THE UNIVERSITY OF AUCKLAND
NEW ZEALAND
Te Whare Wānanga o Tāmaki Makaurau

DATA to insight
An Introduction to Data Analysis

3.9 Exercise: Relationships between numeric variables – *R version*

Note: *Copying and pasting text (e.g. R code) from a pdf is not reliable. For that reason we have also provided the code in* ***a text file***

This exercise will enable you to construct graphs of two numeric variables as discussed in the previous video. The skills addressed are:

1. Creating a scatterplot of two numeric variables.
2. Adding a trend line (if suitable) and label any interesting points.
3. Train your eyes to observe and draw an envelope around the scatter.

We will use the **gapminder_2008** dataset in the **FutureLearnData** package.

Creating a scatterplot of two numeric variables

When we call **iNZightPlot(x,y)** using two numeric variables *x* and *y* we will get a scatterplot. The 1st variable (*x*) gets plotted against the horizontal axis and the 2nd variable (y) is plotted against the vertical axis. So we should put the ***predictor variable* first** and the ***outcome variable* second**.

[**Note:** *I've just seen that have reversed the order of the variables in this exercise from the way it was in the (older) iNZight version*. The order below is more natural. The change affects nothing that matters in the exercise but the graphs are "the other way around" from the iNZight version.**]**

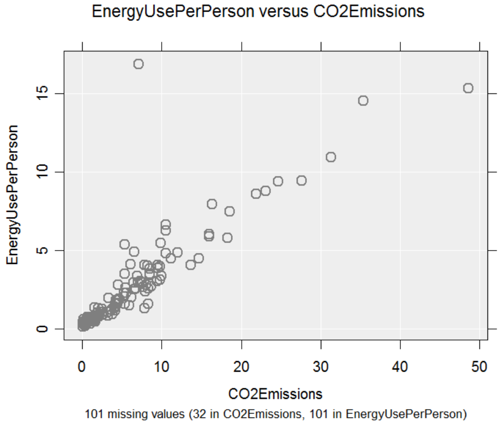
#R Code	Commentary

# Setup 	
library(iNZightPlots)
library(FutureLearnData)
 	
data(gapminder_2008)	

# Plot y=EnergyUsePerPerson** by **x=CO2emissions

```
iNZightPlot(CO2Emissions, EnergyUsePerPerson,
        data=gapminder_2008)
```

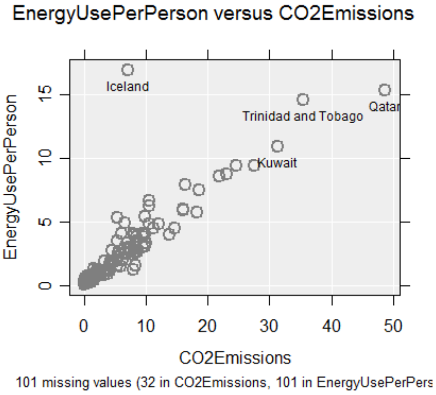
# Note the reversal in the order of the names between the call and ordinary English usage for what we want to do!



Recall: *the plot call is in the order iNZightPlot(x,y,…)*

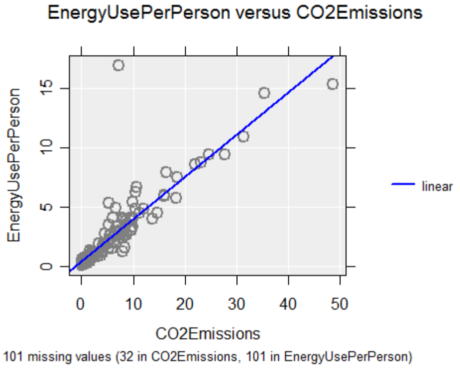
# Label the 4 most extreme points by** the variable **Country

```
iNZightPlot(CO2Emissions, EnergyUsePerPerson,
        data=gapminder_2008, locate.extreme=4, locate=Country)
```



