Econometrics: Methods and Applications

Diego López Tamayo * Based on MOOC by Erasmus University Rotterdam

Contents

"There are two things you are better off not watching in the making: sausages and econometric estimates." -Edward Leamer

*El Colegio de México, diego.lopez@colmex.mx

Requirements

- You need some basic background in statistics and matrices.
- You can use any statistical package that is available to you, for example packages like R, Stata, EViews and other. The main requirement is that you can run regressions to get coefficients and standard errors.

Introduction

The following notes and code chunks are made in R statistical package, you can also found the Do-File for Stata 16 in the download sections of my website. Both files follow the same structure and use the same data sets.

All the data sets are downloadable from my Github repository

In the following notes we will cover: **simple regression, multiple regression, model specification, endogeneity, binary choice, and time series.**

For example:

Suppose you wish to predict the number of airplane passengers worldwide for next year.

- In **simple regression**, you use a single factor to explain airplane passenger traffic, for example, worldwide economic growth.
- In **multiple regression**, you use additional explanatory factors, such as the oil price, the price of tickets, and airport taxes.
- **Model specification** answers the question which factors to incorporate in the model, and in which way.
- **Endogeneity** is concerned with possible reverse causality. For example, if economic growth does not only lead to more air traffic, but reversely, increased air traffic also influences economic growth.
- **Binary choice** considers the micro level of individual decisions whether or not to travel by plane, in terms of factors like family income and the price of tickets.
- In **time series** analysis, you analyze trends and cycles in airplane passenger traffic in previous years, to predict future developments.

Building Blocks

Required background on matrices, probability and statitics:

Matrices Recommended: S.Grossman. *Elementary Linear Algebra*

- Matrix summation, matrix multiplication
- Square matrix, diagonal matrix, identity matrix, unit vector
- Transpose, trace, rank, inverse
- Positive and negative (semi)definite matrix
- Gradient vector, Hessian matrix
- First and Second Order Conditions for optimization of vector functions

Probability Recommended: Casella & Berger. *Statistical Inference*

- Univariate and multivariate random variables
- Probability density function (pdf)
- Cumulative density function (cdf)
- Expectation, expectation of functions
- Mean, variance, standard deviation
- Covariance, correlation
- Mean, variance, and covariance of linear transformations
- Independence
- Higher order moments, skewness, kurtosis

- Normal distribution, standard normal distribution
- Multivariate normal distribution
- Linear transformations of normally distributed random variables
- Chi-squared distribution, Student t-distribution, F-distribution

Statistics Recommended: J. Wooldridge *Introductory Econometrics: A Modern Approach*

- Statistic, estimator, estimate
- Standard error
- Confidence interval
- Unbiasedness
- Efficiency
- Consistency
- Sample mean, sample variance
- Hypothesis, null and alternative hypothesis
- Test statistic
- Type I and Type II error
- Size and power of a statistical test
- Significance level
- Critical value, critical region
- P-value
- T-statistic, Chi-squared statistic, F-statistic

Parameter Estimation

Suppose you have 26 observations of the yearly return on the stock market. We call a set of observations a sample. Bellow, you will see a histogram of the sample. The returns in percentages are on the x-axis and the y-axis gives the frequency. The sample mean equals 9.6%. What can we learn from this sample mean about the mean of the return distribution over longer periods of time? Can we be sure that the true mean is larger than zero?

Dataset S1

Contains 26 yearly returns based on the S&P500 index. Returns are constructed from end-of-year prices Pt as rt = (Pt − Pt-1)/Pt-1. Data has been taken from the public FRED database of the Federal Reserve Bank of St. Louis.

```
dataset_s1 <- read_csv(
  "https://raw.githubusercontent.com/diego-eco/diego-eco.github.io/master/downloads/dataset_s1.csv")
```

A simple stat description of our dataset:

```
summary(dataset_s1$Return)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -0.384858  0.007479  0.125918  0.096418  0.255936  0.341106
```

```
# mean,median,25th and 75th quartiles,min,max
```

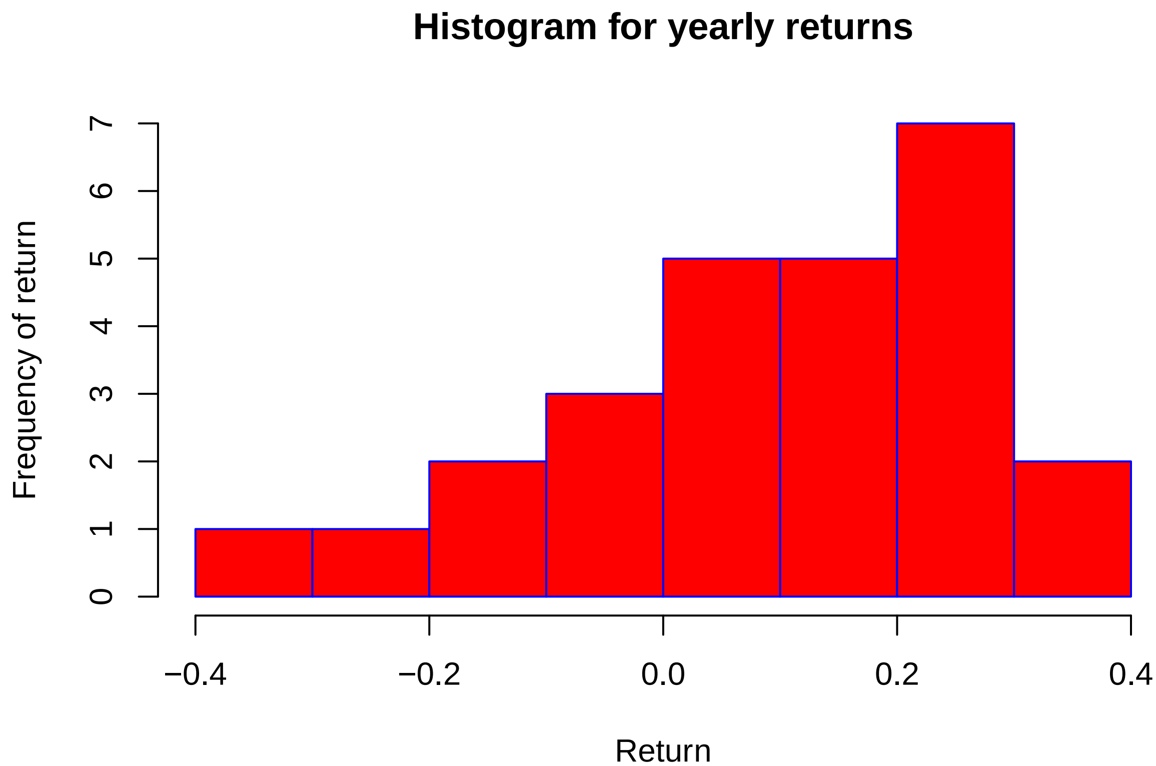
```
Hmisc::describe(dataset_s1$Return)
```

```
## dataset_s1$Return
##         n   missing  distinct      Info      Mean       Gmd       .05       .10
##        26         0        26         1   0.09642    0.2008 -0.207851 -0.115909
##       .25       .50       .75       .90       .95
##  0.007479  0.125918  0.255936  0.284259  0.306564
##
## lowest : -0.3848579 -0.2336597 -0.1304269 -0.1013919 -0.0655914
## highest:  0.2666859  0.2725047  0.2960125  0.3100818  0.3411065
```

```r
# n, nmiss, unique, mean, 5,10,25,50,75,90,95th percentiles
# 5 lowest and 5 highest scores
```

