ggplot

Francisco Guzman

May 12, 2016

In this recipe, we will learn:

- How to install packages (reinforce)
- How to load data from a file
- How to plot nice scatter plots with ggplot
- How to save your graph as a jpg and pdf (command line)

```
#Let's install gglot (ggplot2) through the command line
#install.packages('ggplot2')
library('ggplot2')
```

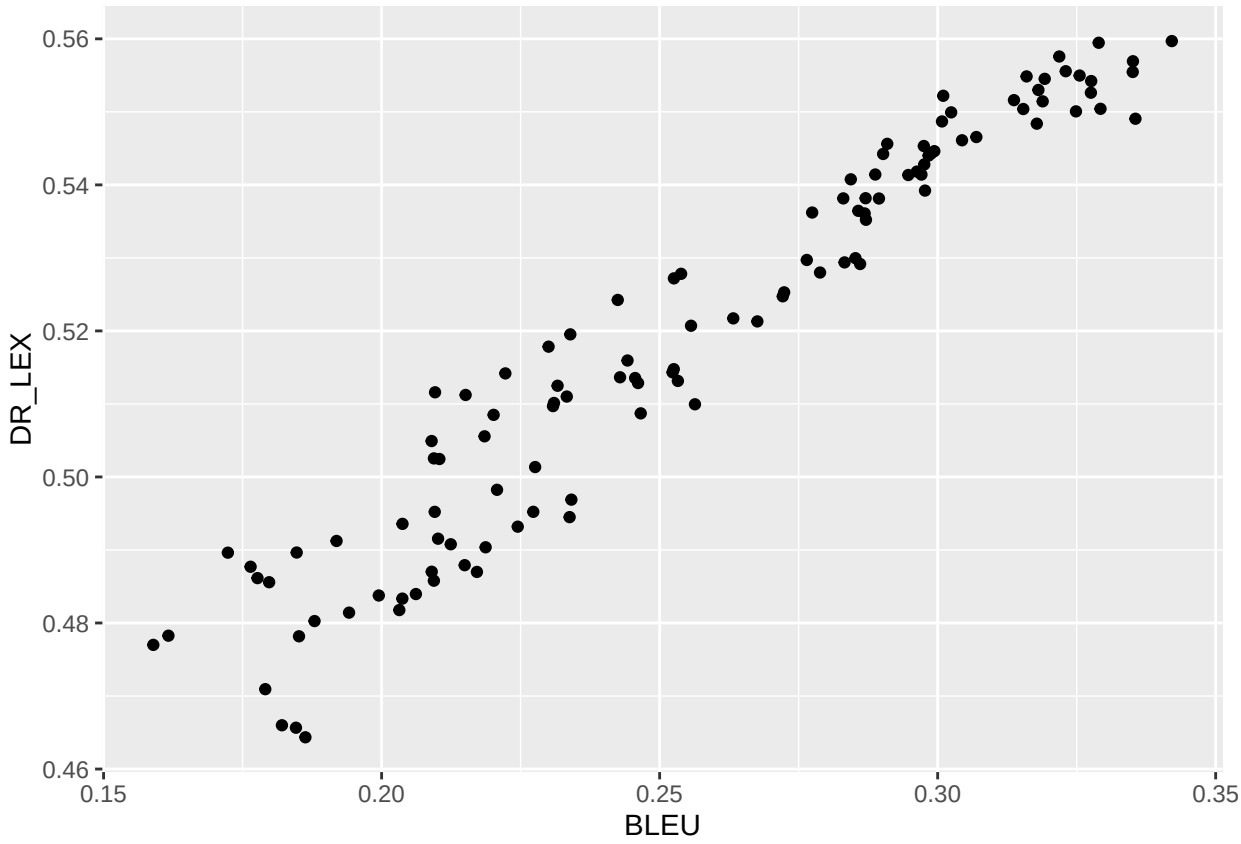
```
# We use read.table, we give the arguments:
# header=false, indicating that the columns names are not in the first line of the dataset
# col.names, indicating the names of the columns manually,
# sep, the separator (default "\t" tab)
 df <-read.table('data/mte_metrics2.dat',header=FALSE,col.names = c("lang_pair","testset","system","BLE

#let's look at the head
head(df)
```