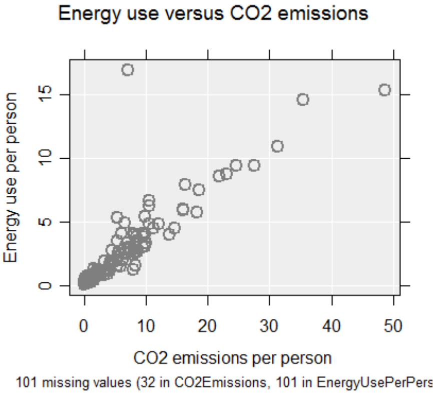
# Add a trend line

```
iNZightPlot(CO2Emissions, EnergyUsePerPerson,
        data=gapminder_2008, trend="linear")
```

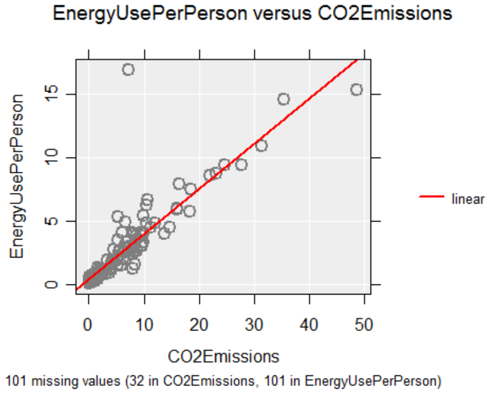


# Changing from the default Axes labels and title

```
iNZightPlot(CO2Emissions, EnergyUsePerPerson,
        data=gapminder_2008, xlab="CO2 emissions per person ",
        ylab="Energy use per person", main="Energy use versus CO2
        emissions")
```



```
# Put on a trend line but coloured red, instead of the default (blue)

iNZightPlot(CO2Emissions, EnergyUsePerPerson,
        data=gapminder_2008, trend="linear",
        col.trend=list(linear="red"))
```

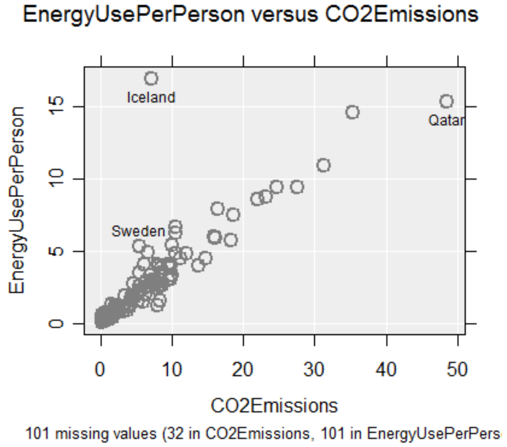


```
#  Label specified data points

ids = (1:nrow(gapminder_2008))[gapminder_2008$Country %in%
c("Qatar","Iceland","Sweden")]


iNZightPlot(CO2Emissions, EnergyUsePerPerson,
data=gapminder_2008, locate.id=ids, locate=Country)
```

Commentary

***Tell me** the row numbers of all the rows of **gapminder_2008** for which the value of Country is in the set listed in c(…).*
Store** these row numbers in **ids

*"**locate.id**" is used to tell iNZightPlot the row numbers of the points it should label*
*"**locate=Country**" tells iNZightPlot to use the contents of the variable Country to label those points*



The options added to the plots above can all be used together. Try putting several of them together in the same call to iNZightPlot. There are lots of others you will find out about over time, e.g. if you are putting a line on your plot then adding ", lwd=2" to your call (without the "") will double the thickness of the line.

Try to identify more or other countries (*spelling and lower/upper case is critical*).

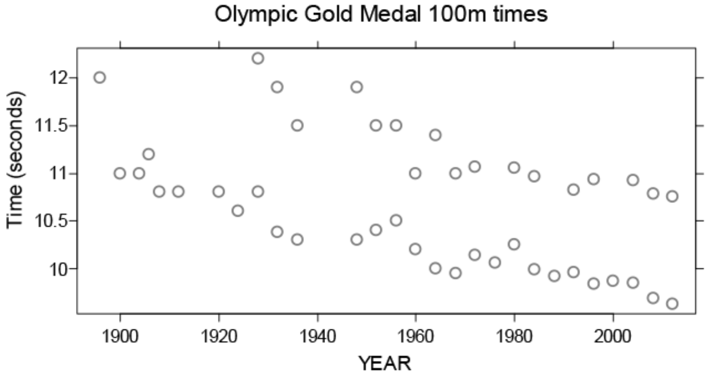
Try working with other variables.

