

Introduction to Linear Regression with R

Introduction to Econometrics

Antonio Rodriguez

24 enero 2023

Rmarkdown themes

The overall theme of your notebook is controlled by the option `theme` in the `yaml` preamble. Supported themes include `cerulean`, `cosmo`, `flatly`, `journal`, `lumen`, `paper`, `readable`, `sandstone`, `simplex`, `spacelab`, `united`, and `yeti`.

The highlighting theme is controlled by the option `highlight`, usually placed immediately below the theme. Supported styles include `default`, `tango`, `pygments`, `kate`, `monochrome`, `espresso`, `zenburn`, `haddock`, `breezedark`, and `textmate`.

See this gallery for examples. For more themes, you can use the extension package `prettydoc`. And you can also modify existing styles, or even create your own style from scratch, with `css` modifiers. See immediately below the `yaml` preamble of the source `Rmd` file for a simple example.

Load dataset

```
library(readxl)
df <- read_xlsx("C:/Users/anton/OneDrive/Escritorio/Growth.xlsx", trim_ws=TRUE)
head(df)
```



```
## # A tibble: 6 x 8
##   country_name growth    oil rgdp60 tradeshare yearsschool rev_coups assasinat-1
##   <chr>          <dbl> <dbl> <dbl>      <dbl>      <dbl>      <dbl>      <dbl>
## 1 India          1.92     0   766.      0.141      1.45      0.133      0.867
## 2 Argentina      0.618     0  4462.      0.157      4.99      0.933      1.93
## 3 Japan          4.30     0  2954.      0.158      6.71       0        0.200
## 4 Brazil         2.93     0  1784.      0.160      2.89      0.100      0.100
## 5 United States  1.71     0  9895.      0.161      8.66       0        0.433
## 6 Bangladesh    0.708     0   952.      0.221      0.790     0.306      0.175
## # ... with abbreviated variable name 1: assasinations
```

Use Pablo Casas's package

```
df_status(df) # Give information about zeros and missing values and type of variables
```

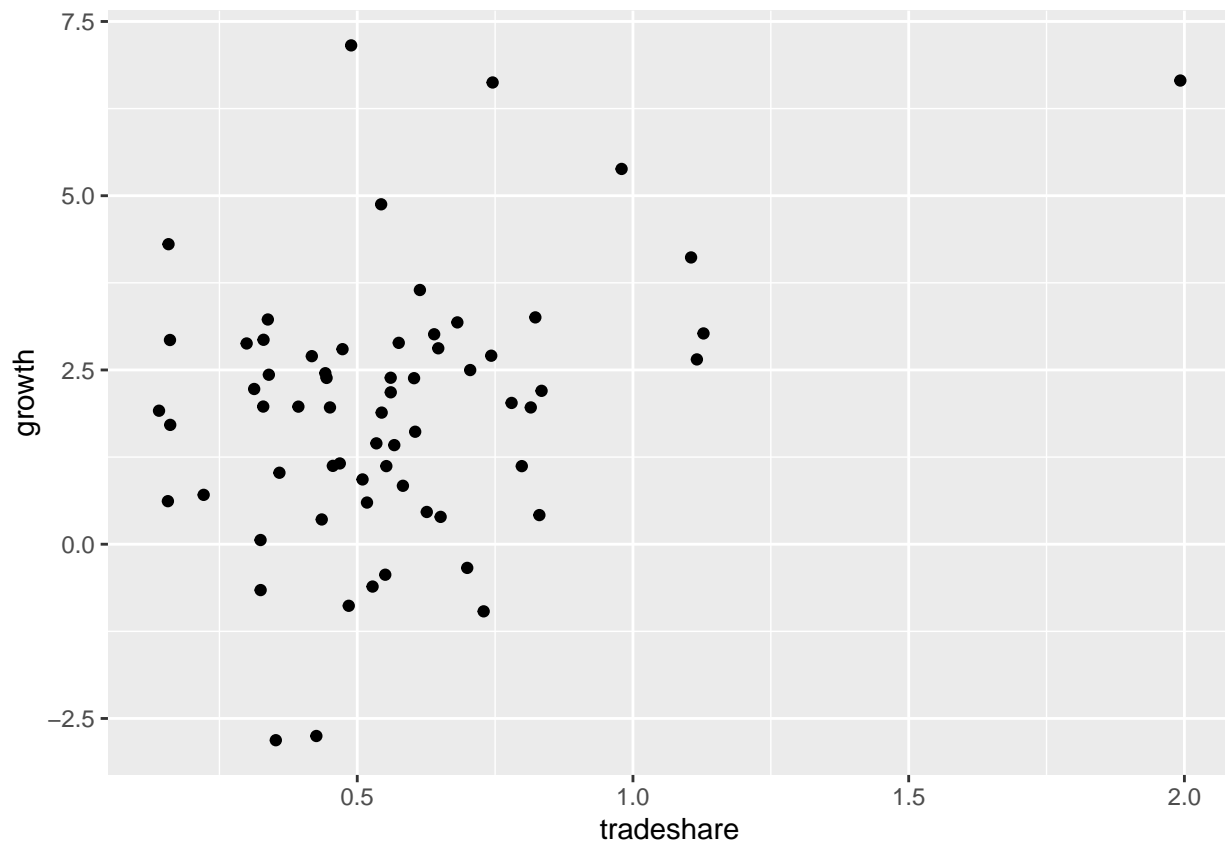


```
##           variable q_zeros p_zeros q_na p_na q_inf p_inf      type unique
## 1 country_name      0     0.00   0    0    0    0 character      65
```

```
## 2      growth      0    0.00    0    0    0    0    numeric    65
## 3      oil      65  100.00    0    0    0    0    numeric     1
## 4     rgdp60     0    0.00    0    0    0    0    numeric    65
## 5  tradeshare     0    0.00    0    0    0    0    numeric    65
## 6  yearsschool     0    0.00    0    0    0    0    numeric    63
## 7    rev_coups    22   33.85    0    0    0    0    numeric    23
## 8 assassinations  18   27.69    0    0    0    0    numeric    26
```