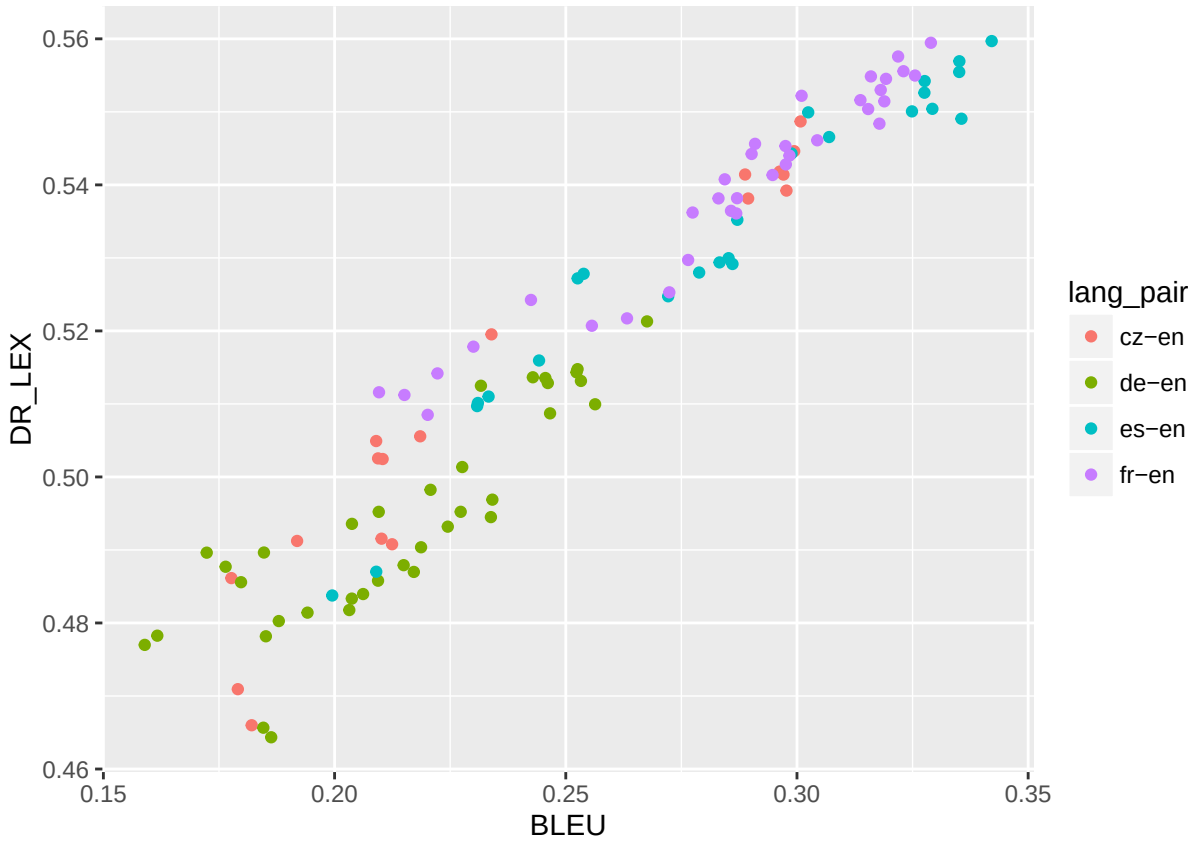
```
>>>   lang_pair             testset                         system      BLEU
>>> 1     cz-en newssyscombtest2011                     bbn-combo 0.3007811
>>> 2     cz-en newssyscombtest2011             cmu-heafield-combo 0.2887835
>>> 3     cz-en newssyscombtest2011 cmu-heafield-combo-contrastive 0.2894421
>>> 4     cz-en newssyscombtest2011                            cst 0.1820762
>>> 5     cz-en newssyscombtest2011                cst-contrastive 0.1790802
>>> 6     cz-en newssyscombtest2011                       cu-bojar 0.2094103
>>>      DR_LEX
>>> 1 0.5486888
>>> 2 0.5414202
>>> 3 0.5381331
>>> 4 0.4659940
>>> 5 0.4709361
>>> 6 0.5025401
```

```
#this file shows the scores given by two different MT metrics: BLEU and DR_LEX to the same set of trans
```

