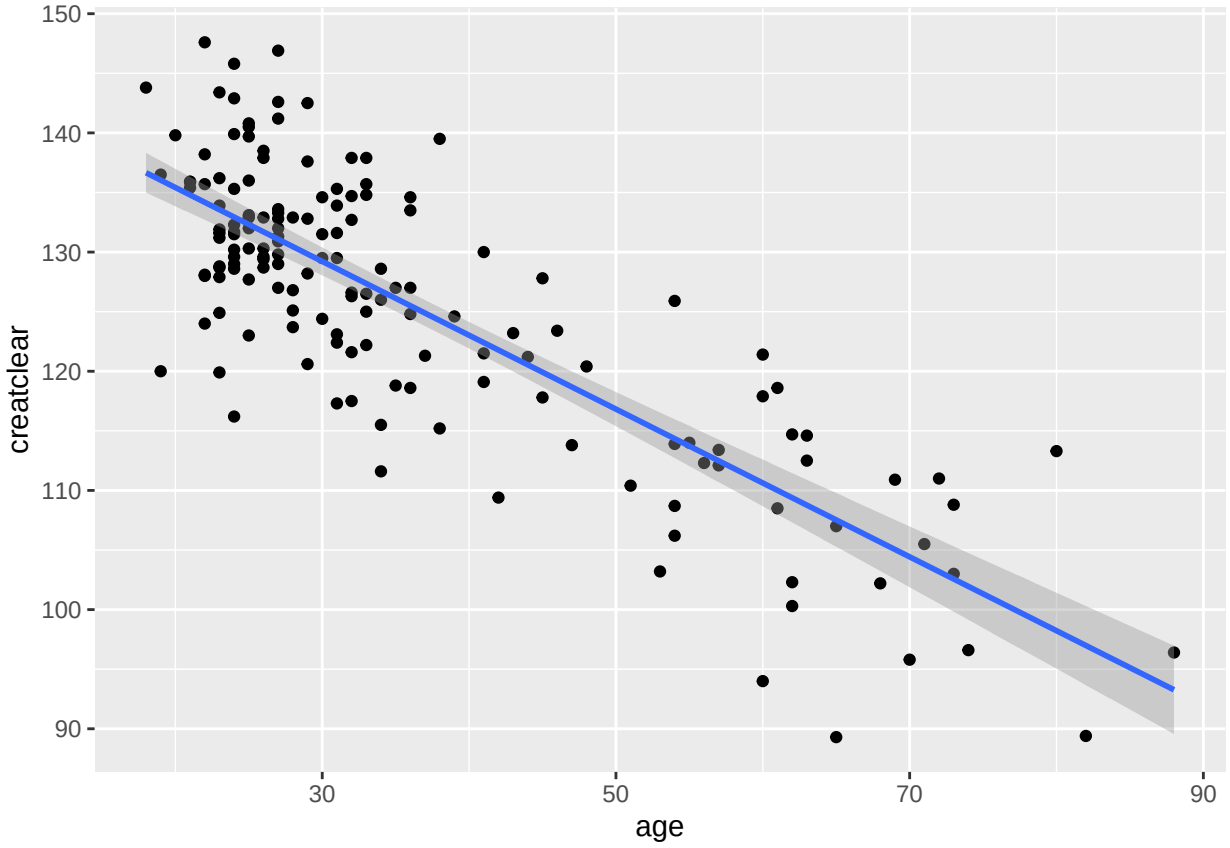
HW3

Kelly Kim

2024-02-05

```
## Rows: 157 Columns: 2
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (2): age, creatclear
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v purrr     1.0.2
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.4.4     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
## Registered S3 method overwritten by 'mosaic':
##   method                           from
##   fortify.SpatialPolygonsDataFrame ggplot2
##
##
## The 'mosaic' package masks several functions from core packages in order to add
## additional features.  The original behavior of these functions should not be affected by this.
##
##
## Attaching package: 'mosaic'
##
##
## The following object is masked from 'package:Matrix':
##
##     mean
##
##
## The following objects are masked from 'package:dplyr':
##
##     count, do, tally
##
##
## The following object is masked from 'package:purrr':
##
##     cross
##
```

```
##
## The following object is masked from 'package:ggplot2':
##
##     stat
##
##
## The following objects are masked from 'package:stats':
##
##     binom.test, cor, cor.test, cov, fivenum, IQR, median, prop.test,
##     quantile, sd, t.test, var
##
##
## The following objects are masked from 'package:base':
##
##     max, mean, min, prod, range, sample, sum
##
##
## `geom_smooth()` using formula = 'y ~ x'
```



