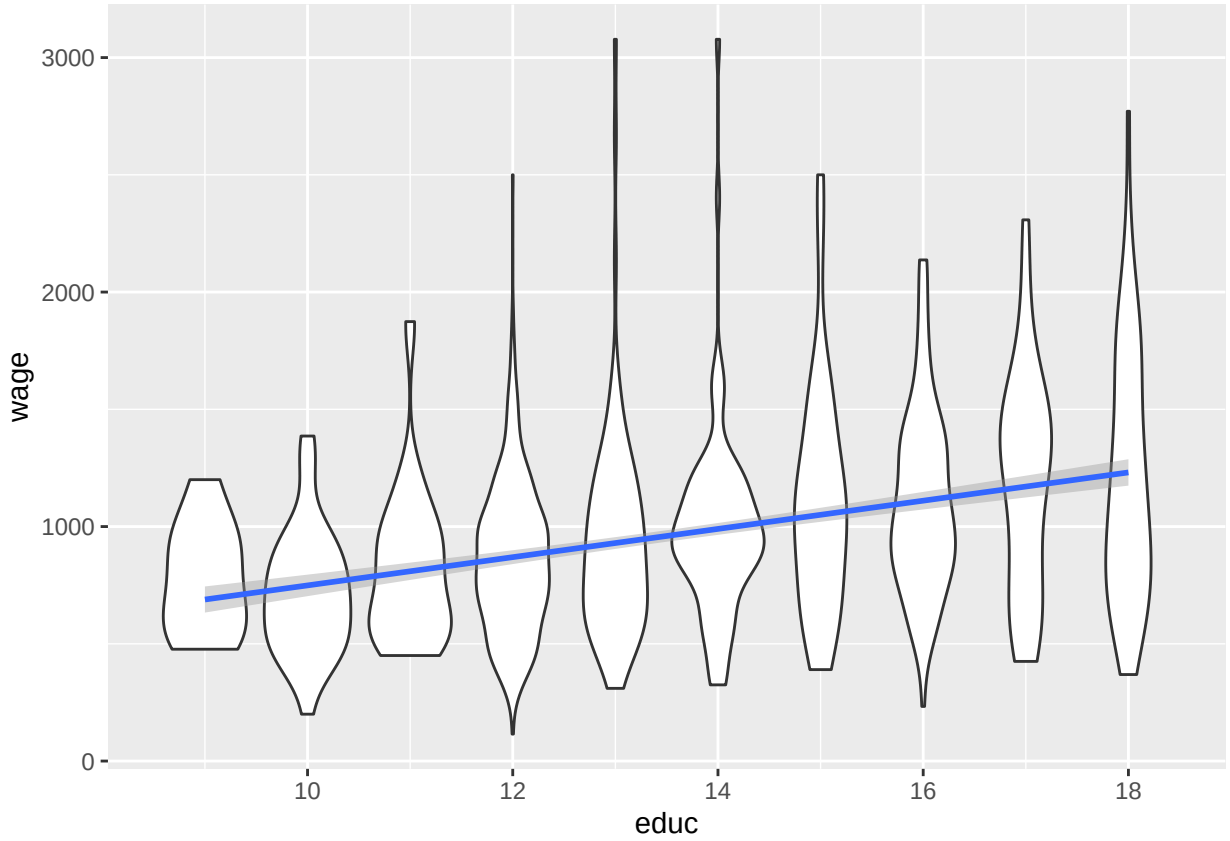
```
##       wage             hours             IQ             KWW
##  Min.   : 115.0   Min.   :20.00   Min.   : 50.0   Min.   :12.00
##  1st Qu.: 669.0   1st Qu.:40.00   1st Qu.: 92.0   1st Qu.:31.00
##  Median : 905.0   Median :40.00   Median :102.0   Median :37.00
##  Mean   : 957.9   Mean   :43.93   Mean   :101.3   Mean   :35.74
##  3rd Qu.:1160.0   3rd Qu.:48.00   3rd Qu.:112.0   3rd Qu.:41.00
##  Max.   :3078.0   Max.   :80.00   Max.   :145.0   Max.   :56.00
##
##       educ           exper           tenure            age
##  Min.   : 9.00   Min.   : 1.00   Min.   : 0.000   Min.   :28.00
##  1st Qu.:12.00   1st Qu.: 8.00   1st Qu.: 3.000   1st Qu.:30.00
##  Median :12.00   Median :11.00   Median : 7.000   Median :33.00
##  Mean   :13.47   Mean   :11.56   Mean   : 7.234   Mean   :33.08
##  3rd Qu.:16.00   3rd Qu.:15.00   3rd Qu.:11.000   3rd Qu.:36.00
##  Max.   :18.00   Max.   :23.00   Max.   :22.000   Max.   :38.00
##
##     married          black            south            urban
##  Min.   :0.000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.:1.000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000
##  Median :1.000   Median :0.0000   Median :0.0000   Median :1.0000
##  Mean   :0.893   Mean   :0.1283   Mean   :0.3412   Mean   :0.7176
##  3rd Qu.:1.000   3rd Qu.:0.0000   3rd Qu.:1.0000   3rd Qu.:1.0000
##  Max.   :1.000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
##
##       sibs           brthord           meduc           feduc
##  Min.   : 0.000   Min.   : 1.000   Min.   : 0.00   Min.   : 0.00
##  1st Qu.: 1.000   1st Qu.: 1.000   1st Qu.: 8.00   1st Qu.: 8.00
##  Median : 2.000   Median : 2.000   Median :12.00   Median :10.00
##  Mean   : 2.941   Mean   : 2.277   Mean   :10.68   Mean   :10.22
##  3rd Qu.: 4.000   3rd Qu.: 3.000   3rd Qu.:12.00   3rd Qu.:12.00
##  Max.   :14.000   Max.   :10.000   Max.   :18.00   Max.   :18.00
##                   NA's   :83       NA's   :78      NA's   :194
##      lwage
##  Min.   :4.745
##  1st Qu.:6.506
##  Median :6.808
##  Mean   :6.779
```

```
##  3rd Qu.:7.056
##  Max.   :8.032
##
```

