Assignment: Exploratory Data Analysis

David Vichansky

20-11-2020

Here is an example file you can write.

First, load the packages:

```r
library(ISLR)
library(tidyverse)
```

```
## -- Attaching packages ------------------------------ tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.1
## v tidyr   1.1.1     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.0
```

```
## -- Conflicts --------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(haven)
library(readxl)
library(tinytex)
library(ggplot2)
library(scales)
```

```
##
## Attaching package: 'scales'
```

```
## The following object is masked from 'package:purrr':
##
##     discard
```

```
## The following object is masked from 'package:readr':
##
##     col_factor
```

Load '.csv' files for CWUR, Shanghai and Times data sets.

```r
cwur <- read.csv("data/cwurData.csv", header = TRUE)
shanghai <- read.csv("data/shanghaiData.csv", header = TRUE)
times <- read.csv("data/timesData.csv", header = TRUE)
```

Inspect the data headings to get a feel of what to work on.

```r
# cwur
head(cwur)
```

```
##   world_rank                           institution        country national_rank
## 1          1                    Harvard University            USA             1
## 2          2 Massachusetts Institute of Technology            USA             2
## 3          3                   Stanford University            USA             3
## 4          4               University of Cambridge United Kingdom             1
## 5          5    California Institute of Technology            USA             4
## 6          6                  Princeton University            USA             5
##   quality_of_education alumni_employment quality_of_faculty publications
## 1                    7                 9                  1            1
## 2                    9                17                  3           12
## 3                   17                11                  5            4
## 4                   10                24                  4           16
## 5                    2                29                  7           37
## 6                    8                14                  2           53
##   influence citations broad_impact patents  score year
## 1         1         1           NA       5 100.00 2012
## 2         4         4           NA       1  91.67 2012
## 3         2         2           NA      15  89.50 2012
## 4        16        11           NA      50  86.17 2012
## 5        22        22           NA      18  85.21 2012
## 6        33        26           NA     101  82.50 2012
```

```r
#lapply(cwur, class)

# shanghai
head(shanghai)
```

```
##   world_rank                             university_name national_rank
## 1          1                          Harvard University             1
## 2          2                     University of Cambridge             1
## 3          3                         Stanford University             2
## 4          4          University of California, Berkeley             3
## 5          5 Massachusetts Institute of Technology (MIT)             4
## 6          6          California Institute of Technology             5
##   total_score alumni award  hici    ns   pub   pcp year
## 1       100.0  100.0 100.0 100.0 100.0 100.0  72.4 2005
## 2        73.6   99.8  93.4  53.3  56.6  70.9  66.9 2005
## 3        73.4   41.1  72.2  88.5  70.9  72.3  65.0 2005
## 4        72.8   71.8  76.0  69.4  73.9  72.2  52.7 2005
## 5        70.1   74.0  80.6  66.7  65.8  64.3  53.0 2005
## 6        67.1   59.2  68.6  59.8  65.8  52.5 100.0 2005
```

```
#lapply(shanghai, class)

# times
head(times)
```

```
##   world_rank                       university_name                  country
## 1          1                    Harvard University United States of America
## 2          2    California Institute of Technology United States of America
## 3          3 Massachusetts Institute of Technology United States of America
## 4          4                   Stanford University United States of America
## 5          5                  Princeton University United States of America
## 6          6               University of Cambridge           United Kingdom
##   teaching international research citations income total_score num_students
## 1     99.7          72.4     98.7      98.8   34.5        96.1       20,152
## 2     97.7          54.6     98.0      99.9   83.7        96.0        2,243
## 3     97.8          82.3     91.4      99.9   87.5        95.6       11,074
## 4     98.3          29.5     98.1      99.2   64.3        94.3       15,596
## 5     90.9          70.3     95.4      99.9      -        94.2        7,929
## 6     90.5          77.7     94.1      94.0   57.0        91.2       18,812
##   student_staff_ratio international_students female_male_ratio year
## 1                 8.9                    25%                   2011
## 2                 6.9                    27%           33 : 67 2011
## 3                 9.0                    33%           37 : 63 2011
## 4                 7.8                    22%           42 : 58 2011
## 5                 8.4                    27%           45 : 55 2011
## 6                11.8                    34%           46 : 54 2011
```

```
#lapply(times, class)
```

