Lesson 1 Practice Hands-On

Directions

This Hands-On will **not** be graded, but we encourage you to complete it. However, the best way to become great data scientist is to practice.

It is a well known phenomena that most of us shrink throughout the day each day. The effects of gravity cause that our height measured at the end of the day is less than our height measured at the beginning of the day. Fortunately, at night, our bodies stretch out again, so that from one morning to the next, each of us has returned to the morning height from the day before.

In the dataset below, there are AM and PM height measurements (in mm) for students from a boarding school in India.

Hands on Part 1

Take the **following dataset**, and complete simple linear regression in R. Make sure to test, note, and correct for all assumptions if possible!

Hands on Part 2

Take the above dataset, and complete simple linear regression in Python. Make sure to test, note, and correct for all assumptions if possible!

Create a slide presentation that walks through your assumptions and overall findings in R and Python. Try to explain it in laymen's terms. Please also submit your code files for grading!

Caution!

Be sure to zip and submit your entire directory when finished!

Lesson 1 Practice Hands-On Solution

Below you will find the solutions for Parts I & II of the Lesson 1 Practice Hands-On.

Part I

```
#DSO106 Modeling L1 Practice Hands-on Solution

#Load in Libraries
library("car")
library("caret")
library("gvlma")
library("predictmeans")

#Test Assumptions

## Linearity

scatter.smooth(x=heights$AM_Height, y=heights$PM_Height,
main="Morning by Evening Height")
```

```
### Since this looks linear, the assumption of linearity has
been met!

## Homoscedasticity

### Run the Basic Model

lmModHeights = lm(PM_Height~AM_Height, data=heights)

### Graph it

par(mfrow=c(2,2))
plot(lmModHeights)

#### Looking at the graphs, there should be an approximately
flat line, and it looks like the top left curves and the bottom
left graph has a dip, so the assumption of homoscedasticity may
not be met.

### Breusch-Pagan Test

lmtest::bptest(lmModHeights)

#### Since this test was not significant, there is
homoscedasticity! The assumption is met!

### Non-Constant Variance Test

car::ncvTest(lmModHeights)

#### Same here - it wasn't significant, so the assumption has
been met!

## Homogeneity of Variance

### Looking at the graphs from the last assumption, this may not
have been passed. But continuing for learning purposes!

### GVLMA test
```

```
gvlma(lmModHeights)

#### All assumptions acceptable! Wow!

## Screening for outliers in x space

###Cook's D

CookD(lmModHeights, group=NULL, plot=TRUE, idn=3, newwd=TRUE)

#### Looks like observations 3, 4, and 12 may be a problem

### Leverage values

lev = hat(model.matrix(lmModHeights))
plot(lev)

heights[lev>.2,]

#### Going by leverage values, only 3 is really an issue

## Screening for outliers in y space

car::outlierTest(lmModHeights)

### This test was significant, so it's likely there is at least
one outlier

## Screening for outliers in x and y space (influential points)

summary(influence.measures(lmModHeights))

### Looks like the values on the list are 3, 11, and 37.  Should
probably try a model in which outliers are removed from the data

## Creating a new model without outliers to test against the
model with outliers

heightsNoO <-
heights[c(1,2,5,6,7,8,9,10,13,14,15,16,17,18,19,20,21,22,23,24,2
5,26,27,28,29,30,31,32,33,34,35,36,38,39,40,41),]
```

```
lmModHeightsNoO = lm(PM_Height~AM_Height, data=heightsNoO)

## Look at the model summaries for each

summary(lmModHeights)

### Looks like morning height is a significant predictor of
evening height and explains 99% of the variance in evening
height.

summary(lmModHeightsNoO)

### Very similar results with the model with no outliers, so
it's fine to keep and use the original model with all the data!
```

Part II

HERE is a Jupyter Notebook file containing the solution in Python. You will need to download it, save it, and then open it with your own Jupyter Notebook.