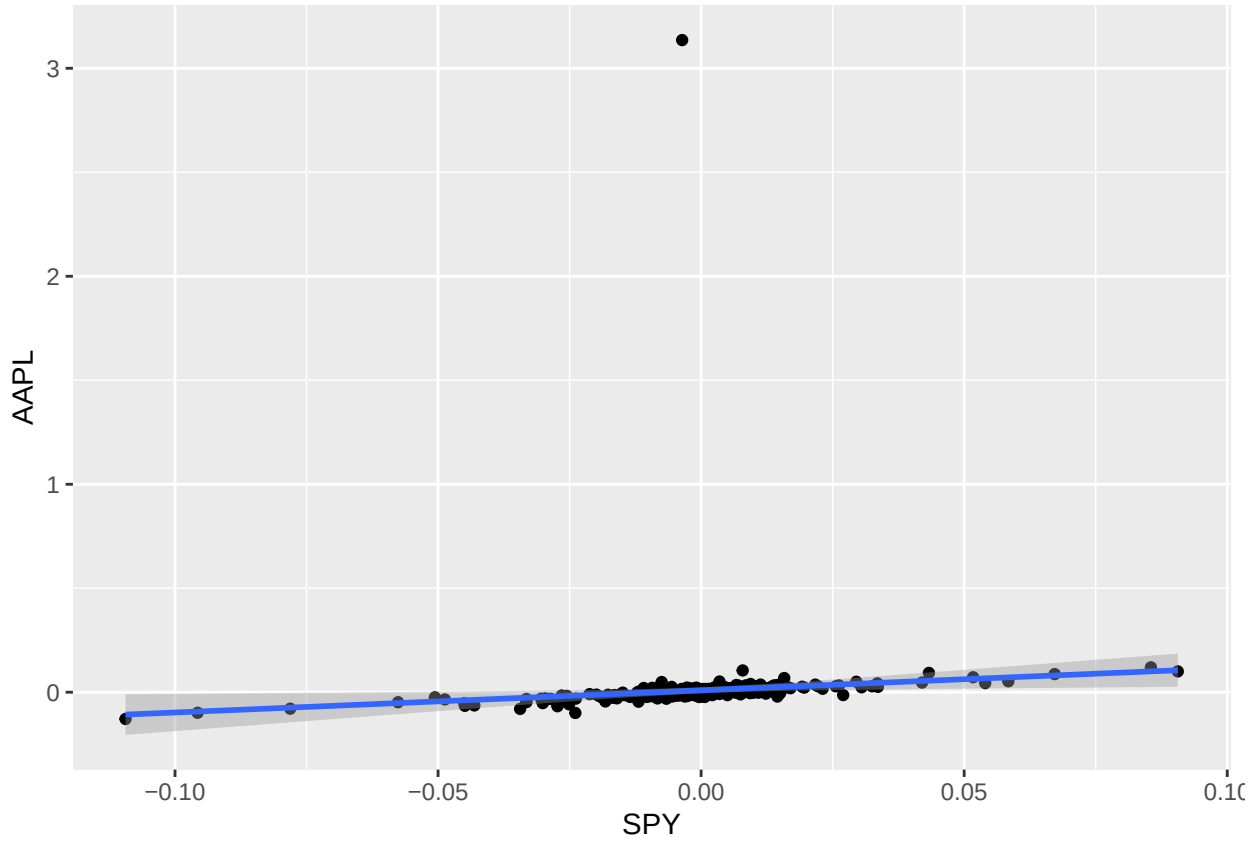

```
## (Intercept)         age
## 147.8129158  -0.6198159
```

creatclear = -0.6198(age) + 147.8129 age = 55 creatclear = -0.6198 * 55 + 147.8129 = 113.7239 a) I drew a linear regression line to predict creatine clearance rate of a 55-year-old patient and plugged in 55 in the equation. creatclear = -0.6198 * 55 + 147.8129 = 113.7239 The predicted creatine clearance rate of a 55-year-old patient is 113.7239mL/minute.

b) Creatinine clearance rate changes -0.6198ml/minute per year. I drew a linear regression line where x = age and y = clearance rate and the slope indicates how the clearance rate changes in average as the patient gets 1 year older. And the slope is -0.6198.

c)age = 40 -0.6198 * 40 + 147.8129 = 123.0209 135 age = 60 -0.6198 * 60 + 147.8129 = 110.6249 112 The patient whose age is 40 years old is healthier for his/her age since the residual for him/her is 11.9791(135 - 123.0209) while the residual for the patient at 60 year-old is 1.3751 (112 - 110.6249). The residual of the 40-year-old patient is much larger than that of the 60-year-old patient, meaning that the 40-year-old patient has higher creatinine rate at his/her age than that of the 60-year-old patient.