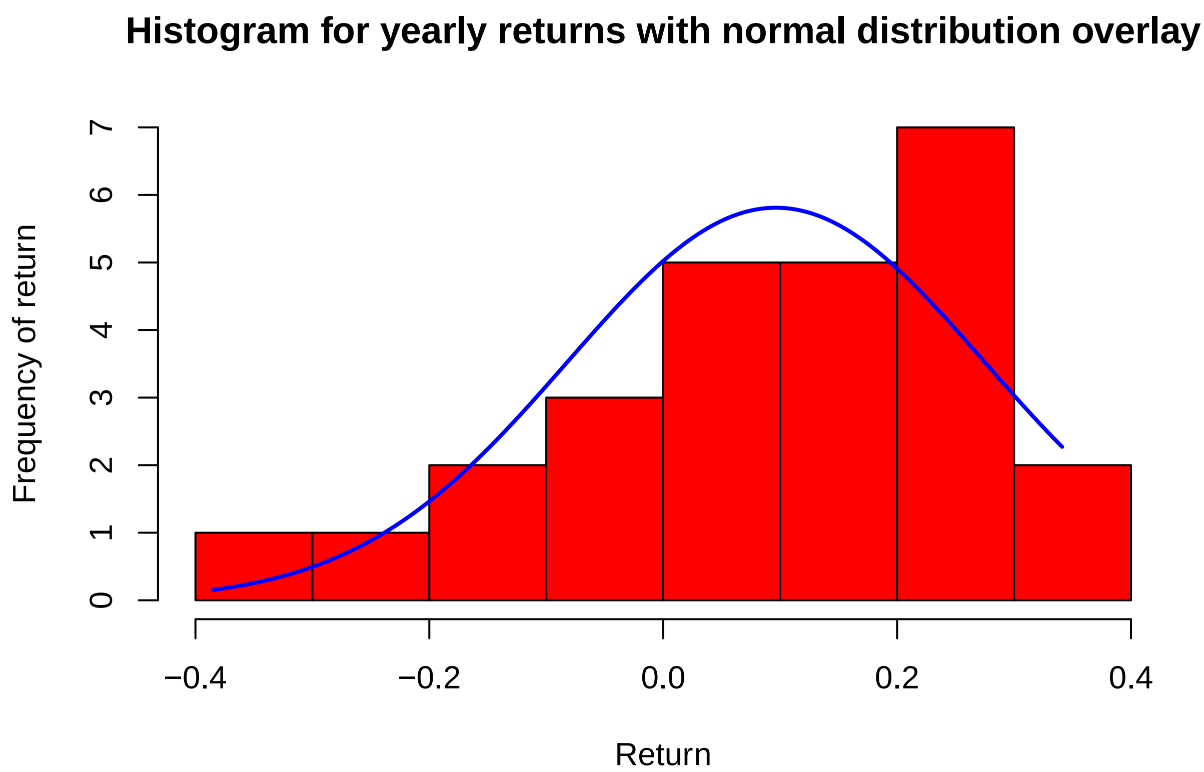
An histogram of the yearly returns on S&P500 index:

```r
hist(dataset_s1$Return, main="Histogram for yearly returns",
     xlab="Return", ylab="Frequency of return",
     border="blue", col="red")
```



Let's add a Normal denisty curve on top of the distribution:

```r
plotNormalHistogram(dataset_s1$Return, prob = FALSE, col = "red",
                    main = "Histogram for yearly returns with normal distribution overlay",
                    xlab="Return", ylab="Frequency of return",
                    linecol = "blue", lwd = 2)
```



Dataset Training Exercise S1 Uses 1000 simulated values from a normal distribution (mean 0.06, standard deviation 0.015).

```
trainexers_s1 <- read_csv(
  "https://raw.githubusercontent.com/diego-eco/diego-eco.github.io/master/downloads/trainexers1.csv")
```

You want to investigate the precision of the estimates of the mean return on the stock market. You have a simulated sample of 1000 yearly return observations $y_i \sim NID(\mu, \sigma^2)$.

1. Construct a series of mean estimates m_i, where you use the first i observations, so $m_i = \frac{1}{i}\sum_{j=1}^{i} y_j$. Calculate the standard error for each estimate m_i. Make a graph of m_i and its 95% confidence interval, using the rule of thumb of 2 standar deviations.
2. Suppose that the standard deviation of the returns equals 15%. How many years of observations would you need to get the 95% confidence interval smaller than 1%?

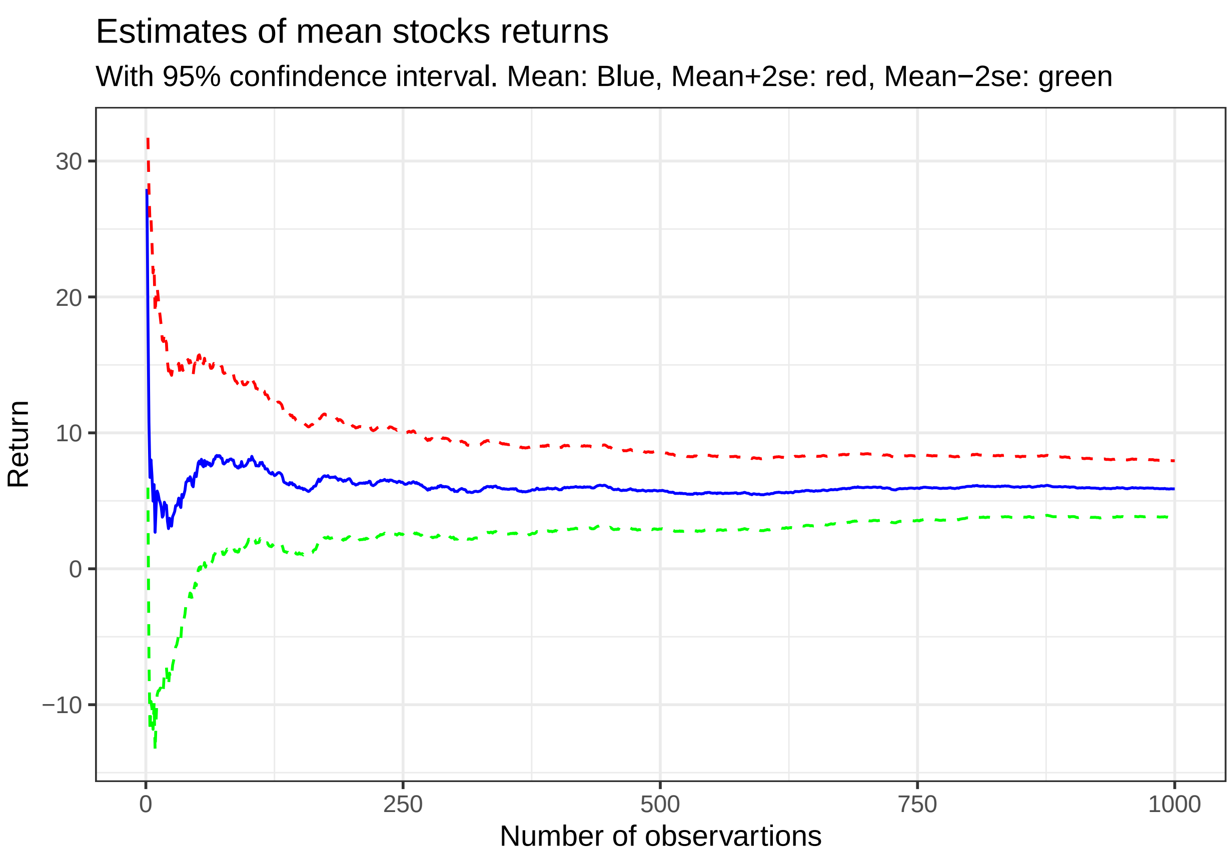
We know $se = \frac{\sigma}{\sqrt{n}}$. Solving $4\frac{\sigma}{\sqrt{n}} = 1 \Rightarrow n = 16\sigma^2$ therefore if $\sigma = 15\%$ yields $16(15^2) = 3,600$ years.

The Standard Error is $SE_i = \sqrt{var(m_i)} = \sqrt{\frac{1}{i-1}\sum_{j=1}^{i}(y_j - m_i)^2}$

```
# We create a new collumn for our estimates
trainexers_s1 <- trainexers_s1 %>% mutate(estimates=0)
# We add to each row the estimate with a for loop
for (i in 1:length(trainexers_s1$Return)){
  trainexers_s1[i,3]=(1/i)*(sum(trainexers_s1[1:i,2]))
}

# We create a new collumn for our standard errors
trainexers_s1 <- trainexers_s1 %>% mutate(std_errors=0)
# We add to each row the standard error with a for loop
for (i in 1:length(trainexers_s1$Return)){
  trainexers_s1[i,4]=sqrt(var(trainexers_s1[1:i,3]))
```

```r
}

# We create the +- 2 Standar Errors
trainexers_s1 <- trainexers_s1 %>% mutate(plus2se=0,minus2se=0)
# We fill the rows with a for loop
for (i in 1:length(trainexers_s1$Return)){
  trainexers_s1[i,5] = trainexers_s1[i,3]+2*trainexers_s1[i,4]
  trainexers_s1[i,6] = trainexers_s1[i,3]-2*trainexers_s1[i,4]
}
```

```r
# We create the graph
plot1 <- ggplot(data=trainexers_s1, aes(x=Observation)) +
  geom_line(aes(y = estimates,color='Mean'), color = "blue") +
  geom_line(aes(y = plus2se), color="red", linetype="dashed") +
  geom_line(aes(y = minus2se), color="green", linetype="dashed") +
  labs(title="Estimates of mean stocks returns  ",
       subtitle="With 95% confindence interval. Mean: Blue, Mean+2se: red, Mean-2se: green", y="Return"

plot1 + theme_bw()
```



Statistical Testing

We assumed an IID normal distribution for a set of 26 yearly returns on the stock market and calculated a sample mean of 9.6% and sample standard deviation of 17.9%. Suppose that you consider investing in the stock market. You then expect to earn a return equal to μ percent every year.

