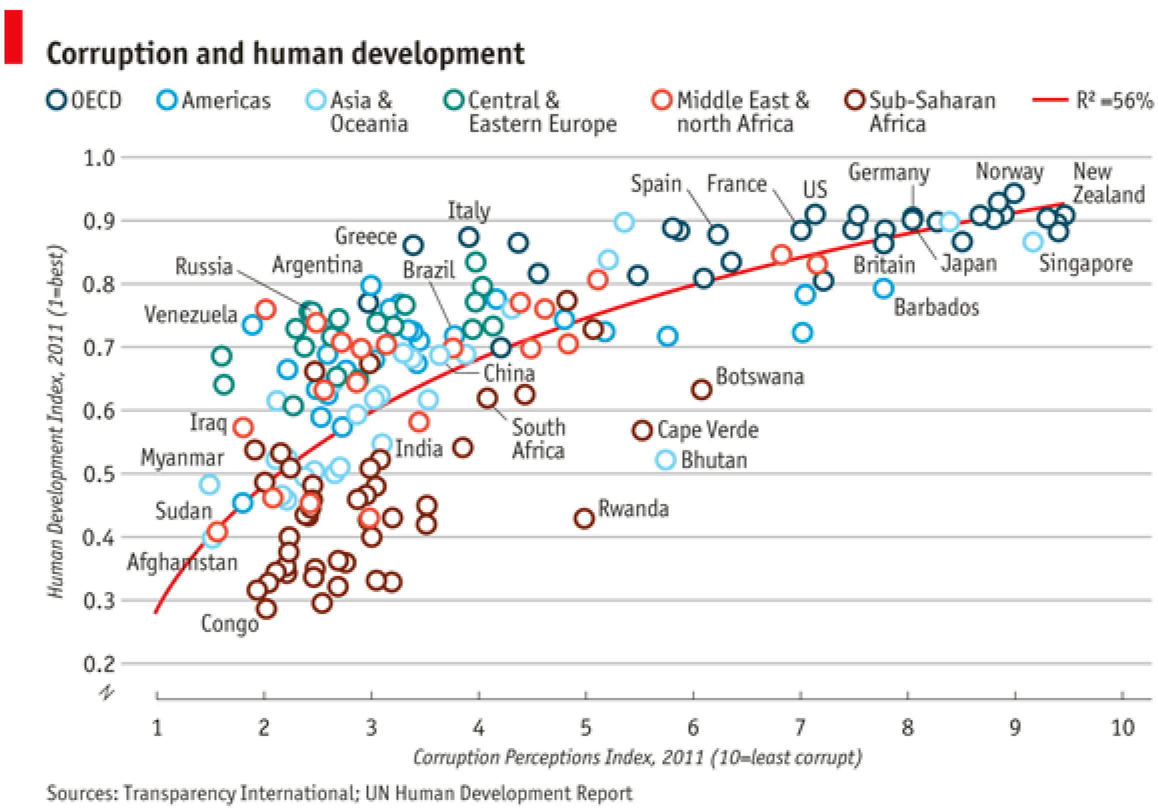
Computing for Data Science

FALL SEMESTER 2017

INSTRUCTOR: Mr. Atul Nag
atulnag@curaj.ac.in

Recreating a plot from Economist

recreating this plot from [The Economist](#):



1) Import the ggplot2 data.table libraries and use fread to load the csv file 'Economist_Assignment_Data.csv' into a dataframe called df (Hint: use drop=1 to skip the first column)
2) Check the head of df

3) Use ggplot() + geom_point() to create a scatter plot object called pl. You will need to specify x=CPI and y=HDI and color=Region as aesthetics
4) Change the points to be larger empty circles. (You'll have to go back and add arguments to geom_point() and reassign it to pl.) You'll need to figure out what shape= and size=
5) Add geom_smooth(aes(group=1)) to add a trend line
6) We want to further edit this trend line. Add the following arguments to geom_smooth (outside of aes):
    a) method = 'lm'
    b) formula = y ~ log(x)
    c) se = FALSE
    d) color = 'red'

    For more info on these arguments, check out the documentation under the Arguments list for details.