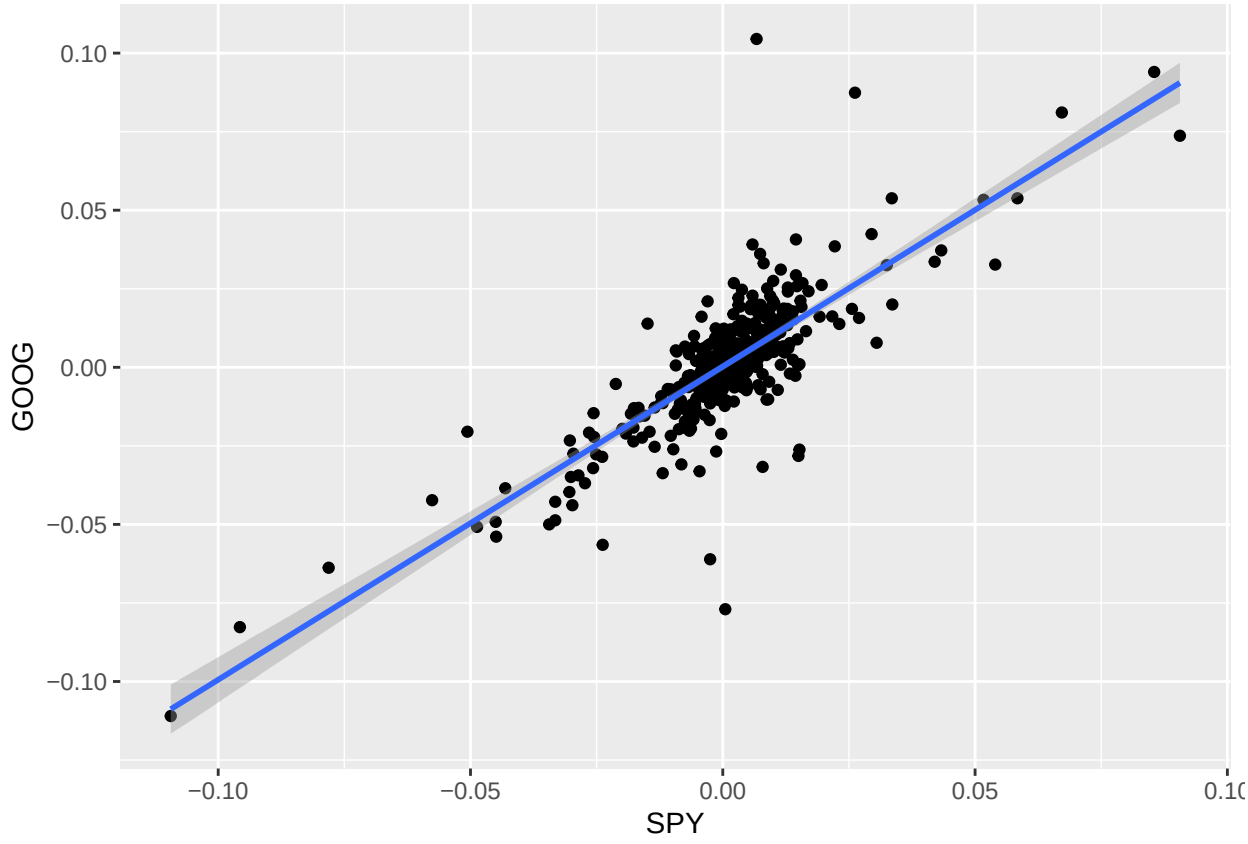
```
## Rows: 424 Columns: 8
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl  (7): SPY, AAPL, GOOG, MRK, JNJ, WMT, TGT
## date (1): Date
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
## `geom_smooth()` using formula = 'y ~ x'
```




```
## (Intercept)         SPY
## 0.009189277 1.065601182
```

```
## [1] 0.01336246
```