Obtain the range of years for which our data sets cover.

```
cwur %>% summarise(
  min = min(year),
  max = max(year)
)
```

```
##    min  max
## 1 2012 2015
```

```
shanghai %>% summarise(
  min = min(year),
  max = max(year)
)
```

```
##    min  max
## 1 2005 2015
```

```
times %>% summarise(
  min = min(year),
  max = max(year)
)
```

```
##    min  max
## 1 2011 2016
```

We would like to work with only a select few columns such as name, county, world ranking, national ranking (when exists), total score and year. Thus create new data frames, whilst renaming the column names to homogenous names. Furthermore, add a column to reflect which data set it is from and filter based on common years (2012-2015) only.

```
cwur <- cwur %>%
  select(world_rank, institution, country, national_rank, score, year) %>%
  rename( university_name = institution, total_score = score) %>%
  filter(year %in% (2012:2015)) %>%
  add_column(publication = "cwur")

shanghai <- shanghai %>%
  select(world_rank, university_name, national_rank, total_score, year) %>%
  filter(year %in% (2012:2015)) %>%
  add_column(publication = "shanghai")

times <- times %>%
  select(world_rank, university_name, country, total_score, year) %>%
  filter(year %in% (2012:2015)) %>%
  add_column(publication = "times")
```

We now assume that each university name is stated the same way in all three of our data frames, we do this in order to add ('mutate') the columns which are missing to make all three data frames the same size (identical) by performing an 'inner join'. Note: filter to remove N/A becasue our assumption is not bullet-proof.

```
shanghai <- shanghai %>%
  mutate(university_name = cwur$university_name[match(shanghai$university_name, cwur$uni
  mutate(country = cwur$country[match(shanghai$university_name, cwur$university_name)])
  filter(!is.na(university_name), !is.na(country))

times <- times %>%
  mutate(university_name = cwur$university_name[match(times$university_name, cwur$univer
  mutate(country = cwur$country[match(times$university_name, cwur$university_name)]) %>%
  add_column(national_rank = NA) %>% # because no such data exists
  filter(!is.na(university_name), !is.na(country))
```

Now that we are satisfied that our dataframe are of the same (column) size, let us combine them into one giant data frame which would help some of the data analysis/visualisation work we will do later on. Also round the 'total_score' to nearest decimal point.

```
ranking <- rbind(cwur, shanghai, times)
```

Check the data types in data frame.

```
lapply(ranking, class)
```

```
## $world_rank
## [1] "character"
##
## $university_name
## [1] "factor"
##
## $country
## [1] "factor"
##
## $national_rank
## [1] "character"
##
## $total_score
## [1] "character"
##
## $year
## [1] "integer"
##
## $publication
## [1] "character"
```

Convert 'world_rank', 'national_rank' and 'total_score' to numerical values.

```
ranking <- ranking %>%
  mutate(world_rank=as.numeric(world_rank)) %>%
  mutate(national_rank=as.numeric(national_rank)) %>%
  mutate(total_score=as.numeric(total_score))
```

```
## Warning: Problem with `mutate()` input `world_rank`.
## x NAs introduced by coercion
## i Input `world_rank` is `as.numeric(world_rank)`.
```

```
## Warning in mask$eval_all_mutate(dots[[i]]): NAs introduced by coercion
```

```
## Warning: Problem with `mutate()` input `national_rank`.
## x NAs introduced by coercion
## i Input `national_rank` is `as.numeric(national_rank)`.
```

```
## Warning in mask$eval_all_mutate(dots[[i]]): NAs introduced by coercion
```

```
## Warning: Problem with `mutate()` input `total_score`.
## x NAs introduced by coercion
## i Input `total_score` is `as.numeric(total_score)`.
```

```
## Warning in mask$eval_all_mutate(dots[[i]]): NAs introduced by coercion
```