    Assign all of this to pl2

7) It's really starting to look similar! But we still need to add labels, we can use geom_text! Add geom_text(aes(label=Country)) to pl2 and see what happens. (Hint: It should be way too many labels)
8) Labeling a subset is actually pretty tricky! So we're just going to give you the answer since it would require manually selecting the subset of countries we want to label!
9) Almost there! Still not perfect, but good enough for this assignment. Later on we'll see why interactive plots are better for labeling. Now let's just add some labels and a theme, set the x and y scales and we're done!
    Add theme_bw() to your plot and save this to pl4
10) Add scale_x_continuous() and set the following arguments:
    - name = Same x axis as the Economist Plot
    - limits = Pass a vector of appropriate x limits
    - breaks = 1:10
11) Now use scale_y_continuous to do similar operations to the y axis!
12) Finally use ggtitle() to add a string as a title.

Bike Sharing Demand

For this project you will be doing the Bike Sharing Demand Kaggle challenge. The main point of this project is to get you feeling comfortable with Exploratory Data Analysis and begin to get an understanding that sometimes certain models are not a good choice for a data set.

In this case, we will discover that Linear Regression may not be the best choice given our data!

Get the Data

You can download the data or just use the supplied csv. The data has the following features:

- datetime - hourly date + timestamp
- season - 1 = spring, 2 = summer, 3 = fall, 4 = winter
- holiday - whether the day is considered a holiday
- workingday - whether the day is neither a weekend nor holiday
- weather -
    - 1: Clear, Few clouds, Partly cloudy, Partly cloudy
    - 2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist
    - 3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds
    - 4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog
- temp - temperature in Celsius
- atemp - "feels like" temperature in Celsius
- humidity - relative humidity
- windspeed - wind speed
- casual - number of non-registered user rentals initiated
- registered - number of registered user rentals initiated
- count - number of total rentals

1) Read in bikeshare.csv file and set it to a dataframe called bike.
2) Check the head of df
3) Can you figure out what is the target we are trying to predict?

Exploratory Data Analysis

4) Create a scatter plot of count vs temp. Set a good alpha value.
5) Plot count versus datetime as a scatterplot with a color gradient based on temperature. You'll need to convert the datetime column into POSIXct before plotting.
6) What is the correlation between temp and count?
7) Explore the season data. Create a boxplot, with the y axis indicating count and the x axis begin a box for each season.

Feature Engineering

8) Create an "hour" column that takes the hour from the datetime column. You'll probably need to apply some function to the entire datetime column and reassign it. Hint:

   ```
   time.stamp <- bike$datetime[4]
   format(time.stamp, "%H")
   ```

9) Now create a scatterplot of count versus hour, with color scale based on temp. Only use bike data where workingday==1.
   a) Optional Additions:
      i) Use the additional layer: scale_color_gradientn(colors=c('color1',color2,etc..)) where the colors argument is a vector gradient of colors you choose, not just high and low.
      ii) Use position=position_jitter(w=1, h=0) inside of geom_point() and check out what it does.
10) Now create the same plot for non working days

Building the Model

11) Use lm() to build a model that predicts count based solely on the temp feature, name it temp.model
12) Get the summary of the temp.model

Interpreting the intercept (β0):

- It is the value of y when x=0.
- Thus, it is the estimated number of rentals when the temperature is 0 degrees Celsius.
- Note: It does not always make sense to interpret the intercept.

Interpreting the "temp" coefficient (β1):

- It is the change in y divided by change in x, or the "slope".
- Thus, a temperature increase of 1 degree Celsius is associated with a rental increase of 9.17 bikes.
- This is not a statement of causation.
- β1 would be negative if an increase in temperature was associated with a decrease in rentals.

13) How many bike rentals would we predict if the temperature was 25 degrees Celsius? Calculate this two ways:
    a) Using the values we just got above
    b) Using the predict() function
14) Use sapply() and as.numeric to change the hour column to a column of numeric values.
15) Finally build a model that attempts to predict count based off of the following features. Figure out if theres a way to not have to pass/write all these variables into the lm() function. Hint: StackOverflow or Google may be quicker than the documentation.
    a) Season
    b) Holiday
    c) Workingday

    d) Weather
    e) Temp
    f) Humidity
    g) Windspeed
    h) hour (factor)
16) Get the summary of the model
17) Did the model perform well on the training data? What do you think about using a Linear Model on this data?

You should have noticed that this sort of model doesn't work well given our seasonal and time series data. We need a model that can account for this type of trend, read about Regression Forests for more info if you're interested!

18) Optional: See how well you can predict for future data points by creating a train/test split. But instead of a random split, your split should be "future" data for test, "previous" data for train.