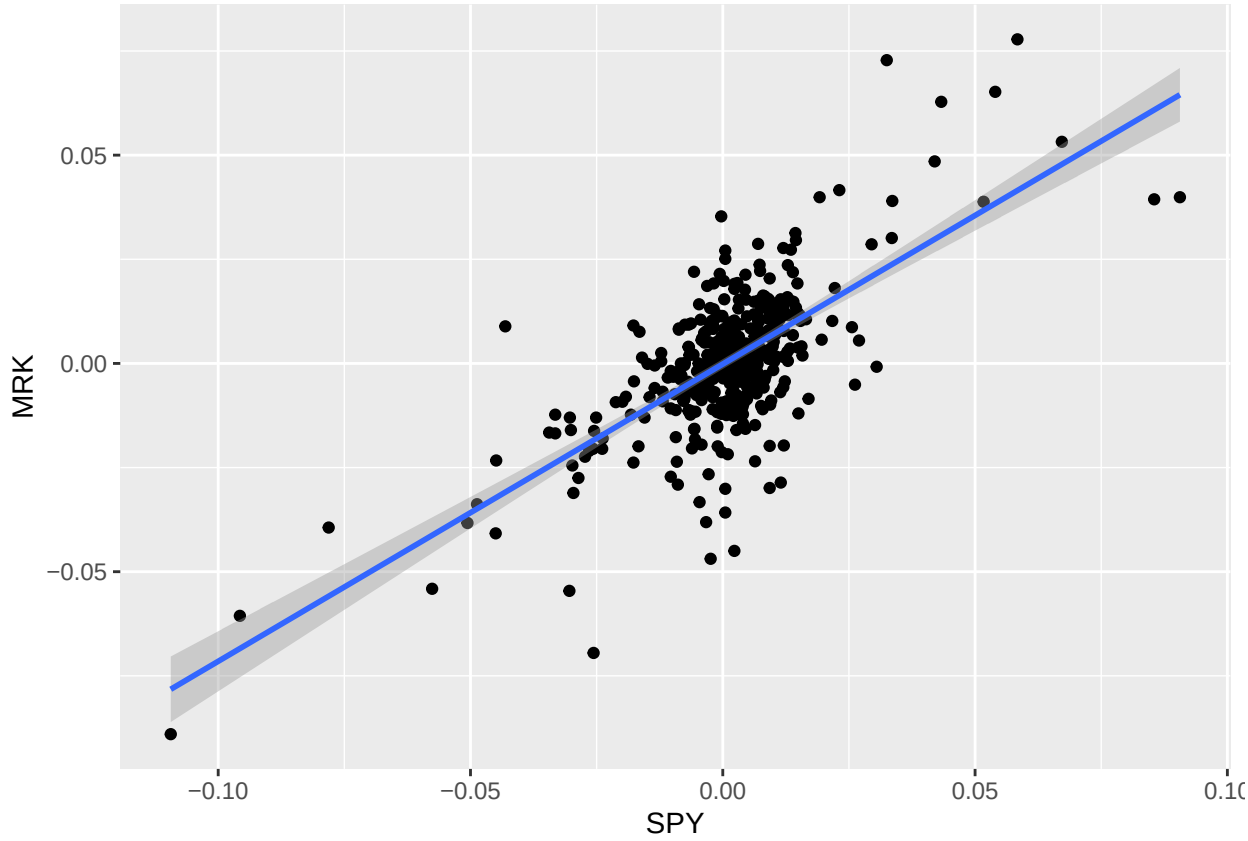
```
## `geom_smooth()` using formula = 'y ~ x'
```




```
##  (Intercept)          SPY
## 0.0002330467 0.9967745749
```

```
## [1] 0.6483015
```