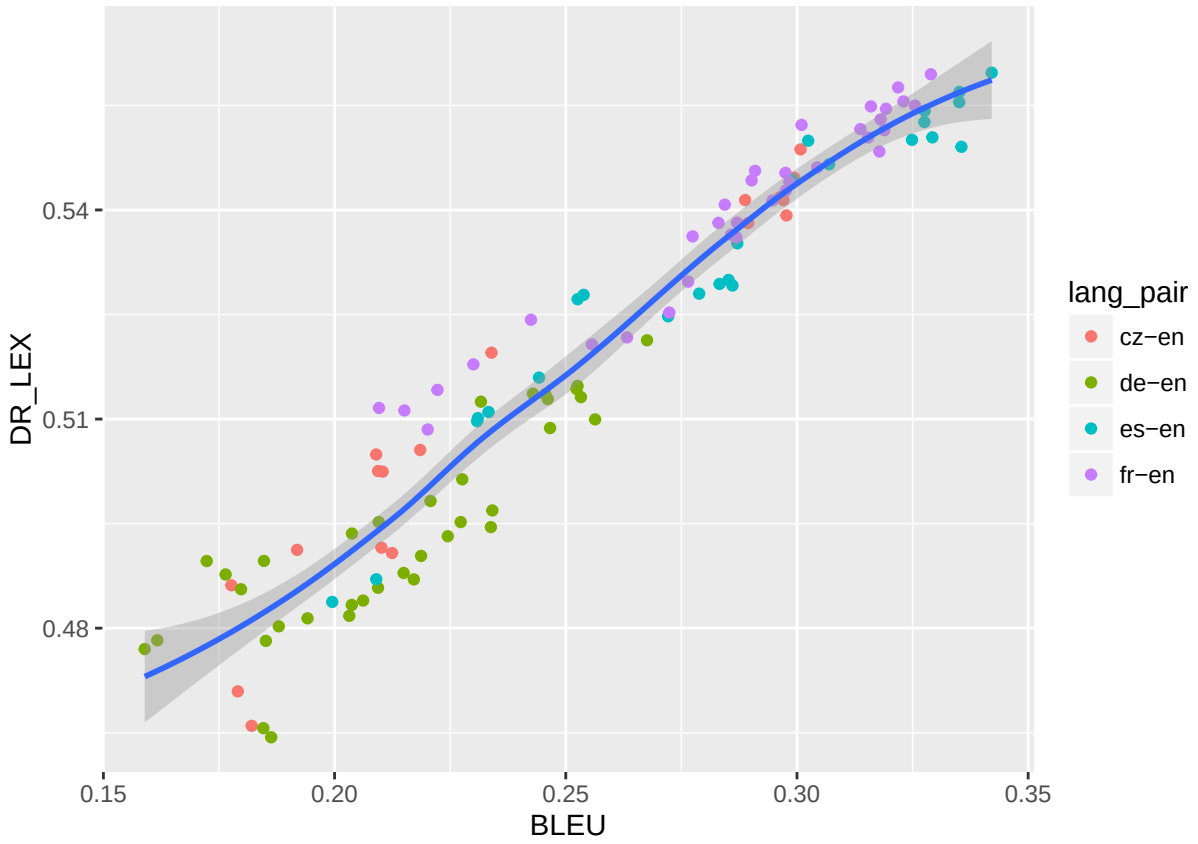
```
#Let's plot the basic ggplot
# here we give the dataframe
# the aes= aesthetics
# and then the geometry, in this case we want a scatter plot of BLEU vs DR_LEX
ggplot(df,aes(x=BLEU,y=DR_LEX)) + geom_point()
```