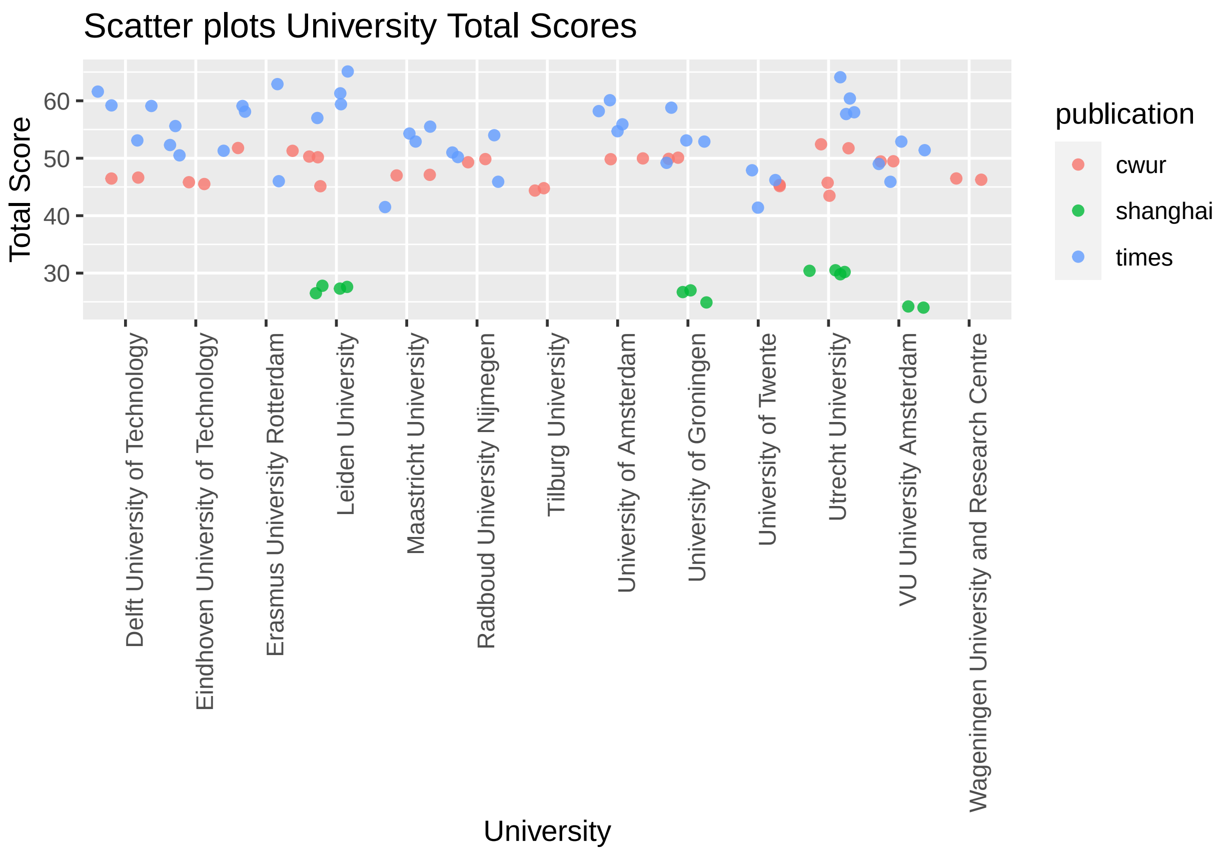
```r
lapply(ranking,class)
```

```
## $world_rank
## [1] "numeric"
##
## $university_name
## [1] "factor"
##
## $country
## [1] "factor"
##
## $national_rank
## [1] "numeric"
##
## $total_score
## [1] "numeric"
##
## $year
## [1] "integer"
##
## $publication
## [1] "character"
```

Plot the scores for universities in the Netherlands and rotate the x-axis.

```r
ggplot(subset(ranking, country == "Netherlands"),
       aes(x = university_name, y = total_score, colour = publication)) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  geom_point(alpha = 0.8,  position = position_jitter()) +
  labs(title = "Scatter plots University Total Scores", x = "University", y = "Total Sco
```