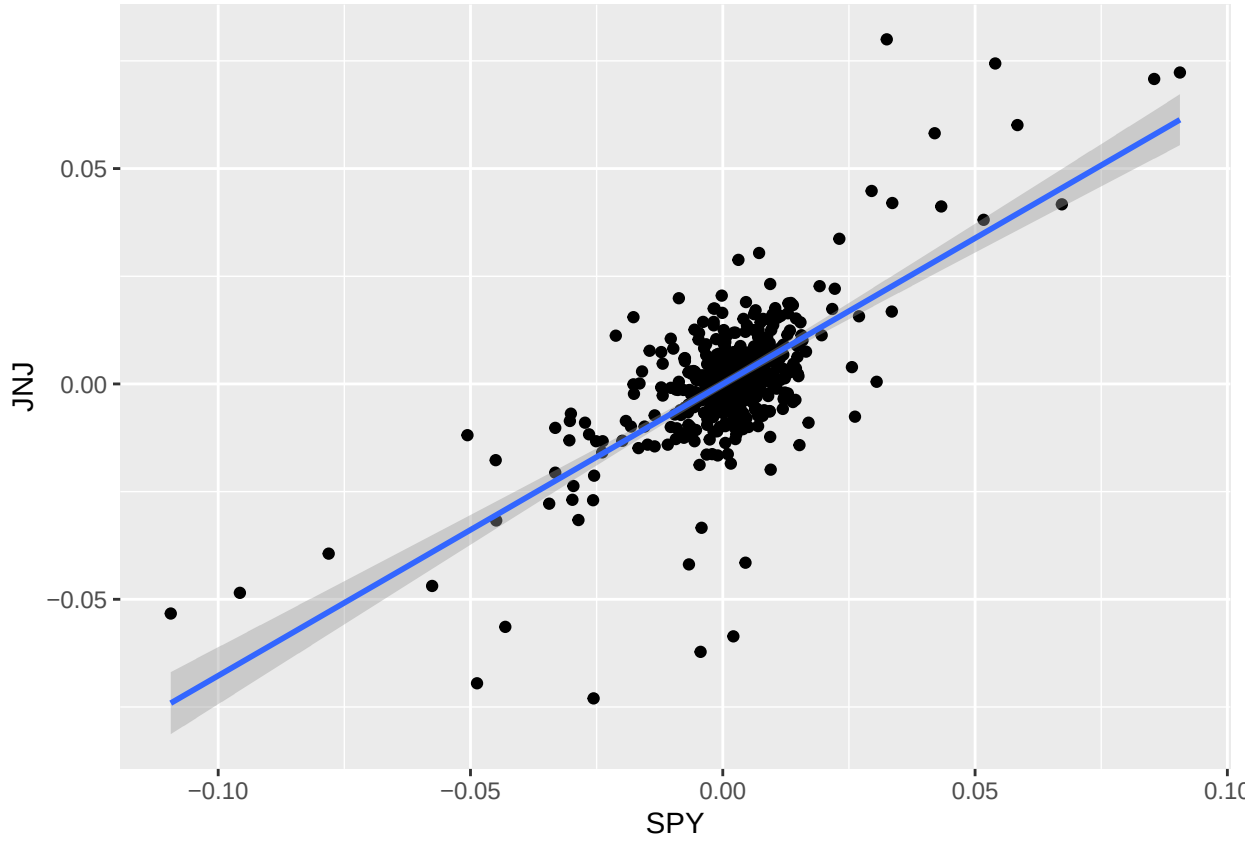
```
## `geom_smooth()` using formula = 'y ~ x'
```




```
##   (Intercept)           SPY
## -0.0001540208  0.7136140905
```

```
## [1] 0.483701
```

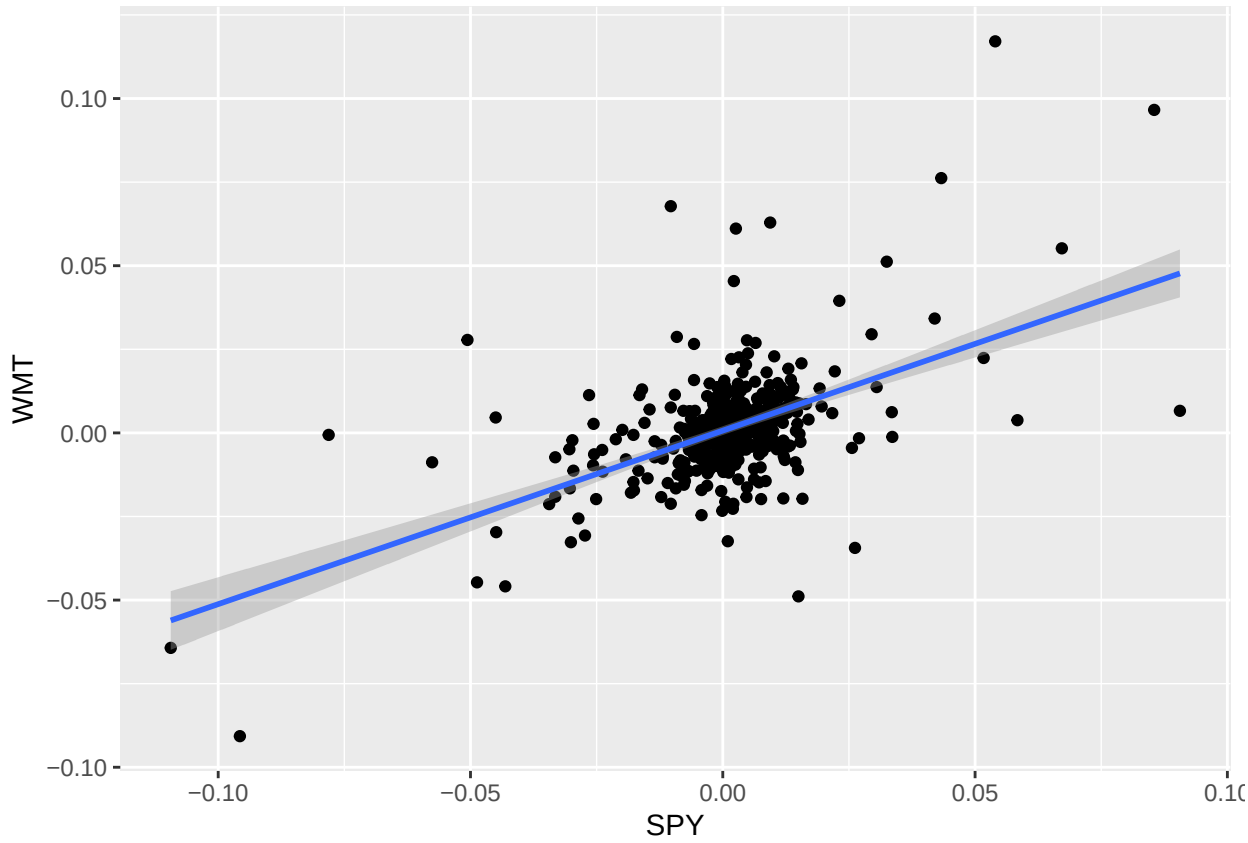
```
## `geom_smooth()` using formula = 'y ~ x'
```




```
##   (Intercept)           SPY
## -2.410714e-05  6.771930e-01
```

```
## [1] 0.501943
```