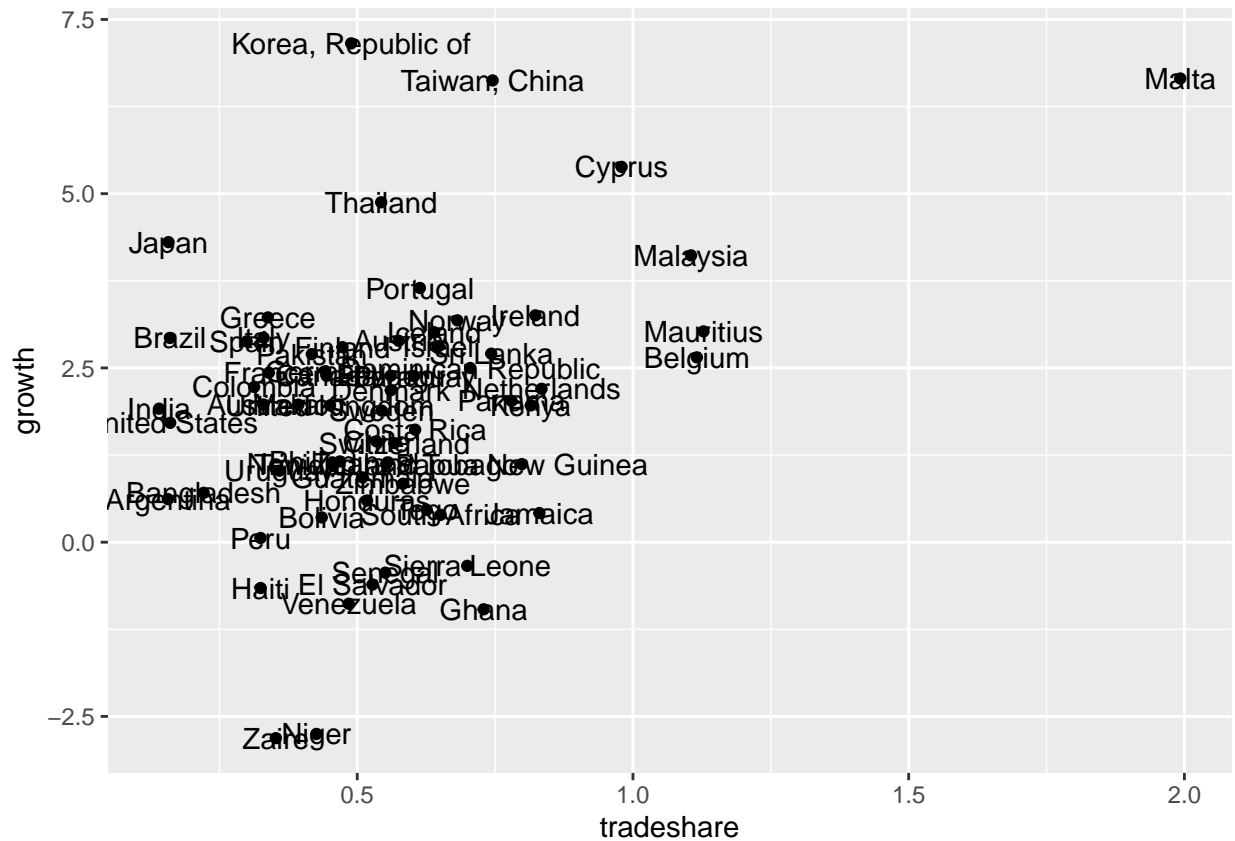
Make a scatterplot of average annual growth rate and average trade share:

```
library(ggplot2)
df$country <- as.factor(df$country_name)
ggplot(data=df, aes(x=tradeshare, y=growth)) + geom_point()
```