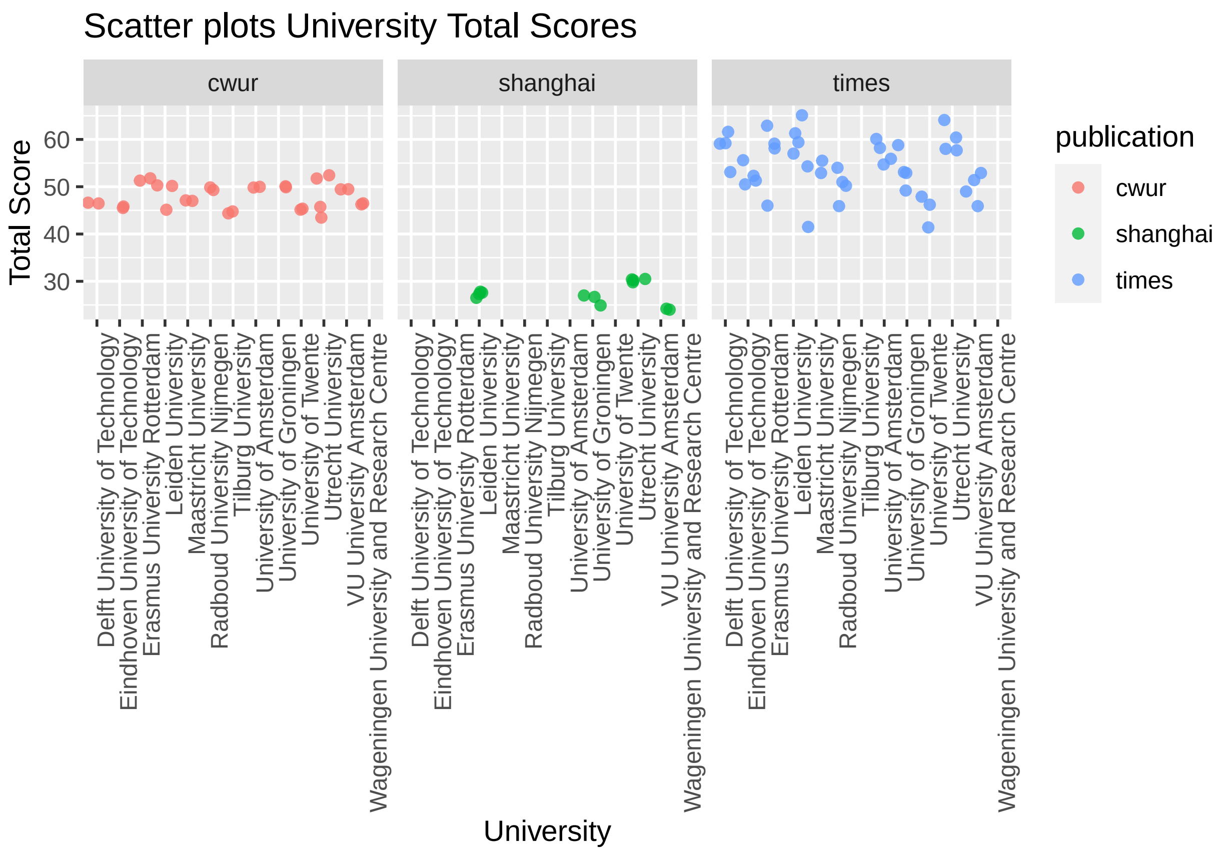
```
## Warning: Removed 23 rows containing missing values (geom_point).
```



Plot the scores for universities in the Netherlands and 'facet_wrap' by publication and year.

```
ggplot(subset(ranking, country == "Netherlands"),
       aes(x = university_name, y = total_score, colour = publication)) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  geom_point(alpha = 0.8,  position = position_jitter()) +
  labs(title = "Scatter plots University Total Scores", x = "University", y = "Total Sco
  facet_wrap(~ publication)
```