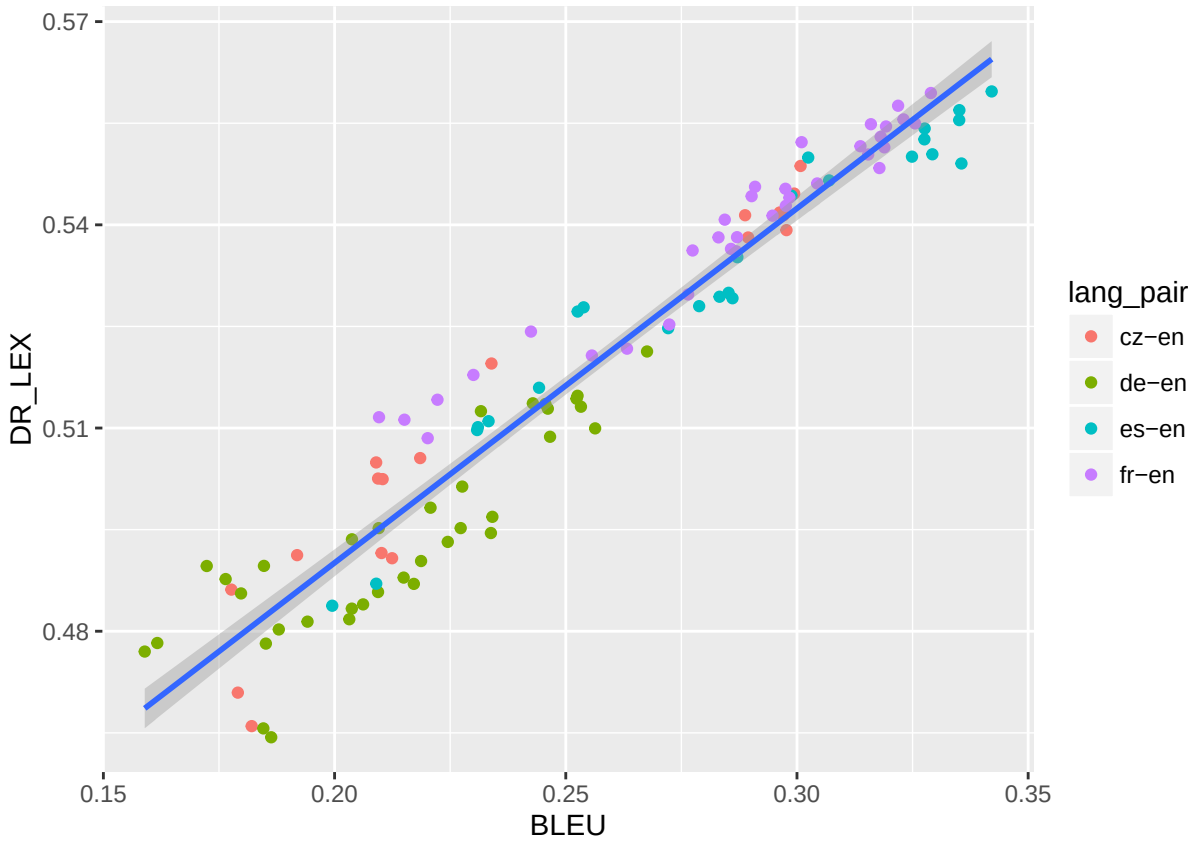

```
#Now, let's color each datapoint according to the language pair of the system
# to do this, we specify another aesthetic, which only applies to the geom_point
ggplot(df,aes(x=BLEU,y=DR_LEX)) + geom_point(aes(col=lang_pair))
```