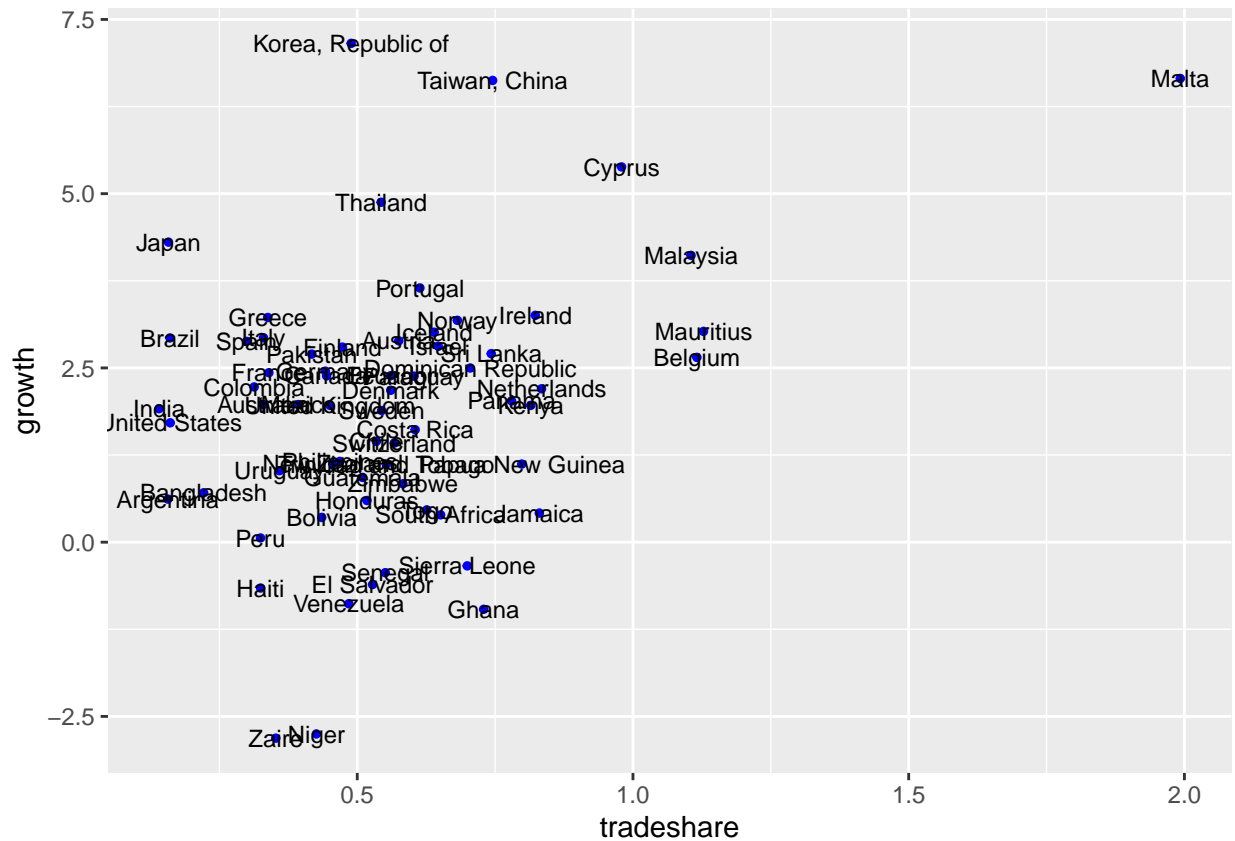
Detect the outlier: Print the country name

```
ggplot(data=df, aes(x=tradeshare, y=growth, label=country)) +
  geom_point() +
  geom_text()
```



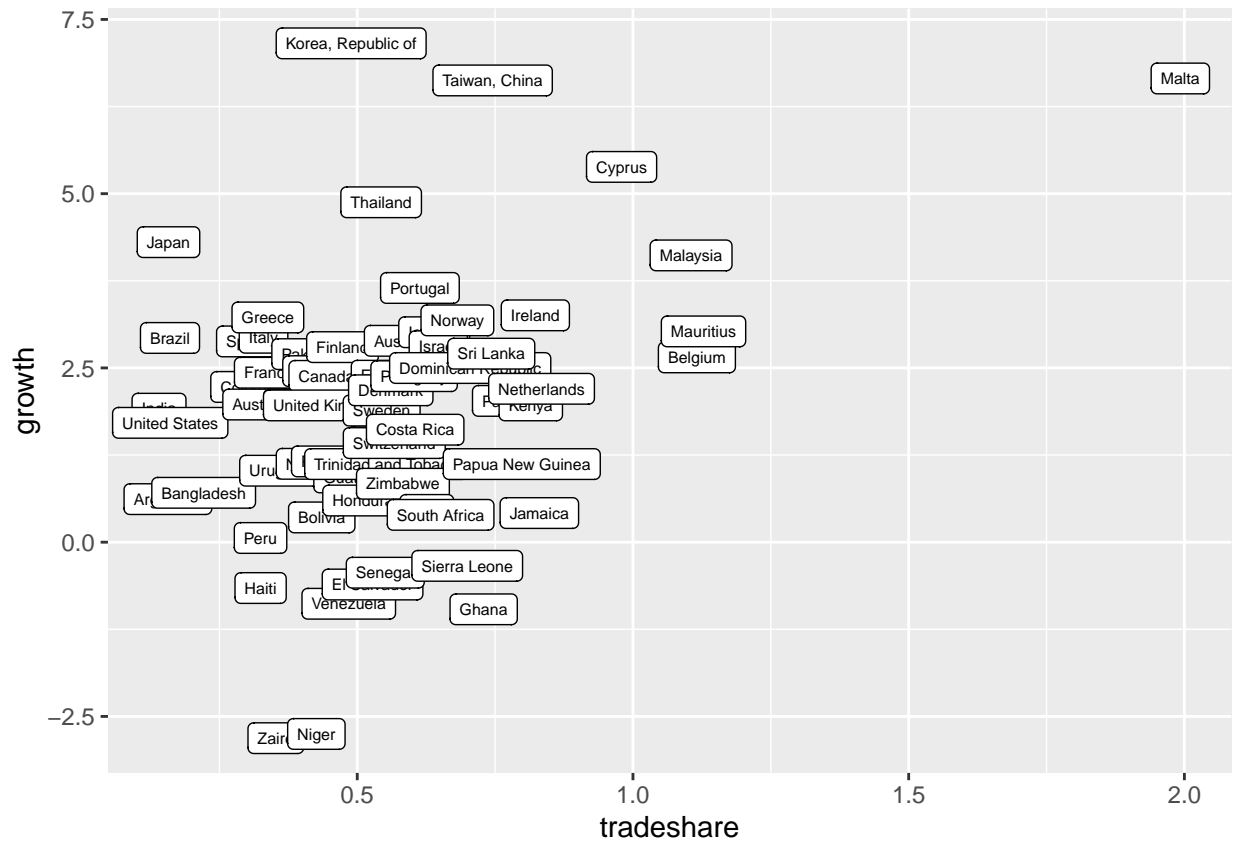
Detect the outlier: Print the country name + Tweak

```
ggplot(data=df, aes(x=tradeshare, y=growth, label=country)) +
  geom_point(col='blue', size=1) +
  geom_text(size=3)
```