Original Plot Data

The original plot data evaluated the wage and `educ` or education with a violin plot from:

```
# A picture of how the linear model does, with marginal distributions
# Save the picture by uncommenting the line below:
# ggsave('violin.png', width=7, height=5, units = "in")
plot_lm <- ggplot(data = wages, aes(x = educ, y = wage)) +
  geom_violin(aes(group = educ)) +
  stat_smooth(data = wages,
              aes(x = educ, y = wage),
              method = 'lm')
plot_lm
```



Form Linear Regression Model with All Predictors

```
# Predict observed crime rate
# wage~. separates wage (response variable) from predictors
model <- lm(wage~., data=wages)
summary(model)
```

Observing the Relationship between Wage and Predictors

To observe the relationship between `wage` and each of the 17 predictors, a matrix scatter plots is developed with `ggplot` and `gridExtra`. Remember, `ggplot2` was imported earlier, so two additional libraries need to be loaded. The library `gridExtra` is to arrange a grid, and `purrr` is used to map the plots. First, a list of the predictor variables is defined as `predictors`. The list of predictors is then utilized to generate scatter plots. A function is created to create a single scatter plot so that mapping can be utilized to create all 17 plots at once and be placed in a matrix for a single image.

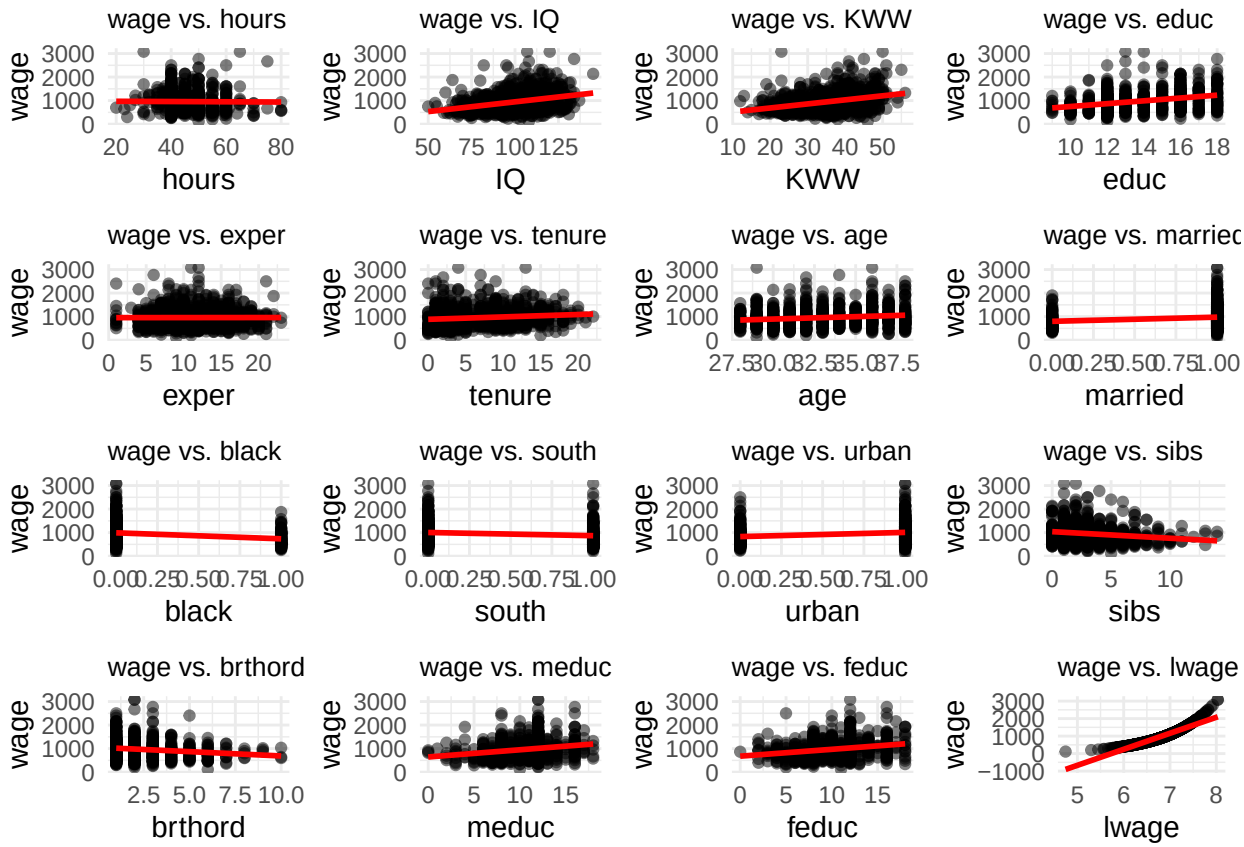
```
library(gridExtra)
library(purrr)

# Create a list of predictor variables
predictors <- c("hours", "IQ", "KWW", "educ", "exper", "tenure", "age",
                "married", "black", "south", "urban", "sibs", "brthord",
                "meduc", "feduc", "lwage")

# Function to create a scatter plot with regression line for each predictor
plot_predictor <- function(predictor) {
  ggplot(wages, aes_string(x = predictor, y = "wage")) +
    geom_point(alpha = 0.5) +
    geom_smooth(method = "lm", se = FALSE, color = "red") +
    theme_minimal() +
    labs(title = paste("wage vs.", predictor))
}

# Create plots for all predictors
plots <- map(predictors, plot_predictor)

# Arrange plots in a grid
scatter <- grid.arrange(grobs = plots, ncol = 4)
scatter
```