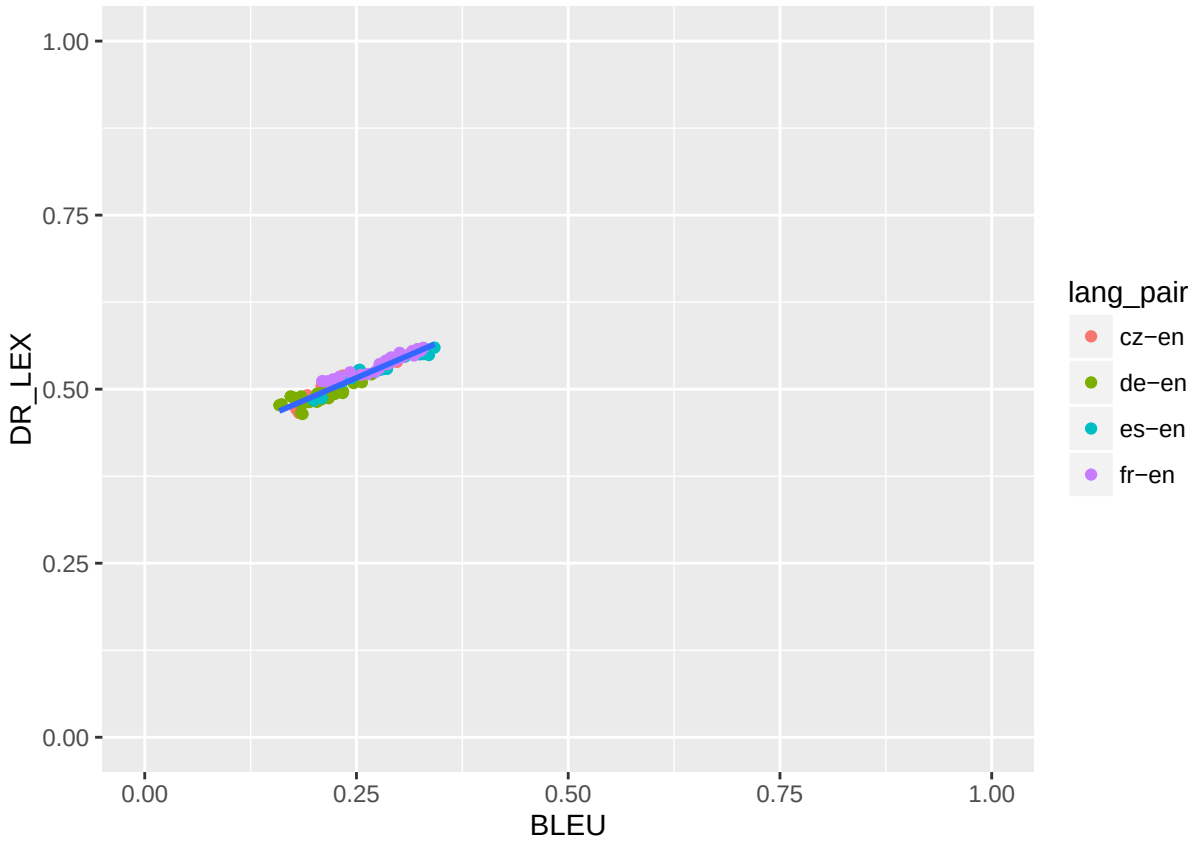

```
#Now, let's add a trendline (by default, loess trendline)
ggplot(df,aes(x=BLEU,y=DR_LEX)) + geom_point(aes(col=lang_pair)) +geom_smooth()
```