Of course, you hope to make a profit. However, a friend claims that the expected return on the stock market is 0. Perhaps your friend is right. How can you use a statistical test to evaluate this claim?

A statistical hypothesis is an assertion about one or more parameters of the distribution of a random variable. Examples are that the mean mu is equal to 0, that it is nonnegative or larger than 5%, or that the standard deviation sigma is between 5 and 15%. We want to test one hypothesis, the null hypothesis against another one, the alternative hypothesis. We denote the null hypothesis by H0 and the alternative by H1. So H0 can be mu = 0 and H1, mu is unequal to 0.

A statistical test uses the observations to determine the statistical support for a hypothesis. It needs a test statistic t which is a function of the vector of observations y and a critical region C. If the value of the test statistic falls in the critical region, we reject the null hypothesis in favor of the alternative, if not we say that we do not reject the null hypothesis. Note that we do not say that we accept the null hypothesis. Suppose that we want to test the null hypothesis that mu is equal to 0, against the alternative that it is unequal to 0, with the variance sigma-squared known.

For a test statistic we use the sample mean. We define a critical region as the range below minus c and beyond c with c a positive constant. Small c is called the critical value. If the sample mean falls below minus c or beyond c, we reject the null hypothesis. The sample mean is then too far away from 0 for the null hypothesis to be true.

- If H null is false and the test rejects it, we call the outcome a true positive.
- If H null is true and the test does not reject it, we call it a true negative.
- If H null is true but a test rejects it, the outcome is a false positive or a type I error. If H null is false but a test does not reject it, the outcome is a false negative or type II error.

The probability of a type I error, so the probability to reject while the null hypothesis is true is called the **size of the test** or the significance level. The probability to reject while the null is false is called the **power of the test**. We prefer tests with small size and large power.

A smaller critical region means that we need larger deviations from the null hypothesis for a rejection. So the significance level decreases. However, this also means that the power of the test goes down. So in determining the critical region, we have to make a trade-off between size and power.

You can see an interactive hypothesis test calculator in my website

Example

Let's finish with the stock market example. The estimated mean and standard deviation were 9.6 and 17.9%.

The t statistic for the mean equal to 0 equals 2.75. The one-sided p-value = 0.54%. So for all significance levels beyond 0.54% we reject the null hypothesis in favor of the mean being positive.

The standard deviation of the stock market return is a measure for the risk of investing in the stock market. Suppose you want to limit your risk measured by the standard deviation to 25%. You test H0 that the standard deviation is equal to 25% against the alternative that it is smaller.

How would you decide?

The test statistic has a value of 12.74, which falls inside the critical region from 0 to 14.61. So we reject that the variance equals 25%. The p-value for a test equals 2.1%.

For more information look this website

```r
# t test for mean = 0
t.test(trainexers_s1$Return, mu=0)
```

```
##
##  One Sample t-test
##
## data:  trainexers_s1$Return
## t = 12.424, df = 999, p-value < 2.2e-16
```

```
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  4.951096 6.808421
## sample estimates:
## mean of x
##  5.879759
```

```r
ttest <- t.test(trainexers_s1$Return, mu=0)
```

Now, we want to determine how the sample size influences test statistics.

1. We want to test hypotheses of the form: $H_0 : \mu = \mu_0$ versus $H_1 : \mu \neq \mu_0$ Construct a series of statistics ti and corresponding p-values for $\mu_0 = 0\%$ and $\mu_0 = 6\%$ where t_i is the t-statistic based on the first i observations. Using the range $i = 5, 6...15$ make a table of t-statistics and p-values for both values of μ_0.

When calculating p-values we must take into account that the test is two sided. Then $p_i = 2\Psi_{i-1}(-|t_i|)$ where Ψ_n is the cumulative distribution function (CDF) of the t distribution function with n degrees of freedom.

The p-values are based on the upper bounds of Critical Region and remember the t distribution is symmetric.

```r
# We add the collumns for the t stat and p value for both mu_0 cases.
trainexers_s1 <- trainexers_s1 %>% mutate(t_stat_0=0,p_value_0=0,t_stat_6=0,p_value_6=0)
# We fill the columns using a for loop.
# These first two loops are for mu_0=0%
for (i in 1:length(trainexers_s1$Return)){
  trainexers_s1[i,7]= (trainexers_s1[i,3]-0)/(trainexers_s1[i,4]/sqrt(i))
}
for (i in 1:length(trainexers_s1$Return)){
  trainexers_s1[i,8]= 2*pt(-as.numeric(trainexers_s1[i,7]),i)
}
# The following two are for mu_0=6%
for (i in 1:length(trainexers_s1$Return)){
  trainexers_s1[i,9]= (trainexers_s1[i,3]-6)/(trainexers_s1[i,4]/sqrt(i))
}
for (i in 1:length(trainexers_s1$Return)){
  trainexers_s1[i,10]= 2*pt(-as.numeric(trainexers_s1[i,9]),i)
}

# We select the sub-sample for observations 5-15
sample_s1 <- trainexers_s1[5:15,]
sample_s1 <- sample_s1 %>% dplyr::select(Observation,t_stat_0,p_value_0,t_stat_6,p_value_6)
kable(sample_s1,booktabs = TRUE) %>%
  kable_styling()
```

Observation	t_stat_0	p_value_0	t_stat_6	p_value_6
5	2.0161737	0.0998565	0.5064665	0.6340658
6	1.9445906	0.0998047	0.2245242	0.8298002
7	1.5675766	0.1609655	-0.3235210	1.2442484
8	2.1789728	0.0609600	0.0679699	0.9474777
9	0.9997587	0.3435470	-1.2368711	1.7525639
10	1.8674154	0.0913953	-0.5650346	1.4154974
11	2.5308207	0.0279326	-0.1227762	1.0955012
12	2.6295448	0.0219941	-0.2420551	1.1871753
13	2.6074661	0.0216964	-0.4752585	1.3575113
14	2.6691808	0.0183292	-0.6216434	1.4558329
15	2.6295596	0.0189492	-0.8633142	1.5984417

Simple Regression.

Motivation

A simple example concerning the weekly sales of a product with a price that can be set by the store manager.

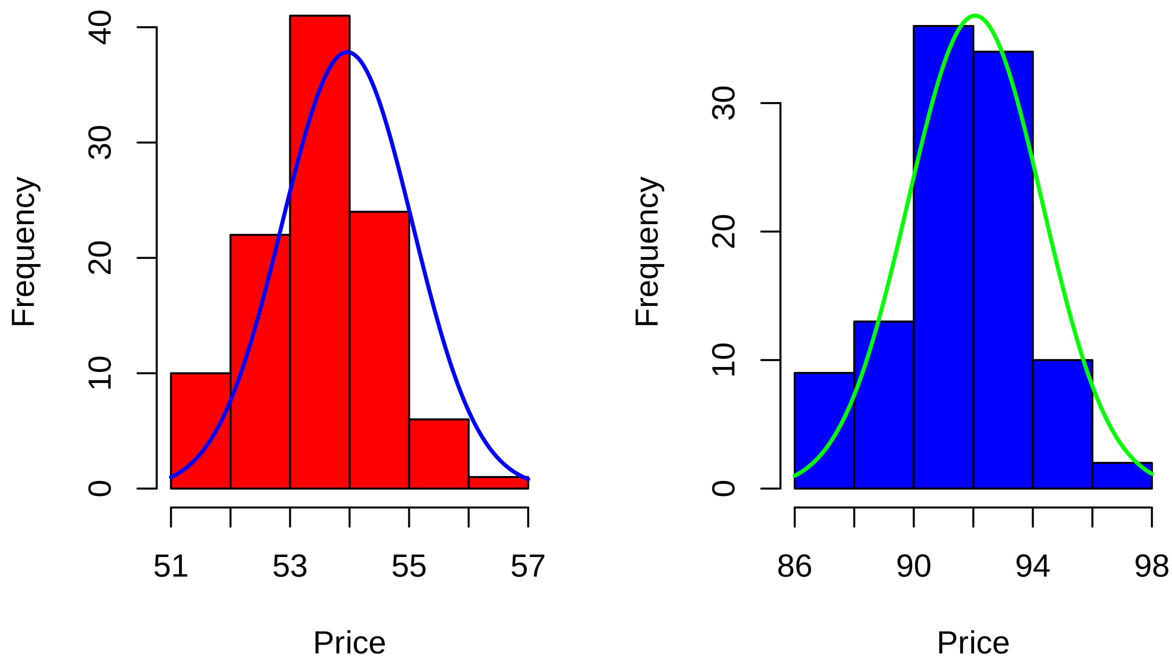
We'll use the following dataset:

Simulated price and sales data set with 104 weekly observations. - Price: price of one unit of the product - Sales: sales volume during the week

```
dataset1 <- read_csv(
  "https://raw.githubusercontent.com/diego-eco/diego-eco.github.io/master/downloads/week1_dataset1.csv")
```

Let's look at our sample:

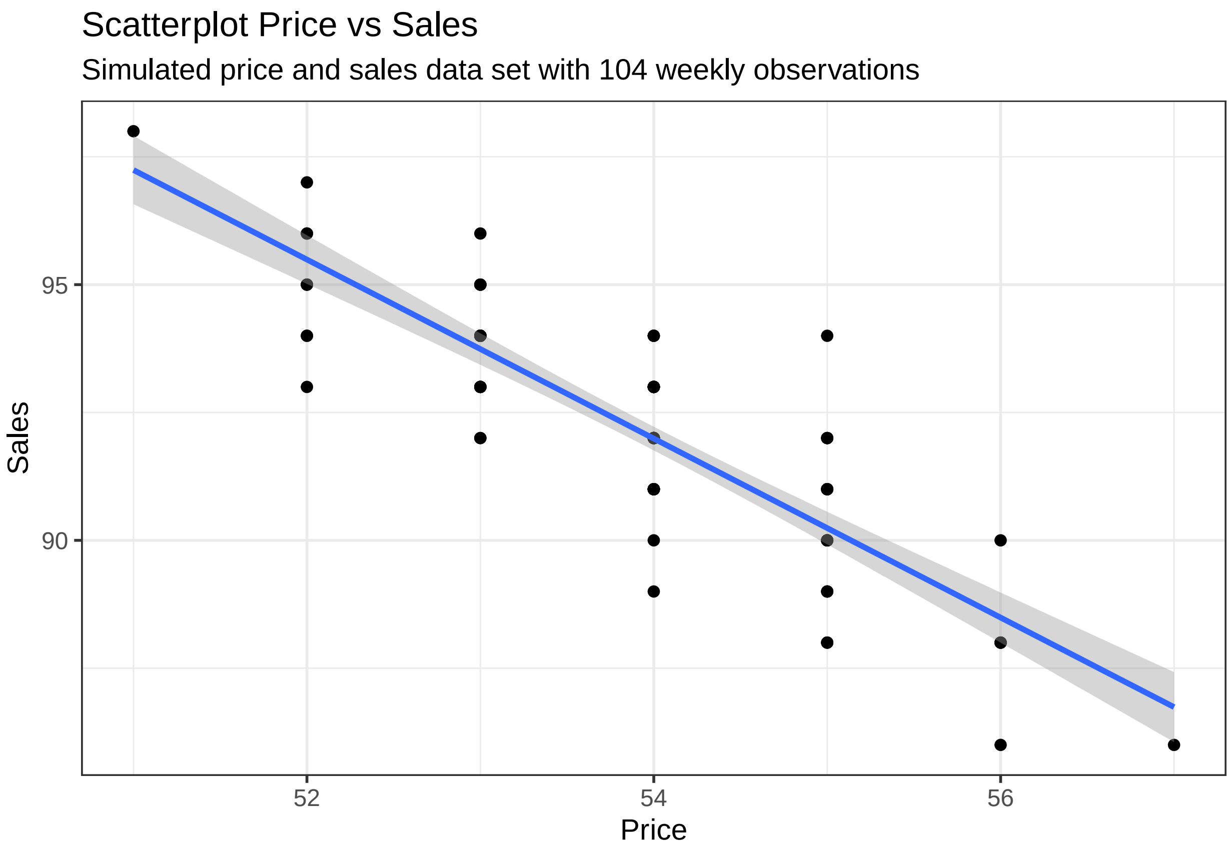
```
par(mfrow=c(1,2))
plotNormalHistogram(dataset1$Price, prob = FALSE, col = "red",
                    xlab="Price", ylab="Frequency",
                    linecol = "blue", lwd = 2)
plotNormalHistogram(dataset1$Sales, prob = FALSE, col = "blue",
                    xlab="Price", ylab="Frequency",
                    linecol = "green", lwd = 2)
```



We expect that lower prices lead to higher sales. The econometrician tries to quantify the magnitude of these consumer reactions to such price changes. This helps the store manager to decide to increase or decrease the price if the goal is to maximize the turnover for this product. Turnover is sales times price. You can see that the majority of weekly sales are somewhere in between 90 and 95 units, with a minimum of 86 and a maximum of 98. Sales of 92 and 93 units are most often observed, each 19 times. The store manager can freely decide each week on the price level, presented on the next slide.

When we plot sales against price that occur in the same week, we get the following scatter diagram.

```
plot2 <- ggplot(data=dataset1, aes(x=Price,y=Sales)) + geom_point() + geom_smooth(method='lm') +
  labs(title="Scatterplot Price vs Sales ",
       subtitle="Simulated price and sales data set with 104 weekly observations")

plot2 + theme_bw()
```



from the scatter plot of sales and price data, you see that different price levels associate with different sales levels. And this suggests that you can use the price to predict sales.

$$Sales = a + b \cdot Price$$

This equation allows us to predict the effects of a price cut that the store manager did not try before, or to estimate the optimal price to maximize **turnover.**(sales times price)

In simple regression, we focus on two variables of interest we denote by y and x, where one variable, **x**, is thought to be helpful to predict the other, **y**. This helpful variable x we call the regressor variable or the explanatory factor. And the variable y that we want to predict is called the dependent variable, or the explained variable.

We can say from our histogram

$$Sales \sim N(\mu, \sigma^2)$$

This notation means that the observations of sales are considered to be independent draws from the same Normal distribution, with mean mu and variance sigma squared, abbreviated as NID. Note that we use the Greek letters mu and sigma squared for parameters that we do not know and that we want to estimate from the observed data. The probability distribution of sales is described by just two parameters, the mean and the variance. On this slide you see the graph of a standardized normal distribution with mean 0 and variance 1. And if you wish, you can consult the Building Blocks for further details on the normal distribution.

For a normal distribution with mean mu, the best prediction for the next observation on sales is equal to that mean mu. An estimator of the population mean mu is given by the sample mean $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_n$, where y subscript i denotes the i-th observation on sales. The sample mean is called an unconditional prediction of sales, as it does not depend on any other variable.

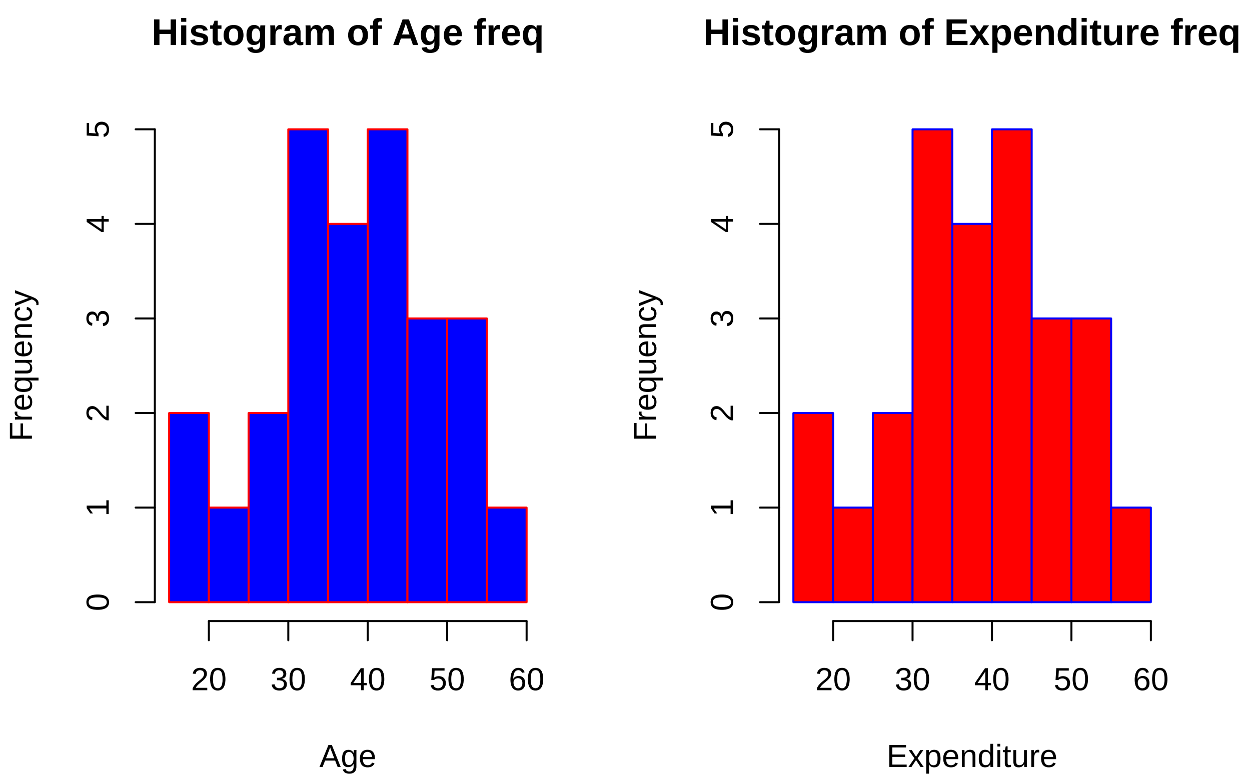
Example:

TrainExer1_1 Simulated data set on holiday expenditures of 26 clients. - Age: age in years - Expenditures: average daily expenditures during holidays

```
dataset2 <- read_csv(
  "https://raw.githubusercontent.com/diego-eco/diego-eco.github.io/master/downloads/trainexer1_1.csv")
```

1. Make two histograms, one of expenditures and the other of age. Make also a scatter diagram with expenditures on the vertical axis versus age on the horizontal axis.

```
par(mfrow=c(1,2))
hist(dataset2$Age,xlab = "Age",col = "blue",border = "red",
     main = "Histogram of Age freq")
hist(dataset2$Age,xlab = "Expenditure",col = "red",border = "blue",
     main = "Histogram of Expenditure freq")
```




```
plot3 <- ggplot(data=dataset2, aes(x=Age,y=Expenditures)) + geom_point() +
  labs(title="Scatterplot Expenditures vs Age ",
       subtitle="Simulated data set on holiday expenditures of 26 clients.")

plot3 + theme_bw()
```

The points in the scatter doesn't associate with a single line, there appears to be two groups in the samples, a group of people younger than 40 and another group older than 40 years old.

- In what respect do the data in this scatter diagram look different from the case of the sales and price data discussed in the last section? Propose a method to analyze these data in a way that assists the travel agent in making recommendations to future clients.

The scatter diagram indicates two groups of clients. Younger clients spend more than older ones. Further, expenditures tend to increase with age for younger clients, whereas the pattern is less clear for older clients.

```r
summary(dataset2$Age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   15.00   35.00   39.50   39.35   45.75   57.00
```

```r
summary(dataset2$Expenditures)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    89.0    96.0   103.0   101.1   106.8   109.0
```

- Compute the sample mean of expenditures of all 26 clients.

```r
#dataset2_descr <- psych::describe(dataset2)
#as_data_frame(dataset2_descr)
# item name ,item number, nvalid, mean, sd,
# median, mad, min, max, skew, kurtosis, se
print(paste("The mean of the expenditures of clients is ",
            mean(dataset2$Expenditures)))
```

```
## [1] "The mean of the expenditures of clients is  101.115384615385"
```

- Compute two sample means of expenditures, one for clients of age forty or more and the other for clients of age below forty.

```
dataset2_over40 <-dataset2 %>% filter(Age>=40)
dataset2_below40 <-dataset2 %>% filter(Age<40)
print(paste("The mean of the expenditures of clients over 40 is ",
            mean(dataset2_over40$Expenditures)))
```

```
## [1] "The mean of the expenditures of clients over 40 is  95.8461538461538"
```

```
print(paste("The mean of the expenditures of clients below 40 is ",
            mean(dataset2_below40$Expenditures)))
```

```
## [1] "The mean of the expenditures of clients below 40 is  106.384615384615"
```

- What daily expenditures would you predict for a new client of fifty years old? And for someone who is twenty-five years old?

Someone of fifty (in older that 40 group) is expected to spend (unconditional prediction) $95.84, someone of twenty-five (in younger that 40 group) is expected to spend (unconditional prediction) $ 106.38

Representation

We formalized the notion that you can use values of variable to predict the values of another variable. As in the previous lecture we will consider again the scatter plot of sales against price. Hence, knowing the price to be high or low results in a different sales prediction. In other words, it helps to explain sales by using price as an explanatory factor.

Therefore we will call sales the **dependent variable**, and price the **explanatory variable or explanatory factor**. For dependent variable y with observations y subscript i, we can assume as we did for the sales data in the first lecture that y is identically distributed as normal with mean mu and variance sigma squared. $y \sim N(\mu, \sigma^2)$.

In that case, the expected value with notation E of y is equal to μ. And the variance of y is equal to sigma squared σ^2. Again, you can consult the building blocks for further details.

An estimator of the population mean μ is given by the **sample mean**, y bar $\bar{y}$.

$$\hat{\mu} = \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

And an estimator for sigma squared is the **sample variance**.

$$\hat{\sigma}^2 = \frac{1}{n-1}\sum_{i=1}^{n}(y_i - \bar{y})^2$$

The idea of using one variable to predict the other instead of just using the sample mean means that we move from an unconditional mean to a conditional mean given a value of x. For example, the conditional mean can be alpha plus beta times x.

- Unconditional prediction with $y \sim N(\mu, \sigma^2)$: $E(y_i) = \mu$
- Conditional prediction with $y \sim N(\alpha + \beta x_i, \sigma^2)$: $E(y_i) = \alpha + \beta x_i$

An alternative way of writing the conditional prediction follows from y, by subtracting the linear relation, alpha plus beta times x. Such that a normally distribute error term would mean mu emerges. $\epsilon_i \sim N(0, \sigma^2)$

$$y_i = \alpha + \beta x_i + \epsilon_i$$

If x_i is fixed (not random) then y_i has mean $\alpha + \beta x_i$ and variance σ^2.

The expressions together form the simple regression model that says that the prediction of y for a given value of x is equal to alpha plus beta times x. This simple regression model contains a single explanatory variable. And therefore, anything that is not in the model is covered by the error epsilon. For example, for the sales and price example, we did not include the prices of competing stores or the number of visitors through the store in each week.

Small values of the errors epsilon one to epsilon n associated with more accurate predictions of sales, than when these errors are large. So if we would have estimates of these errors, then we can evaluate the quality of the predictions. To get these estimates, we first need to estimate alpha and beta.

The parameter beta in the simple regression model has the input notation of the derivative of y with respect to x. $\beta = \frac{\partial y}{\partial x}$. This is also called the **slope of the regression or the marginal effect.**

In economics, we often use the concept of elasticity which measures, for example, the percentage increase in sales associated with 1% decrease in price.This facilitates the interpretation and as the elasticity is scale free, it also allows for a comparison across cases, like related retail stores.

The elasticity is defined as the relative change and y, that is d y, divided by y caused by the relative change d x divided by x.

$$Elasticity = \frac{\frac{\partial y}{y}}{\frac{\partial x}{x}}$$

In our linear model the elasticity is calculated as:

$$\frac{\frac{\partial y}{y}}{\frac{\partial x}{x}} = \frac{\partial y}{\partial x} \cdot \frac{x}{y}$$

If the relationship between prize and sales is linear, the value of the elasticity depends on the value of the sales (y) and prize (x). This dependence makes it difficult, for example, to compare across retail stores with different floor sizes.

To facilitate such comparisons, store managers prefer a measure of elasticity that does not depend on the ratio x over y. To achieve that, one can **transform the y and x variables by taking the natural logarithm, written as log.**

Take the linear model $log(y) = \alpha + \beta \cdot log(x)$ that has an $Elasticity = \beta$.

- In this notes $log()$ denotes the natural logarithm, with base $e = 2.71828$ often also noted as $ln()$
- Remeber the derivative of $log(x)$ with baes e is $\frac{1}{x}$

Notes: A transformation of the data on x_i and y_i (like taking their logarithm) changes the interpretation of the slope parameter β.

- In the model $y_i = \alpha + \beta log(xi) + \epsilon_i$ the $Elasticity = \frac{\beta}{y_i}$.
- In the model $log(y_i) = \alpha + \beta xi + \epsilon_i$ the $Elasticity = \beta \cdot x_i$.

Estimation

A simple regression model $y_i = \alpha + \beta xi + \epsilon_i$.