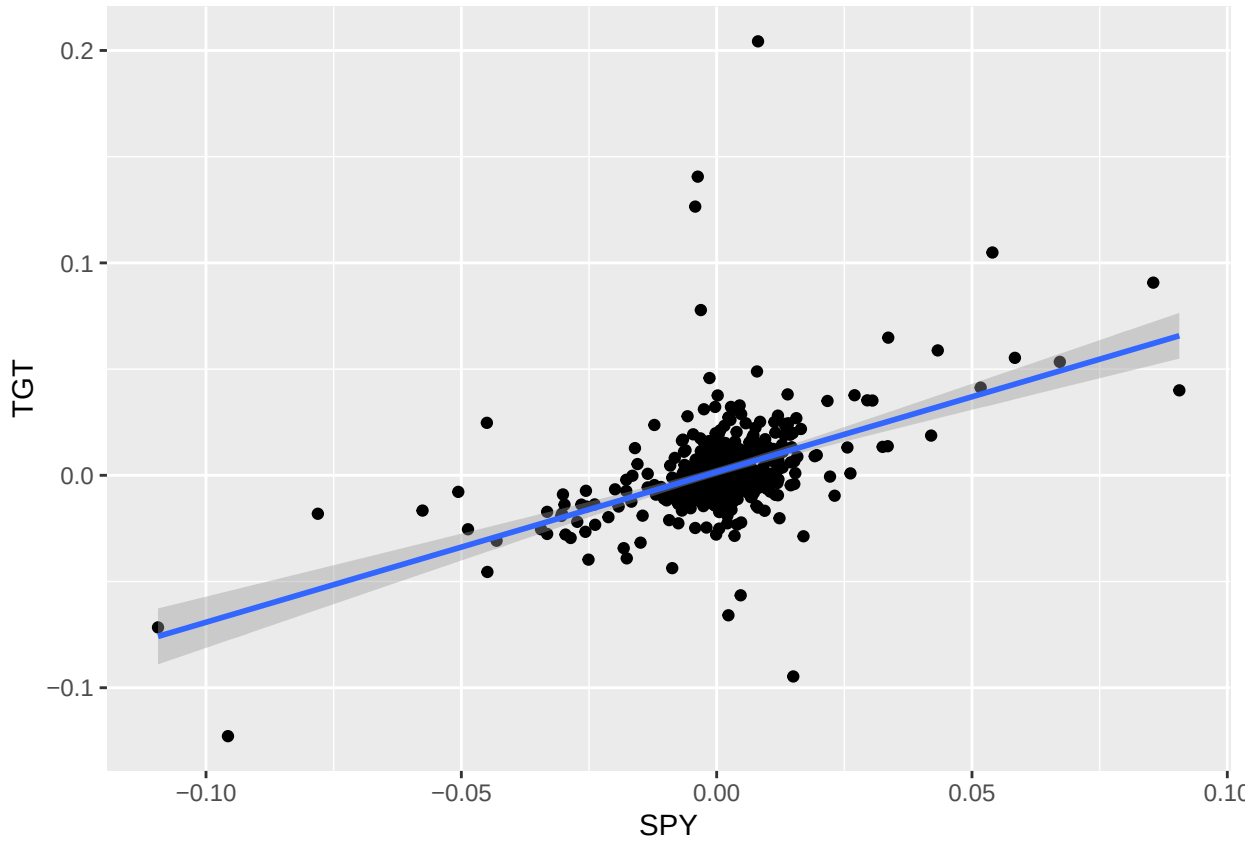
```
## `geom_smooth()` using formula = 'y ~ x'
```




```
##  (Intercept)          SPY
## 0.0006781104 0.5189810644
```

```
## [1] 0.2853233
```