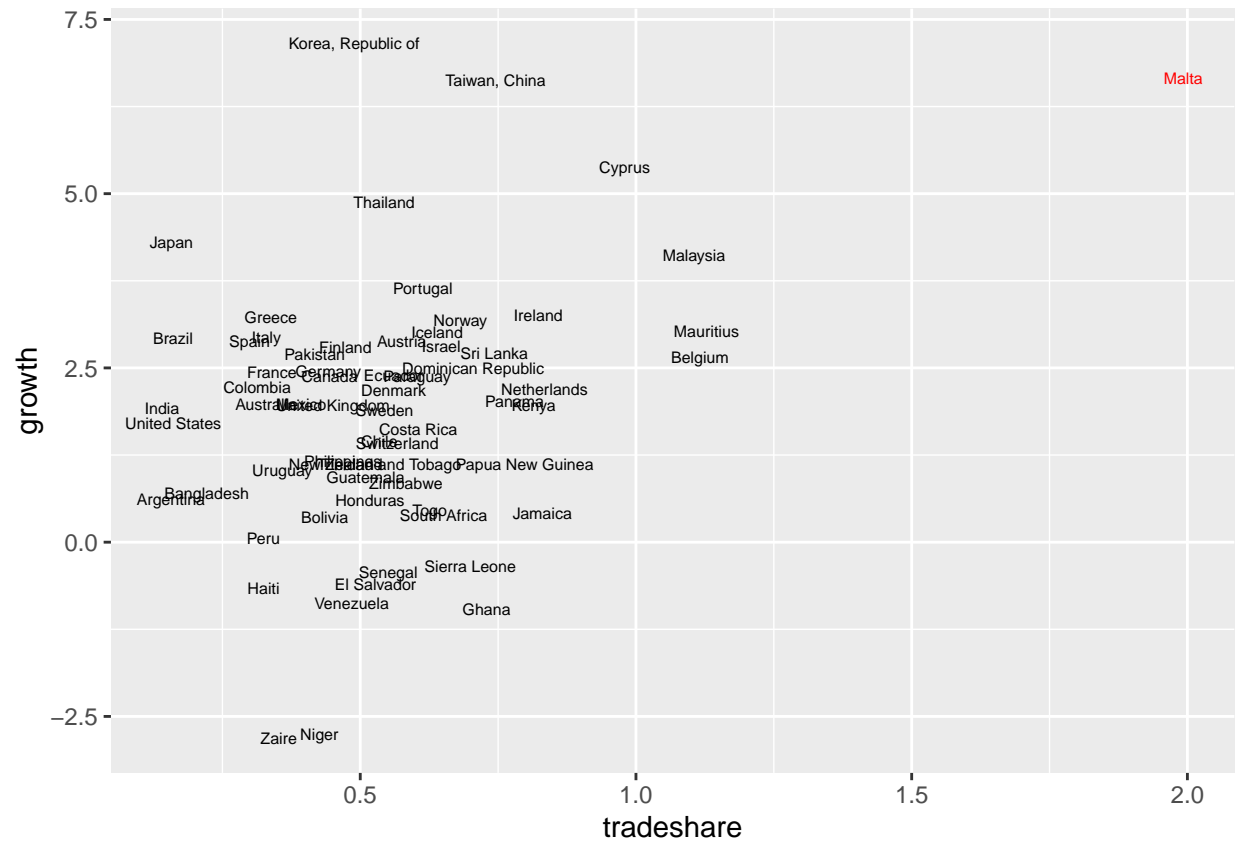
Detect the outlier: Use labels instead of plain text

```
ggplot(data=df, aes(x=tradeshare, y=growth, label=country)) +
  geom_label(size=2)
```