In econometrics, we don't know α, β and ϵ_i, but we do have observations x_i and y_i. We will use observed data on x and y to find optimal values of the coefficients a and b. The line y is a + bx is called the regression line.

$$y_i \approx a + bx_i$$

The line $y = a + bx_i$ is called the **Regression line**. We have n pairs of observations on x and y, and we want to find the line that gives the best fit to these points. The idea is that we want to explain the variation in the outcomes of the variable y by the variation in the explanatory variable x. Think again of the high price low sales combinations, in the previous lecture, versus the low price, high sales combinations.

When we use the linear function a + bx to predict y, then we get residuals e. And we want to choose the fitted line such that these residuals are small. Minimizing the residuals seems a sensible strategy to find the best possible values for a and b.

And a useful objective function is the **sum of squared residuals**.

$$S(a,b) = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (y_i - a - bx_i)^2$$

This way of finding values for a and b is called the **method of least squares, abbreviated as LS.** The minimum of the objective function is obtained by solving the first order conditions. This is done by taking the partial derivatives of the objective function and setting these to 0. To see more of the calculations take a look at the Building Blocks

Solving $\frac{\partial S}{\partial a} = 0$ and $\frac{\partial S}{\partial b} = 0$ yields:

Let us start with the coefficient a. Solving the first order condition gives that minus 2 times the sum of the residuals is equal to 0. Note that when the sum of the residuals equals 0, then one of the residuals is a function of the other, n- 1 residuals.

$$a = \frac{1}{n}\sum_{i=1}^{n} y_i - b\frac{1}{n}\sum_{i=1}^{n} x_i \Rightarrow \text{Simplifying} \Rightarrow a = \bar{y} - b\bar{x}$$

When you take the partial derivative of the objective function to b, you get that the sum of the observations on x times the residuals e is equal to 0. Note that this puts another restriction on the n values of e. **This implies that of the n values of e, two are found from the other n-2 values.** And now we derive the expression for b. We can use a few results on summations and means, which leads to a more convenient expression for b.

$$b = \frac{\sum_{i=1}^{n} x_i(y_i - \bar{y})}{\sum_{i=1}^{n} x_i(x_i - \bar{x})} \Rightarrow \text{Simplifying} \Rightarrow b = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

This important expression shows that b is equal to the sample covariance of y and x divided by the sample variance of x.

I now invite you to consider the following test question. What happens to b if all y observations are equal? The answer is that then b is equal to 0. So if there is no variation in y, there is no need to include any x to predict the values of y.

When we fit a straight line to a scatter of data, we want to know how good this line fits the data. And one measure for this is called the **R-squared**.

The line emerges from explaining the variation in the outcomes of the variable y by means of the variation in the explanatory variable x.

$$y_i = a + bx_i + e_i = \bar{y} - b\bar{x} + bx_i$$

So:

$$y_i - \bar{y} = b(x_i - \bar{x}) + e_i$$

Deviations $y_i - \bar{y}$ partly explained by $x_i - \bar{x}$ but e_i is unexplained.

By construction $\sum_{i=1}^{n} e_i = 0$ and $\sum_{i=1}^{n} x_i e_i = 0$ hence $\sum_{i=1}^{n} (x_i - \bar{x}) e_i = 0$

Squaring and summing (SS) both sides of $y_i - \bar{y} = b(x_i - \bar{x}) + e_i$ therefore gives:

$$\sum_{i=1}^{n} (y_i - \bar{y})^2 = b^2 \sum_{i=1}^{n} (x_i - \bar{x})^2 + \sum_{i=1}^{n} e_i^2$$
$$SSTotal = SSExplained + SSResidual$$

Now R-squared is defined as the fraction of the variation in y that is explained by the regression model. When R-squared is 0, there is no fit at all. When the R-squared is 1, the fit is perfect.

$$R^2 = \frac{SSExplained}{SSTotal} = 1 - \frac{\sum_{i=1}^{n} e_i^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2}$$

Next we estimate the **unknown variance of the epsilons from the residuals**. σ_ϵ^2 is estimated from residuals $e_i = y_i - a - bx_i$. Residuals $e_i, i = 1, 2, ...n$ have $n - 2$ free values (as seen before). Then

$$s_\epsilon^2 = \frac{1}{n-2} \sum_{i=1}^{n} (e_i - \bar{e})^2$$

Let's look at an example:

Dataset: TrainExer13 Winning time 100 meter athletics for men at Olympic Games 1948-2004. - Year: calendar year of Olympic Game (1948-2004) - Game: order number of game (1-15) - Winmen: winning time 100 meter athletics for men (in seconds)

```
dataset3 <- read_csv(
  "https://raw.githubusercontent.com/diego-eco/diego-eco.github.io/master/downloads/trainexer13.csv")
```

A simple regression model for the trend in winning times is

$$W_i = \alpha + \beta G_i + \epsilon_i$$

1. Compute a and b, and determine the values of R^2 and s.

To solve this exercise with the tools we know so far, we can use our formulas:

```
print(paste("The mean of the winning time 100 meter athletics for men is ",
            mean(dataset3$`Winning time men`)))
```

```
## [1] "The mean of the winning time 100 meter athletics for men is  10.082"
```

```
print(paste("The mean of the order number of game is ",
            mean(dataset3$Game)))
```

```
## [1] "The mean of the order number of game is  8"
```

First, we calculate the sample mean for W_i and $G_i \Rightarrow \frac{1}{n} \sum_{i=1}^{n} W_i = 10.082$, $\frac{1}{n} \sum_{i=1}^{n} G_i = 8$

$$b = \frac{\sum_{i=1}^{15} (W_i - \bar{W})(G_i - \bar{G})}{\sum_{i=1}^{15} (G_i - \bar{G})^2} = -0.038$$

```r
b = sum((dataset3$`Winning time men`-mean(dataset3$`Winning time men`))*
          (dataset3$Game-mean(dataset3$Game)))/
  sum((dataset3$Game-mean(dataset3$Game))^2)
print(paste("The estimated b is",b))
```

```
## [1] "The estimated b is -0.038"
```

$$a = \frac{1}{n}\sum_{i=1}^{n} W_i - b\frac{1}{n}\sum_{i=1}^{n} G_i = 10.386$$

```r
a = mean(dataset3$`Winning time men`) - b*mean(dataset3$Game)
print(paste("The estimated a is",a))
```

```
## [1] "The estimated a is 10.386"
```

Let's calculate the errors $e_i = W_i - a - bG_i$ for $i = 1, 2, ...15$:

```r
# Create a column for the errors
dataset3 <- dataset3 %>% mutate(errors=0)
# Fill this column with a for loop
for (i in 1:length(dataset3$errors)) {
  dataset3[i,4]=dataset3[i,3]-a-b*dataset3[i,1]
}
```

$$R^2 = 1 - \frac{\sum_{i=1}^{n} e_i^2}{\sum_{i=1}^{n}(W_i - \bar{W})^2} = 0.673$$

```r
r_squared = 1 - (sum(dataset3$errors^2)/sum((dataset3$`Winning time men`-mean(dataset3$`Winning time men
print(paste("R^2 is",r_squared))
```

```
## [1] "R^2 is 0.673372859902738"
```

$$s_\epsilon^2 = \frac{1}{15-2}\sum_{i=1}^{n}(e_i - \bar{e})^2 = 0.013$$

```r
var_res = (1/length(dataset3$errors))*sum((dataset3$errors - mean(dataset3$errors))^2)
print(paste("The variance of residuals is",var_res))
```

```
## [1] "The variance of residuals is 0.0130746666666667"
```

```r
sd_res = sqrt(var_res)
print(paste("The standard deviation of residuals is",sd_res))
```

```
## [1] "The standard deviation of residuals is 0.114344508686105"
```

All these calculations, of course, could be automatically done with a regression package such as **lm()**

```r
lm1 <- lm(`Winning time men` ~ Game , data = dataset3)
```