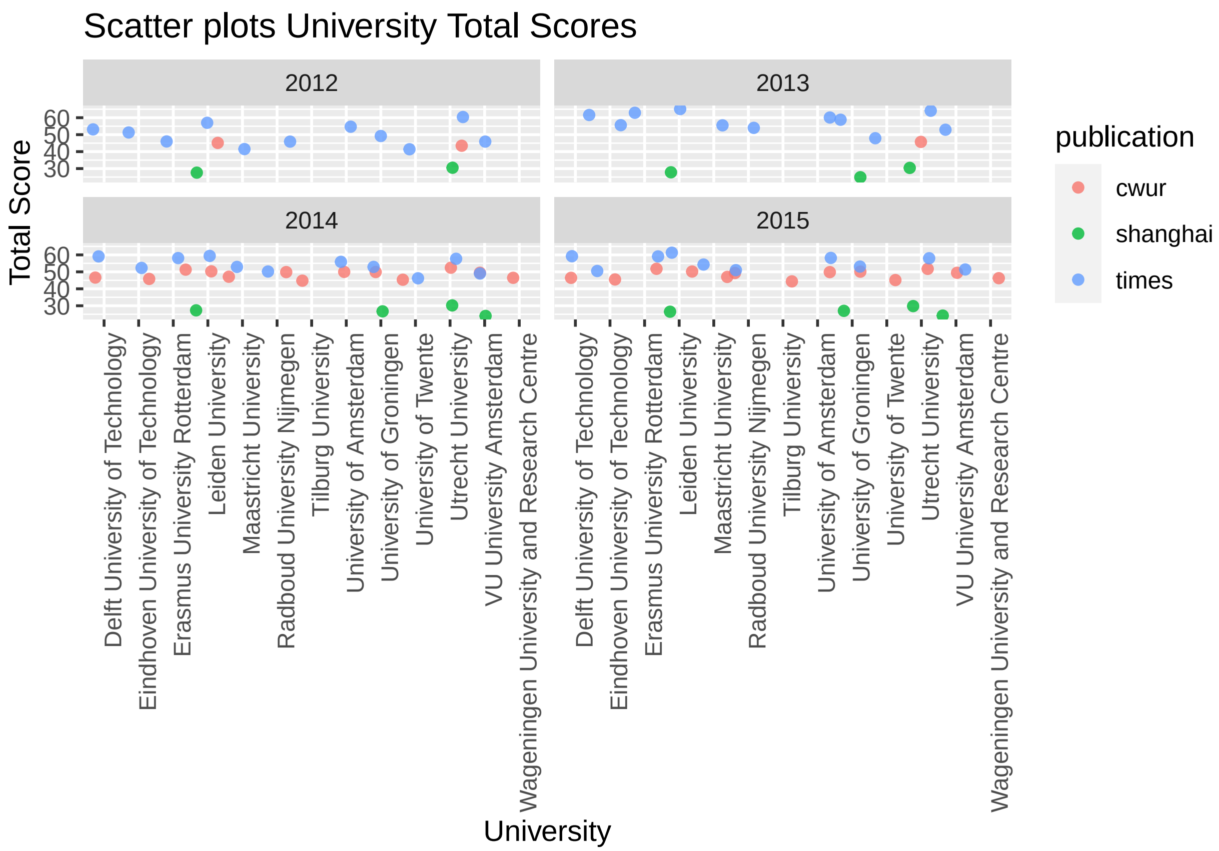
```
## Warning: Removed 23 rows containing missing values (geom_point).
```



Plot the scores for universities in the Netherlands and 'facet_wrap' by publication.

```
ggplot(subset(ranking, country == "Netherlands"),
       aes(x = university_name, y = total_score, colour = publication)) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  geom_point(alpha = 0.8,  position = position_jitter()) +
  labs(title = "Scatter plots University Total Scores", x = "University", y = "Total Sco
  facet_wrap(~ year)
```