Original Working Code

```
library(ggplot2)
library(reshape)
# If you get an error on this command, you probably need to install ggplot2
# and/or reshape, with:
# > install.packages('ggplot2')
# > install.packages('reshape')
# Then, run the library command again
#"Wage2" dataset can be found more information https://cran.r-project.org/web/packages/wooldridge/woold
#"mroz" datase can be found here https://cran.r-project.org/web/packages/MASS/MASS.pdf


wages<-read.csv('http://inta.gatech.s3.amazonaws.com/wage2.csv')
# Read data from gatech website. This data accompanies the econometrics textbook by Wooldridge:
# http://www.amazon.com/Introductory-Econometrics-A-Modern-Approach/dp/1111531048


summary(wages)
# Print summary statistics for the data frame

table(wages$educ)
# Tabulate education levels in the sample
```

```r
ggplot(data=wages, aes(x=educ, y=wage)) + geom_point() + stat_smooth(formula=y~x)
# Look at the distribution of education and wages in a scatter plot

model.results <- lm(wage ~ educ, data=wages)
# Fit a linear model where y is wage and x is education

print(model.results)
# print a simple version of the model results

summary(model.results)
# Print a more detailed version of model results

data.to.predict <-data.frame(educ=c(1,12,14,50))
data.to.predict$predicted.wage <- predict(model.results, data.to.predict)
# Predict outcomes for different people with different levels of education, 1,
# 12, 14, and 50 years
data.to.predict

# To control for more variables, add them to the Right hand side of the
# 'formula', which is the 'wage ~ educ' piece of the code
model.results.detail <- lm(wage ~ educ + IQ, data=wages)
summary(model.results.detail)
# Note that adding IQ here reduces the coefficient on education, which makes
# sense per the discussion of omitted variables that we have done
# Try using different combinations of variables to see what works and makes sense.

ggplot(data=wages, aes(x=educ, y=wage)) + geom_violin(aes(group=educ)) + stat_smooth(data=wages, aes(x=
# A picture of how the linear model does, with marginal distributions
# Save the picture by uncommenting the line below:
# ggsave('violin.png', width=7, height=5, units = "in")

lfp <- read.csv('http://inta.gatech.s3.amazonaws.com/mroz.csv')
# Load labor force participation data
summary(lfp)

linear.model <- lm(inlf ~ nwifeinc + educ + exper + expersq + age + kidslt6 + kidsge6, data = lfp)
summary(linear.model)
# Fit a linear model

logit.model <- glm(inlf ~ nwifeinc + educ + exper + expersq + age + kidslt6 + kidsge6, data = lfp, famil
# Fit a logistic model using the 'glm' command
summary(logit.model)

probit.model <-glm(inlf ~ nwifeinc + educ + exper + expersq + age + kidslt6 + kidsge6, data = lfp, famil
# Fit a probit model, also using the 'glm' command.
# Note how the family command changes us from logit to probit

##############
#Prepare to plot predicitons for education from 0 to 20. Don't worry about
# understanding this code
predicted <- data.frame(educ=seq(0,20))
predicted$nwifeinc <- 17.7
predicted$exper<-9
```

```
predicted$expersq<-81
predicted$age<-42
predicted$kidslt6<-.238
predicted$kidsge6<-1.35
predicted$Logit<-predict(logit.model, newdata=predicted, type="response")
predicted$Probit<-predict(probit.model, newdata=predicted, type="response")
predicted$Linear<-predict(linear.model, newdata=predicted)

subdata <- predicted[,c("educ", "Linear", "Probit","Logit")]
msd<-melt(subdata, id="educ")
ggplot(msd) + geom_line(aes(x=educ, y=value, colour=variable)) +
  scale_colour_manual(values=c("red","green","blue"), name="") +
  ggtitle("Binary Response") +
  theme(plot.title = element_text(lineheight=8, face="bold", size=26)) +
  theme(legend.text = element_text(size=18)) +
  theme(axis.title = element_text(size=18)) +
  theme(legend.title = element_text()) +
  labs(x="Education", y="Probability Woman Is In Labor Force")
#ggsave('binary_response.png', width=7, height=5, units = "in")
# Complete plotting logit and probit comparison
#################

# Stepwise regression
library(MASS)
start.model<-lm(wage ~ hours + IQ + KWW + educ + exper + tenure + age + married + black + south + urban
# Give an initial model, which will be the most coefficients we'd want to ever use
summary(start.model)
stepwise.model<- step(start.model)
# The command "step" adds and subtracts coefficients to maximize a measure of
# goodness of fit
summary(stepwise.model)

stepwise.model.interactions <- step(start.model, scope=wage~.^2)
# The command "step" adds and subtracts coefficients to maximize a measure of
# goodness of fit.
summary(stepwise.model.interactions)
```

References

[1] M. Blackburn and D. Neumark, "Unobserved ability, efficiency wages, and interindustry wage differentials," *The Quarterly Journal of Economics*, vol. 107, no. 4, pp. 1421–1436, 1992.

[2] J. M. Woolridge, *Introductory econometrics: A modern approach*. Cengage Learning, 2019.