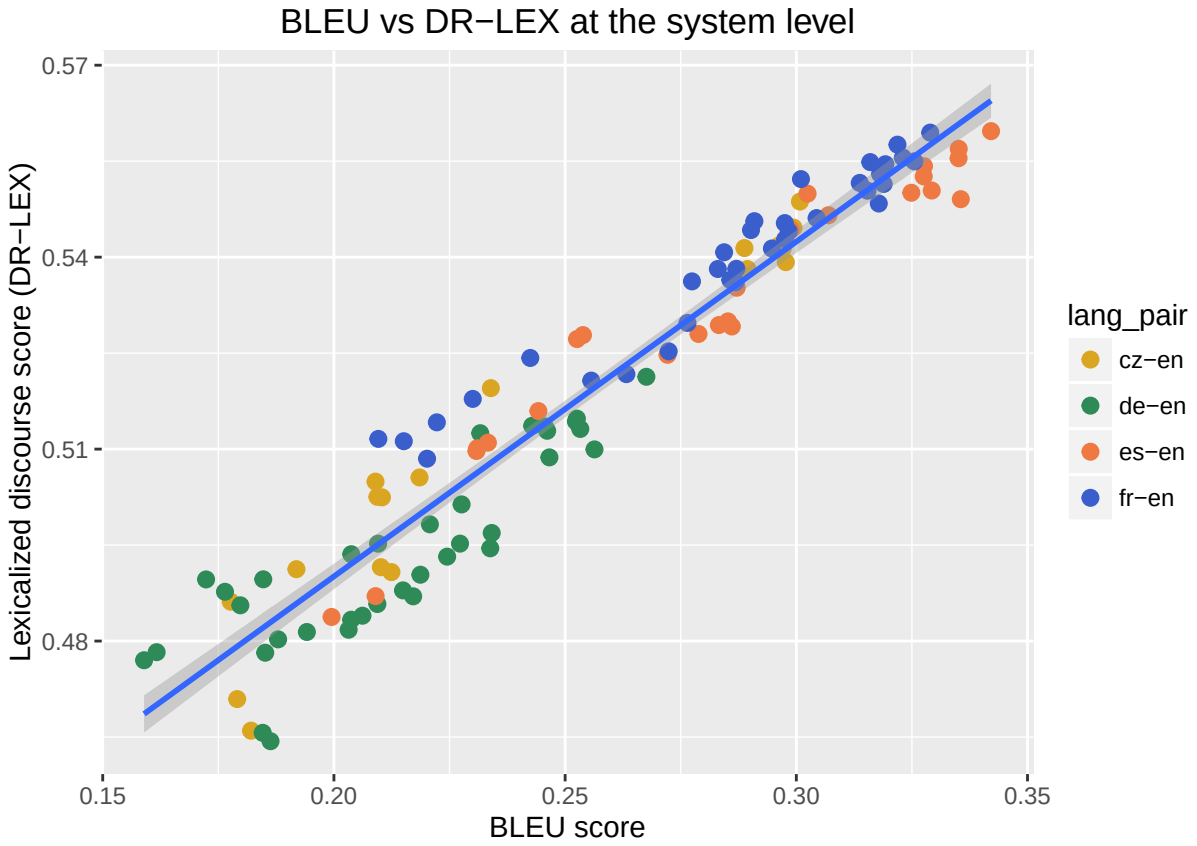

```
# We could choose a linear trendline "lm"= linear model
ggplot(df,aes(x=BLEU,y=DR_LEX)) + geom_point(aes(col=lang_pair)) +geom_smooth(method="lm")
```




```
#If we want, we can modify the scale, so both metrics are plotted from 0..1 (not very helpful)
ggplot(df,aes(x=BLEU,y=DR_LEX)) + geom_point(aes(col=lang_pair)) +geom_smooth(method="lm") + ylim(0,1)+
```




```
#Let's go back to the previous version and add a title (ggtitle), labels and colors and point sizes
ggplot(df,aes(x=BLEU,y=DR_LEX)) + geom_point(aes(col=lang_pair),size=2.5) +geom_smooth(method="lm")+ gg
```




```
#you can check additional colors by using method colors()
```

1 Saving to a file

```
myplot<-ggplot(df,aes(x=BLEU,y=DR_LEX)) + geom_point(aes(col=lang_pair),size=2.5) +geom_smooth(method="


# Saving to a file is done via "devices". By declaring a PDF device, we tell R to plot the
pdf("img/my_first_ggplot.pdf",height=10,width=10) #you can control the aspect by modifiying the width/h
print(myplot)
dev.off() # we tell to finish writing in the pdf device
```

```
>>> RStudioGD
>>>         2
```