Detect the outlier: Highlight the variable name

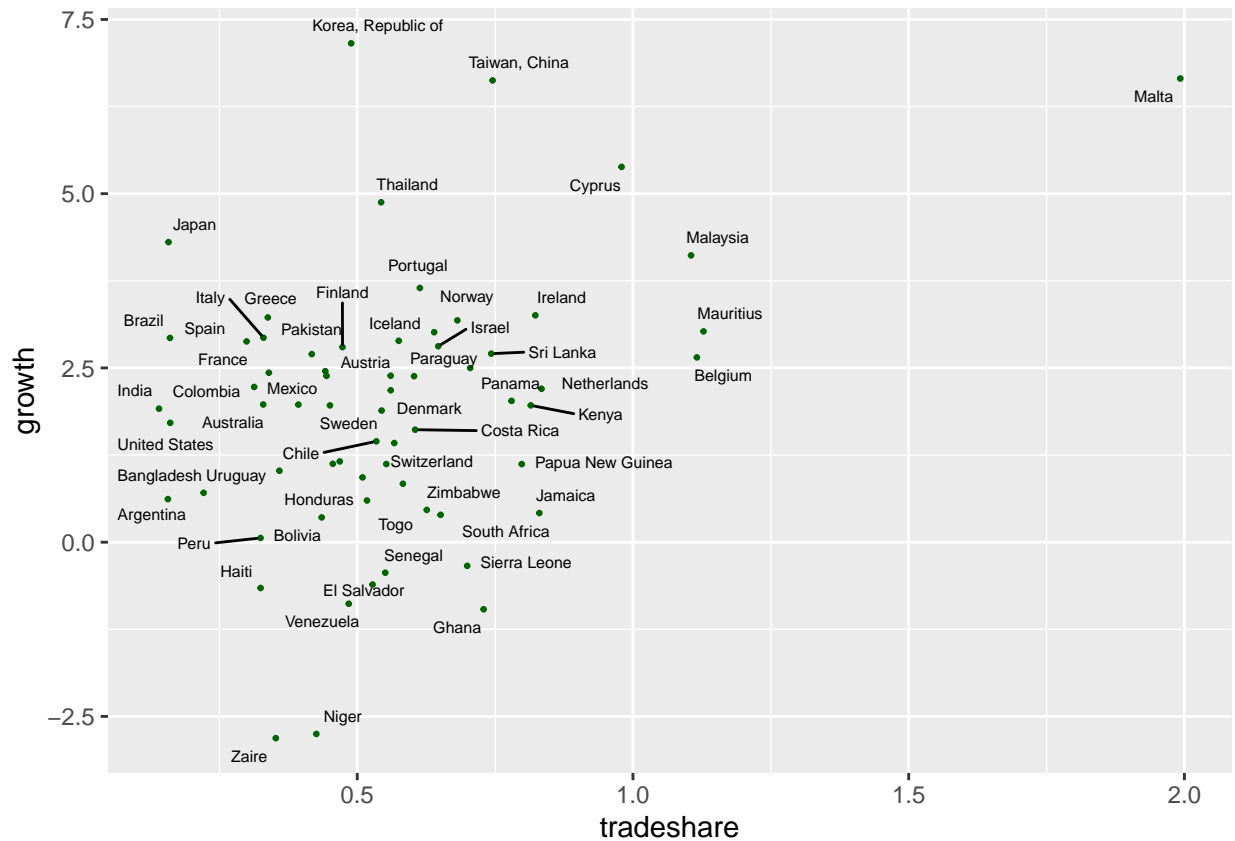
```
ggplot(data=df, aes(x=tradeshare, y=growth, label=country)) +
  geom_text(size=2, aes(colour = I(ifelse(country == "Malta", "red", "black"))))
```



Detect the outlier: Avoid overlapping labels

```
library(ggrepel)
ggplot(data=df, aes(x=tradeshare, y=growth, label=country)) +
  geom_point(col="darkgreen", size=0.5) +
  geom_text_repel(aes(label=country), size=2)
```

```
## Warning: ggrepel: 9 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps
```

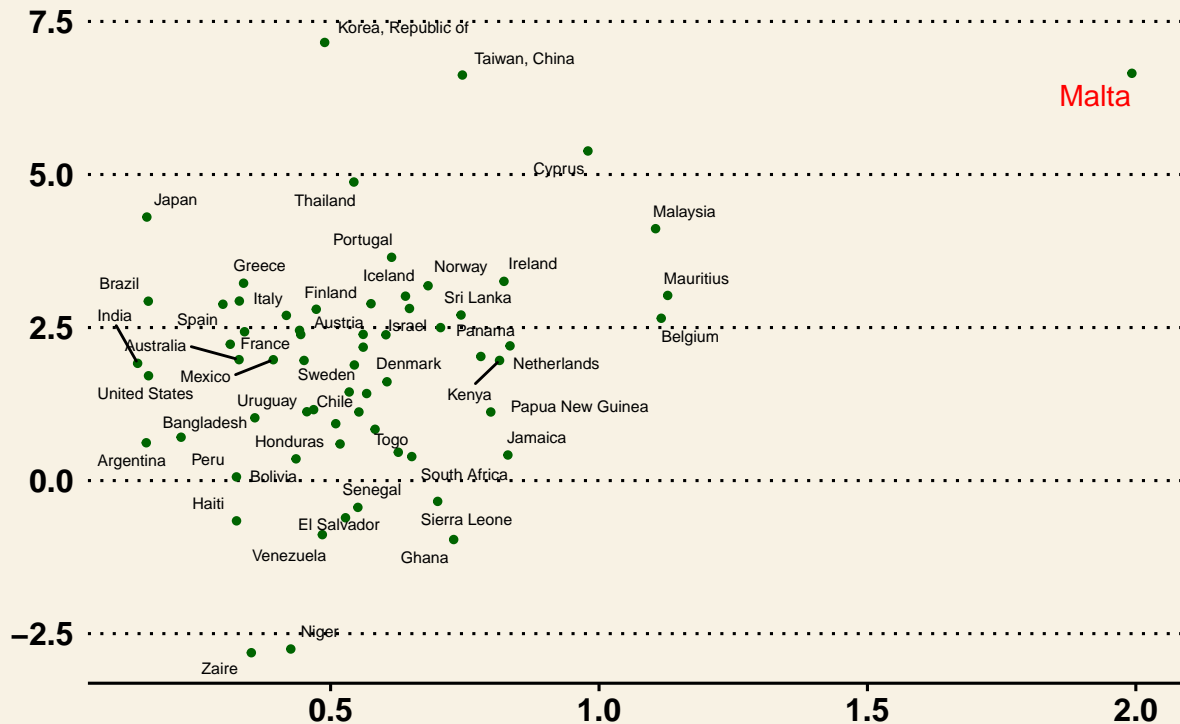


Detect the outlier: Add a theme!

```
ggplot(data=df, aes(x=tradeshare, y=growth, label=country)) +
  geom_point(col="darkgreen", size=1) +
  geom_text_repel(aes(label=country), colour=I(ifelse(df$country == "Malta", "red", "black")), size=I
ggtitle("growth vs trade") +
  theme_wsj()
```

```
## Warning: ggrepel: 15 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps
```

growth vs trade



Investigate correlation:

```
cor.test(df$growth, df$tradeshare)
```

```
##
## Pearson's product-moment correlation
##
## data: df$growth and df$tradeshare
## t = 2.98, df = 63, p-value = 0.0041
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## 0.11790 0.54853
## sample estimates:
## cor
## 0.35168
```