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




```
## (Intercept)         SPY
## 0.001583341 0.707648462
```

```
## [1] 0.2478813
```

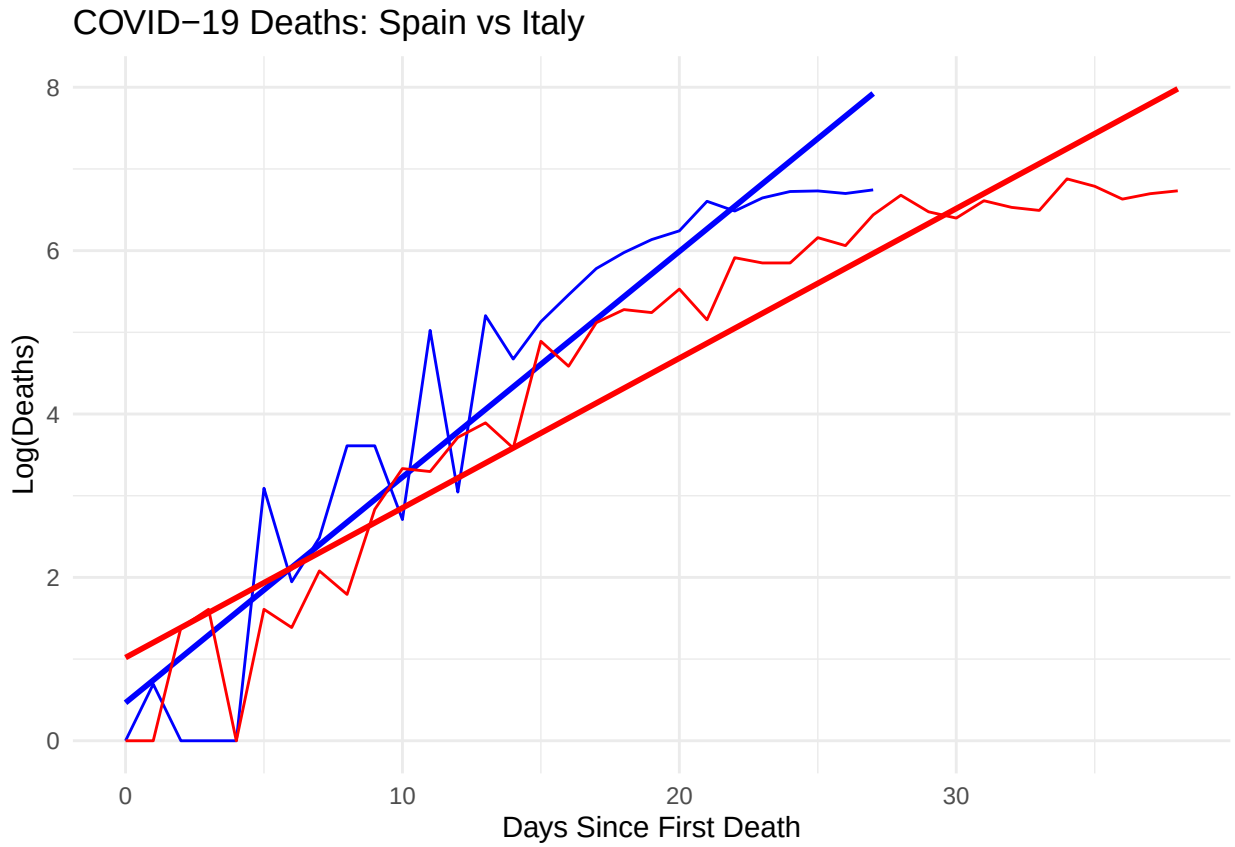
```
## # A tibble: 6 x 4
##   Ticker  Intercept Slope R_squared
##   <chr>       <dbl> <dbl>     <dbl>
## 1 AAPL    0.00919   1.07     0.0134
## 2 GOOG    0.000233  0.997    0.648
## 3 MRK    -0.000154  0.714    0.484
## 4 JNJ    -0.0000241 0.677    0.502
## 5 WMT     0.000678  0.519    0.285
## 6 TGT     0.00158   0.708    0.248
```

Beta is a systematic risk that a company takes, showing how much of percentage of the stock return changes as market portfolio changes by 1%. Beta is calculated as the slope of the regression line, measuring how much the stock's returns change for a unit change in the market returns. A linear regression analysis where the dependent variable is the rate of return of the individual stock and the independent variable is the rate of return of the market(market portfolio increasing by 1%) is needed.