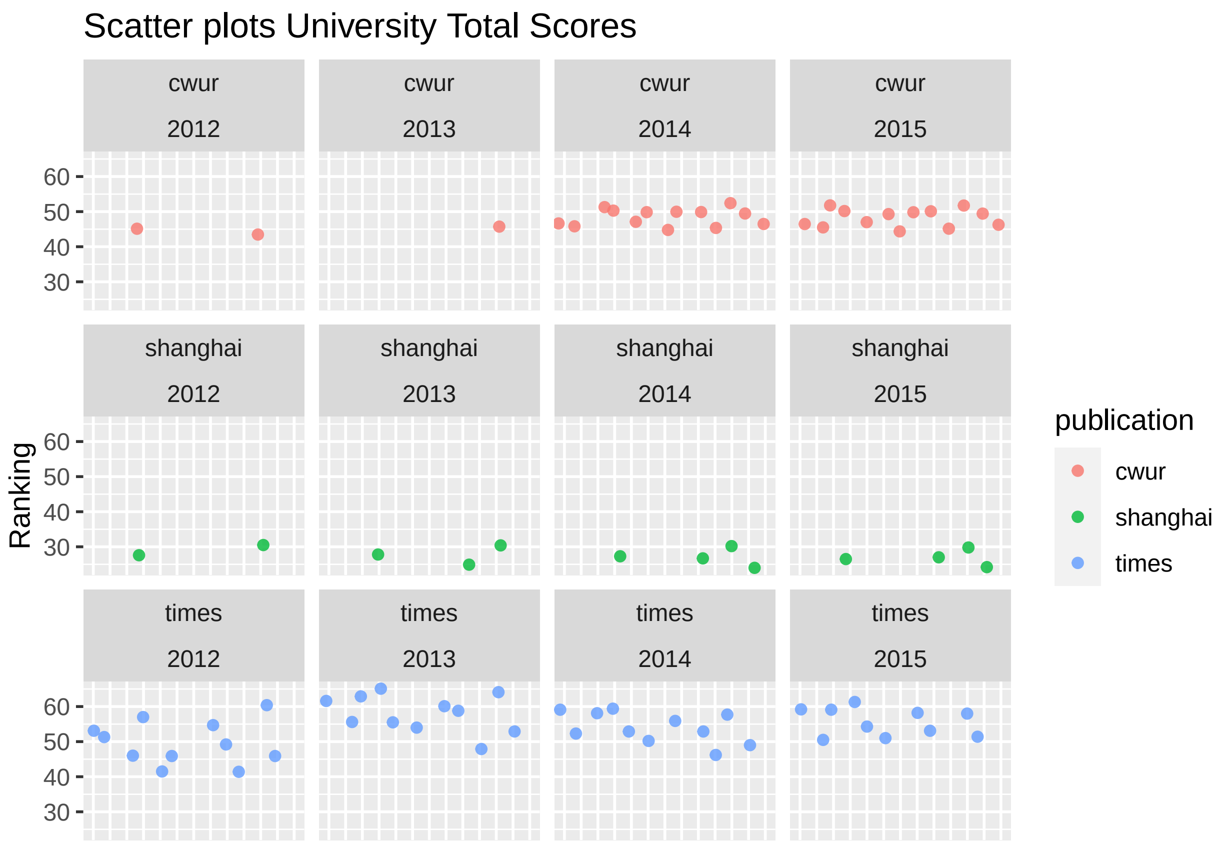
```
## Warning: Removed 23 rows containing missing values (geom_point).
```



Plot the scores for universities in the Netherlands and 'facet_grid' by publication and year. Note: we perform a 'facet_grid' rather than a 'facet_wrap' here because otherwise the rankings become culuated and the values go over '100'.

```
ggplot(subset(ranking, country == "Netherlands"),
       aes(x = university_name, y = total_score, colour = publication)) +
  theme(axis.text.x=element_blank(),
        axis.title.x=element_blank(),
        axis.ticks.x=element_blank()) +
  geom_point(alpha = 0.8,  position = position_jitter()) +
  labs(title = "Scatter plots University Total Scores", x = "University", y = "Ranking")
  facet_wrap(publication ~ year) +
  xlab(NULL)
```

```
## Warning: Removed 23 rows containing missing values (geom_point).
```



Firstly, it was important to determine what data we work with. We therefore created new data frames and only included the relevant columns for the name of university, world ranking, national ranking, country, total score and year. However, because the data frames were of different sizes and had different column headings, we renamed certain columns, changed the data types where needed and matched the missing fields from other data frames by matching using the university name.

Secondly, we used rbind() to join the three data frames into one large data frame called 'ranking' in order to plot and compare the three different publications. We had decided to compare the ranking of universities in one country and chose to filter the data by the Netherlands. The rankings had a too much varied scale so we chose to compare the difference we chose the fields 'Total Score' because this was out of '100' so somewhat of a standardised scale.

Thirdly, after comparing the three different publications we watned to also plot the different across years. However, using 'facet_wrap' would "squash" the plots. Therefore we opted for 'facet_grid' instead in order to split by publication & year and had removed the x-axis labels. Thus we had achieved a 3x4 grid of scatter plots which visually looks more appealing.