Investigate linear regression:

```
ols <- lm(growth ~ tradeshare, data=df)
summary(ols)
```

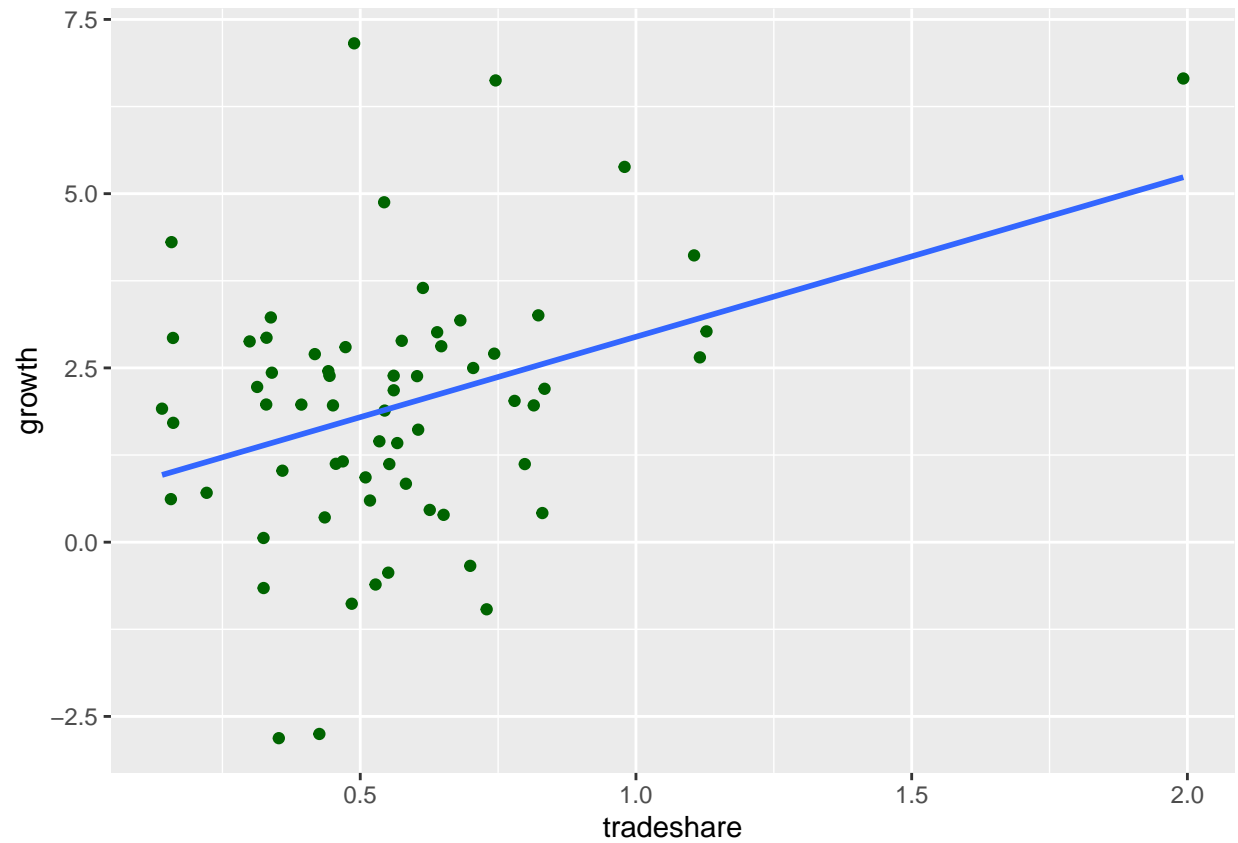


```
##
## Call:
## lm(formula = growth ~ tradeshare, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -4.374 -0.886  0.233  0.925  5.389
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.640      0.490    1.31  0.1961
## tradeshare      2.306      0.773    2.98  0.0041 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.79 on 63 degrees of freedom
## Multiple R-squared:  0.124, Adjusted R-squared:  0.11
## F-statistic: 8.89 on 1 and 63 DF, p-value: 0.00407
```

Add regression line to the scatterplot:

```
ggplot(data=df, aes(x=tradeshare, y=growth)) +
  geom_point(col="darkgreen") +
  geom_smooth(method = "lm", se=FALSE)
```