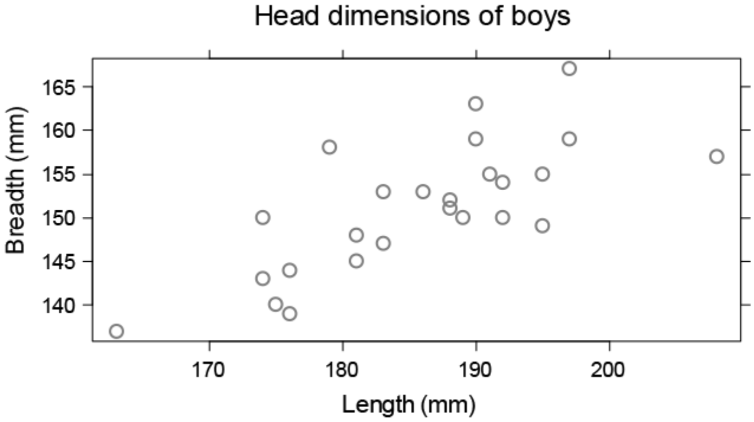
Observations of scatterplots

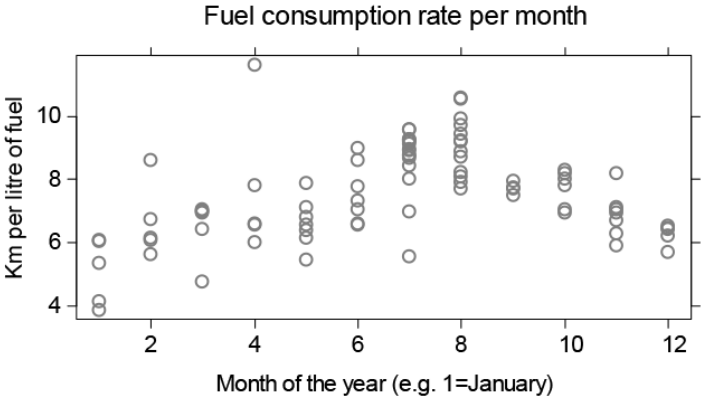
Not all scatterplots show obvious trends with uniform scatter.

For each scatterplot below, train your eye to look for trends, clusters, outliers and draw an envelope around the scatter. You may want to print them out so that you can draw on the above graphs.

For each graph, what do you notice? Post one of your findings on the discussion.







Optional: *Try this new feature (interactive web graphics)*

We will export an iNZightPlot graph as an *Interactive HTML* file and open this file up in our default browser. If that is a modern browser like Chrome, Firefox or Safari (but not Internet Explorer) this will then give you an interactive version of the graph that lets you query it in various ways like hovering over the points, or a trend line, or clicking them, or selecting more than one using the Ctrl or Shift keys, or by dragging.

The save process can be slow if there are a lot of dots to be drawn.

The save window allows other variables to be exported along with the plot. This is particularly useful for hover-over if you have a variable that gives the names of the people or objects.

You can give such files to others. They do not need to be connected to iNZight to work.

Here is sample code:

```
# Make a plot and also store the output in myplot
myplot = iNZightPlot(CO2Emissions, EnergyUsePerPerson, data=gapminder_2008, trend="linear")

# Specify a location to store an Interactive HTML file. I will call my file "myintplot.html"
# You will have to change the path to the file because this one is to a location on my desktop!
filepath = "C:/Users/myusername/Desktop/myintplot.html"

exportHTML(myplot, filepath)
browseURL(filepath)   #open the file up in my default browser
```

```
# If the plot is truncated make your R plotting window smaller
```

```
# To add extra variables, e.g. Country …
exportHTML(myplot, filepath, data=gapminder_2008, extra.vars = c("Country", "Populationtotal"))
```

To discuss R issues related to this Exercise,

go to https://gitter.im/iNZightVIT/d2i-R-discussion

If your question relates to an Exercise, say which one you are talking about!