Regression Results

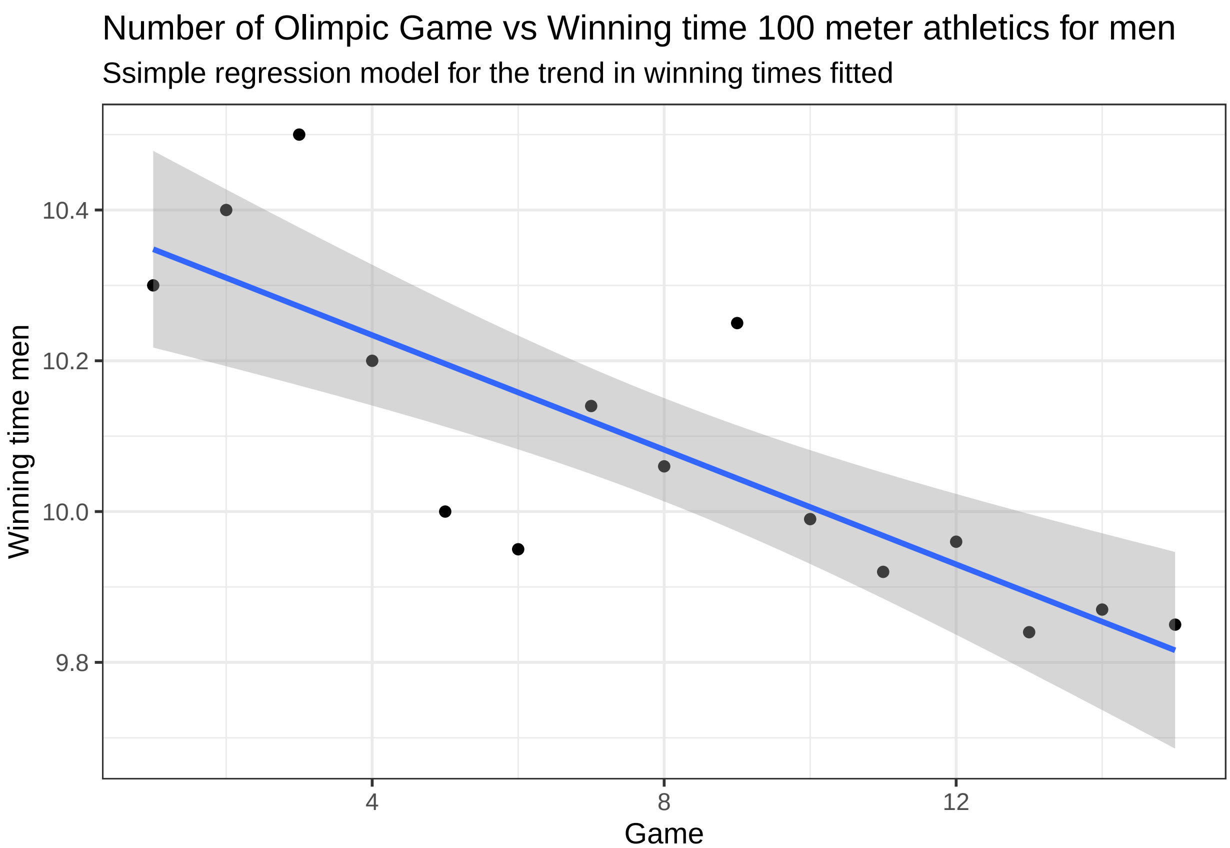
```
                    Dependent variable:
                ---------------------------
                    `Winning time men`
```

Game -0.038*** (0.007)
Constant 10.386*** (0.067)

Observations 15
R2 0.673
Adjusted R2 0.648
Residual Std. Error 0.123 (df = 13)
F Statistic 26.801*** (df = 1; 13)
=== Note: *p<0.1;* ***p<0.05;***
p<0.01

We can also visualize our linear model:

```
plot3 <- ggplot(data=dataset3, aes(x=Game,y=`Winning time men`)) + geom_point() + geom_smooth(method='l
  labs(title="Number of Olimpic Game vs Winning time 100 meter athletics for men ",
       subtitle="Ssimple regression model for the trend in winning times fitted")
plot3 + theme_bw()
```



2. Are you confident on the predictive ability of this model?

Our $R^2 = 0.67$ tell us that about 67% of the variance in the winning times can be explained by the game trends. Moreover, the estimated residuals are quite low relative to the winning times.

3. What prediction do you get for 2008, 2012, and 2016?

Recall our data set goes up by steps of 4 years starting with 1948 with number 1 and finishing with 2004 with number 15, so we need to calculate with the corresponding numbers for the years 2008, 2012, and 2016.

Our estimated model is $W_i = 10.39 - 0.038 \cdot G_i$ so we can use it to predict for G_{16}, G_{17}, G_{18}

In R we save our model as an object so we can use it latter for further analysis, in this case we can predict using a new data. You can predict the corresponding stopping distances using the R function predict().The confidence interval reflects the uncertainty around the mean predictions. To display the 95% confidence intervals around the mean the predictions, specify the option interval = "confidence":

The output contains the following columns:

- fit: the predicted sale values for the three new advertising budget
- lwr and upr: the lower and the upper confidence limits for the expected values, respectively. By - - - default the function produces the 95% confidence limits.

```
new.games <- data.frame(Game = c(16, 17, 18))
predict(lm1,new.games, interval = "confidence")
```

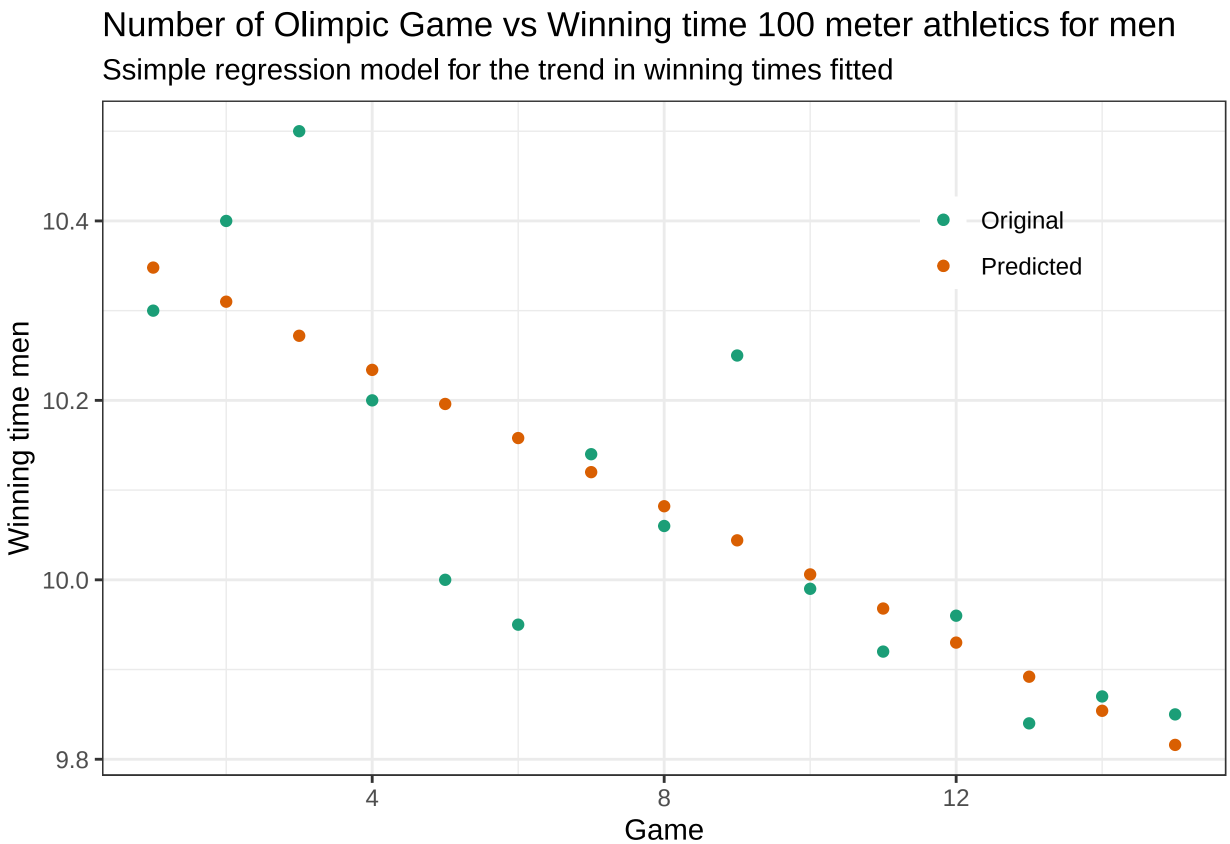
```
##     fit      lwr      upr
## 1 9.778 9.633821 9.922179
## 2 9.740 9.581688 9.898312
## 3 9.702 9.529256 9.874744
```

How do we add our predicted values to the data frame?

```
# Add predictions to data set
predicted <- rbind(predict(lm1, interval = "confidence"))
dataset3 <- cbind(dataset3, predicted)
```

Let see our original values and our predicted values:

```
plot3a <- ggplot(data=dataset3, aes(x=Game,y=`Winning time men`,color='Original')) + geom_point() +
  geom_point(aes(y = fit, color = "Predicted")) +
  labs(title="Number of Olimpic Game vs Winning time 100 meter athletics for men ",
       subtitle="Ssimple regression model for the trend in winning times fitted")
plot3a + theme_bw() + theme( legend.position = c(.8, .8),
        legend.background = element_rect(fill = "transparent") ) +
  scale_color_brewer(name= NULL, palette = "Dark2")
```



Evaluation

Previously, you learned how to fit a straight line to a scatter of points x and y. You can calculate the coefficients a and b of the regression line and its associated residuals with their standard deviation. And you can use these results to answer the question how to predict a new value of y_0 given a value for x_0.