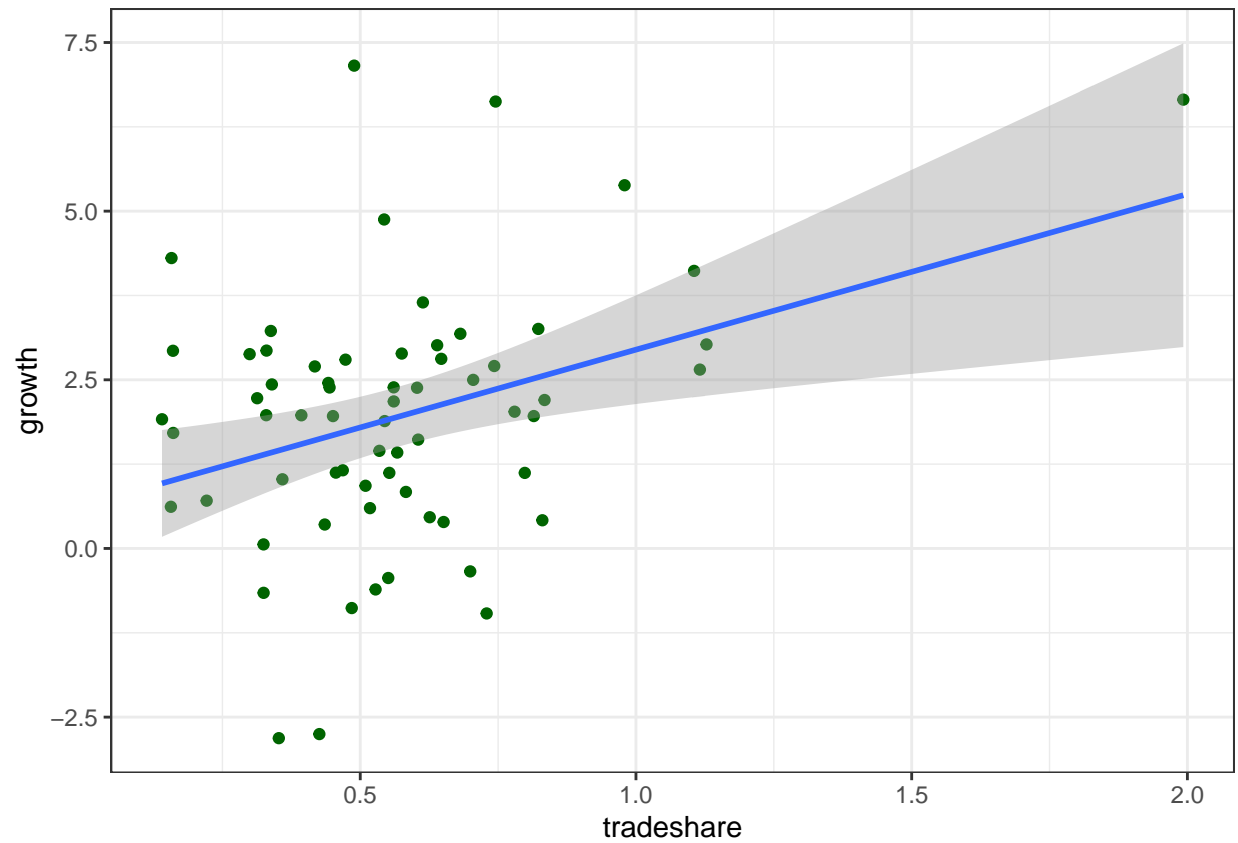
```
## 'geom_smooth()' using formula = 'y ~ x'
```



Add regression line to the scatterplot | confidence interval:

```
ggplot(data=df, aes(x=tradeshare, y=growth)) +  
  geom_point(col="darkgreen") +  
  geom_smooth(method = "lm", se=TRUE) +  
  theme_bw()
```

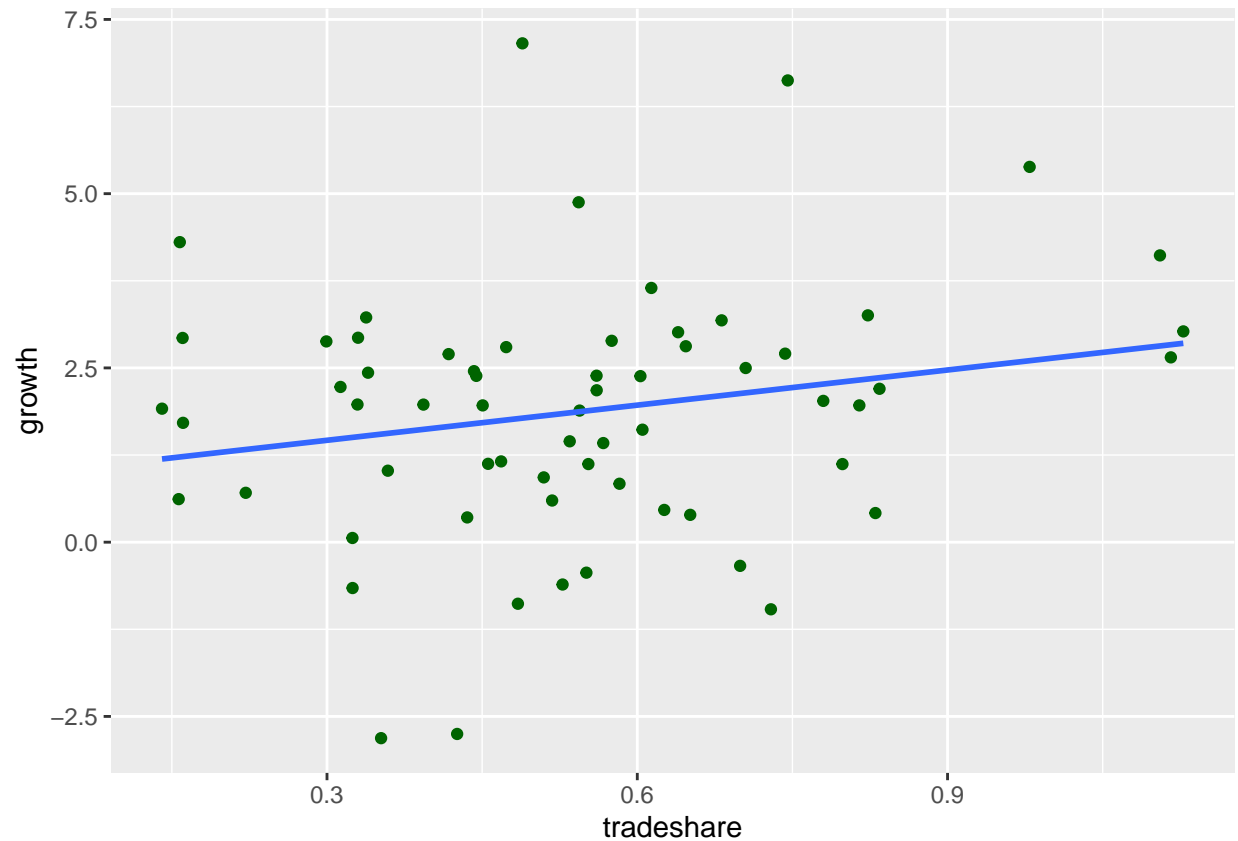
```
## 'geom_smooth()' using formula = 'y ~ x'
```