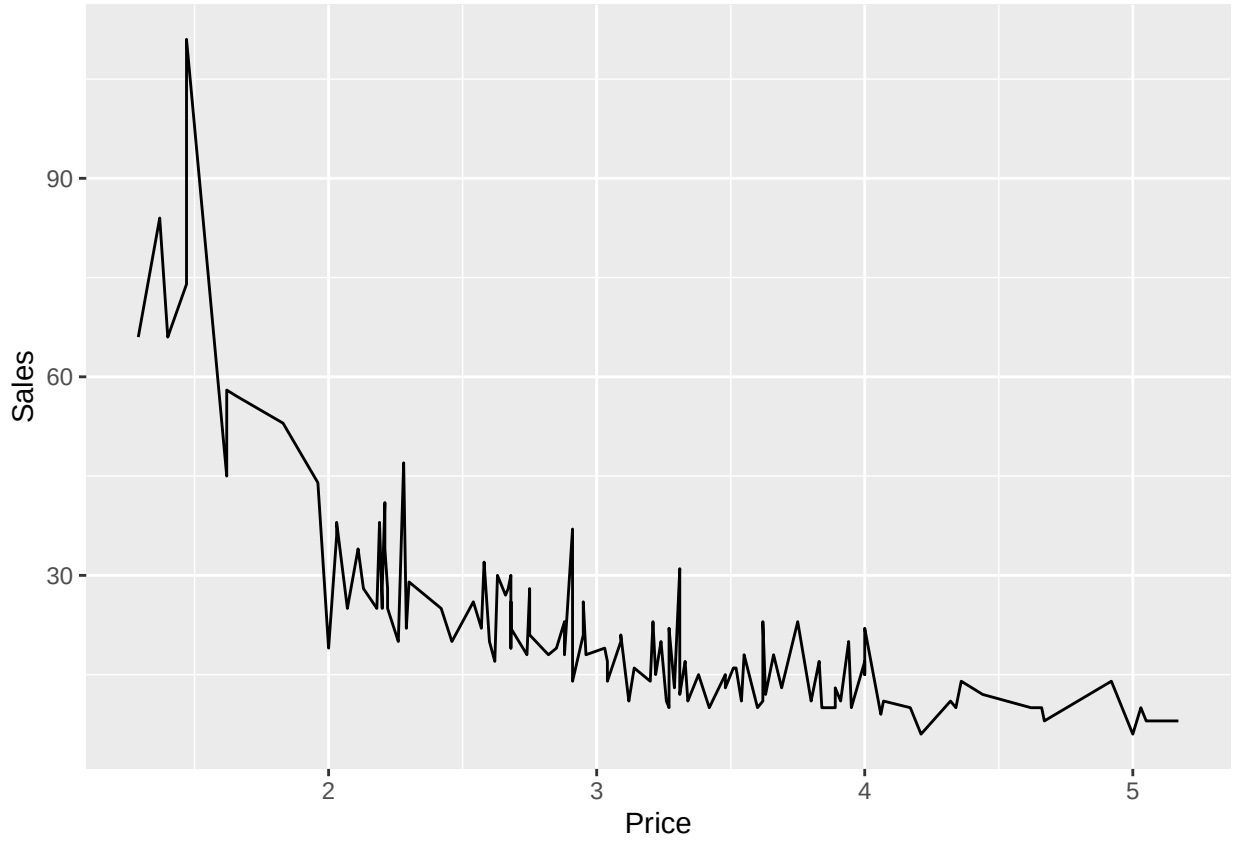
The slope variable indicates how much the stock's returns change as market portfolio increases by 1%. R^2 represents the proportion of variance in the stock's returns explained by changes in the market returns, with higher values indicating better fit. And the intercept represents the expected return of the stock when the market return is zero(no change in market portfolio).

AAPL has the highest systematic risk whereas WMT has the lowest systematic risk.

```
## Rows: 67 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (1): country
## dbl  (2): deaths, days_since_first_death
## date (1): date
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.

## days_since_first_death
##                     4

## days_since_first_death
##                     4

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



An estimated growth rate for Italy is 0.183, and an estimated growth rate for Spain is 1.276. The doubling time for Italy is 4 and the doubling time for Spain is 3.




```
## (Intercept)       price
##    65.78980   -13.78284
```

```
## [1] 3.102069
```

```
## [1] 23.03448
```

The price elasticity for milk is calculated by using the formula: coefficient of price * (average price/average sales). As the slope of the regression model where the sales is the dependent variable and the price is the independent variable is -13.78, the average price is 3.102069 and the average sales is 23.03448, the formula would be (-13.78) * (3.102069/23.03448) = -1.856, indicating that the estimated price elasticity of demand for milk is -1.856. People tend to buy less milk as the price of milk increases.