Regression without the outlier

```
df2 <- subset(df, country != "Malta")
ggplot(data=df2, aes(x=tradeshare, y=growth)) +
  geom_point(col="darkgreen") +
  geom_smooth(method = "lm", se=FALSE)
```

```
## 'geom_smooth()' using formula = 'y ~ x'
```

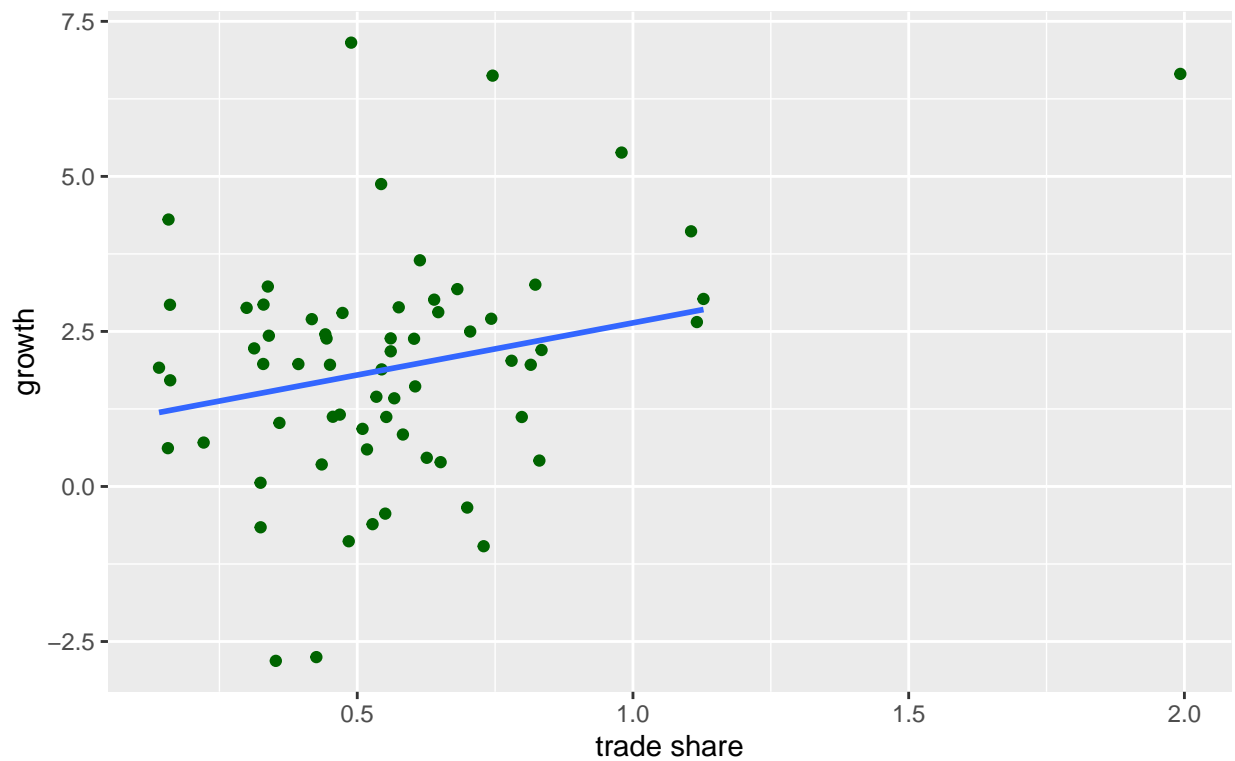


Regression without the outlier

```
ggplot(data=df, aes(x=tradeshare, y=growth)) +
  geom_point(col="darkgreen") +
  geom_smooth(method = "lm", se=FALSE, data=df2) +
  labs(title = 'OLS is not robust to regression',
        caption="Regression line with outlier omitted",
        x="trade share")
```

```
## 'geom_smooth()' using formula = 'y ~ x'
```

OLS is not robust to regression



Regression line with outlier omitted

Predict Malta

```
# get regression coefficients
ols2 <- lm(growth ~ tradeshare, data=df2)
b0 <- coef(ols2)[1]
b1 <- coef(ols2)[2]
# get Malta trade share
x_obs <- df[df$country == "Malta", "tradeshare"]
y_hat <- b0 + b1 * x_obs
y_hat
```

```
## tradeshare
## 1 4.3068
```

```
# or use the built-in predict:
predict(ols2, newdata=x_obs)
```

```
## 1
## 4.3068
```

Prediction with/without Malta in sample, compared

```
# observed value:
y_obs <- df[df$country == "Malta", "growth"]
y_obs
```

```
## # A tibble: 1 x 1
##   growth
##   <dbl>
## 1    6.65
```

```
# predicted value with outlier:
predict(ols, newdata=x_obs)
```

```
##      1
## 5.2361
```

```
# predicted value without outlier:
predict(ols2, newdata=x_obs)
```

```
##      1
## 4.3068
```