The actual value follows from the theoretical expression of the regression model. And, as we do not know the specific epsilon, the point prediction is, of course, a plus b times a value for x.

$$\text{Actual Value : } y_0 = \alpha + \beta x_0 + \epsilon_0$$

$$\text{Predicted Value : } \hat{y}_0 = a + bx_0$$

The interval for the epsilon term is chosen as plus and minus k times the standard deviation of the residuals. This gives the prediction interval for y $\epsilon_0 : (-ks, ks)$ The interval for the epsilon term is chosen as plus and minus k times the standard deviation of the residuals.

$$\text{Predicted Interval for y : } (\hat{y}_0 - ks, \hat{y}_0 + ks)$$

The wider is the prediction interval, the more likely it is that the actual observation is in that interval.

We now turn to the **statistical properties of b**. That is, we want to quantify the uncertainty that we have for an obtained value of b for actual data.

In order to evaluate how accurate b is for beta, we can rely on seven assumptions. The idea is to link the actual value of b to the properties of the errors, epsilon, in the data generating process, for which we can make a set of statistical assumptions.

1. The first assumption is that y is related in a linear way to x. This is called the Data Generating Process or DGP. The idea is that the postulated model for the data matches with the DGP as both are based on a linear relation between x and y.

$$\textbf{A1.}\quad \text{DGP}: y_i = \alpha + \beta x_i + \epsilon_i$$

2. The second assumption is that all observations on x are fixed numbers. Think of a store manager who sets prices each time at the beginning of the week.

$$\textbf{A2.}\quad \text{The n observations of } x_i \text{ are fixed numbers}$$

3. Assumption three is that the n errors epsilon are random draws from a distribution with mean zero.

$$\textbf{A3.}\quad \text{The n error terms } \epsilon_i \text{ are random with } E(\epsilon_i) = 0$$

4. Assumption four says that the variance of the n errors is a constant, which means that all observations are equally informative about the underlying DGP. This assumption is usually called homoscedasticity, meaning something like "equal stretching", and it is opposite of heteroscedasticity.

$$\textbf{A4.}\quad \text{The variance of n error is constant } E(\epsilon_i^2) = \sigma^2$$

5. Assumption five is that the error terms are not correlated.

$$\textbf{A5.}\quad \text{The n error terms uncorrelated } E(\epsilon_i \epsilon_j) = 0 \text{ for all } i \neq j$$

6. Assumption six is that the unknown coefficients alpha and beta are the same for all n observations, and thus there is only a single straight line to fit to the scatter of the data.

$$\textbf{A6.}\quad \alpha \text{ and } \beta \text{ are unknown, but fixed for all observations}$$

7. And the final assumption is that the errors epsilon are jointly, normally and identically distributed.

$$\textbf{A7.}\quad \epsilon_1 ... \epsilon_n \text{ are Jointly Normally Distributed; With A3, A4, A5: } \epsilon_i \sim NID(0, \sigma^2)$$

With these seven assumptions, we can determine the precise statistical properties of the slope estimator b. And in particular we can find expressions for its mean value and its variance. We will start with the mean of b, and we will see how far off b is from beta. The crucial idea is to express b in terms of the random variables epsilon, because the assumptions imply the statistical properties of these epsilons.

Remember that $b = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$

With some algebra we can arrive to:

$$b = \beta + \frac{\sum_{i=1}^{n}(x_i - \bar{x})\epsilon_i}{\sum_{i=1}^{n}(x_i - \bar{x})^2} = \beta + \sum_{i=1}^{n} c_i \epsilon_i$$

With $c_i = \frac{x_i - \bar{x}}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$.

As you can see, this expression for b is not very useful to obtain the estimator parameter, but it will be useful to see that b is an unbiased estimator of β.

$$E(b) = E(\beta) + \sum_{i=1}^{n} c_i E(\epsilon_i)$$

With Assumptions A6 and A3 we can see that

$$E(b) = \beta$$

The amount of uncertainty in the outcome b is measured by the variance. We start again with the familiar expression for b with beta on the right hand side. The variance of b can be derived from the statistical properties of the epsilons.

$$b = \beta + \sum_{i=1}^{n} c_i \epsilon_i$$

$$\sigma_b^2 = var(b) = \sigma^2 \sum_{i=1}^{n} c_i^2 = \frac{\sigma^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

Assumption A7 states that the epsilons are distributed as normal. And as b is a linear function of the epsilons, it is also normal. If you wish, you can consult the Building Blocks for this property of the normal distribution.

$$b \sim N(\beta, \sigma_b^2)$$

As usual, this distribution can be standardized to give the Z-score, which is distributed as standard normal.

$$Z = \frac{b - \beta}{\sigma_b^2}$$

For practical use, we need to estimate the variance sigma b squared. An unbiased estimate of that variance is s squared divided by the sum of squares of the variable x, where s squared is the estimated variance of the residuals.

We replace the unknown σ_b^2 by $s^2 = \frac{1}{n-2}\sum_{i=1}^{n} e_i^2$ then:

$$s_b^2 = \frac{s^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

The t value is defined as b minus beta divided by the standard deviation of b. This t-value is distributed as t with n-2 degrees of freedom. We refer you again to the Building Blocks for the relation between the normal and t-distribution. When n is large enough, the t-distribution is approximately the same as the standard normal distribution.

$$t_b = \frac{b - \beta}{s_b} \sim t(n-2)$$

As a rule of thumb, we reject the null hypothesis that beta is 0 when the absolute t value is larger than 2. The approximate 95% confidence interval of the standard normal distribution is the interval -2 to 2. We can use this to create an approximate confidence interval for beta.

t-test on $H_o : \beta = 0$ based on $t_b = \frac{b}{s_b}$.

Rule of thumb for large n:

$$\text{Reject } H_o \text{ if } t_b < -2 \text{ or } t_b > 2$$

Following this line of thought, we can also derive an approximate 95% prediction interval for a new value of y, corresponding to any given new value of x, as shown on the slide.

In practice when you run a regression, you should always check if you find these assumptions reasonable. At the same time, we should ask how bad it is if some of the assumptions are not precisely met, which is quite common for actual data.

Example:

The purpose of the exersice is to understand the consequences of measurement errors and the amount of bias resulting:

Consider the situation where the x-variable is observed with measurement error, which is rather common for complex macroeconomic variables like national income.

Let x^* be the true, unobserved economic variable, and let the data generating process (DGP) be given b $y_i = \alpha + \beta x_i^* + \epsilon_i^*$ where x_i^* and ϵ_i^* are uncorrelated.

The observed x-values are $x_i = x_i^* + v_i$ with measurement errors vi that are uncorrelated with x_i^* and ϵ_i^*. The **signal-to-noise ratio** is defined as $SN = \frac{\sigma_*^2}{\sigma_v^2}$ where σ_*^2 is the variance of x_i^* and σ_v^2 is the variance of v.

The estimated regression model is $y_i = \alpha + \beta x_i + \epsilon_i$ and we consider the least squares estimator b of β.

- Do you think that the value of b depends on the variance of the measurement errors? Why?

The value of LS estimator b will depend on the variance of the measurement errors because we know that $b = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2}$ which includes the variance of x_i which incorporates both σ_*^2 and σ_v^2.

- It can be shown that $b = \beta + \frac{\sum_{i=1}^n (x_i - \bar{x})(\epsilon_i - \bar{\epsilon})}{\sum_{i=1}^n (x_i - \bar{x})^2}$
- It can be shown that $\epsilon_i = \epsilon_i^* - \beta v_i$

By substituting $x_i = x_i^* + v_i$ in the DGP.

- It can be shown that the covariance between x_i and ϵ_i is equal to $-\beta\sigma_v^2$

The covariance between the error term and x_i is not equal to 0 anymore, it can be seen from expression. $cov(x_i, \epsilon_i) = cov(x_i^* + v_i, \epsilon_i^* - \beta v_i) = -\beta cov(v_i, v_i) = -\beta\sigma_v^2$

- It can be shown that for large sample size n we get $b - \beta \approx \frac{-\beta\sigma_v^2}{\sigma_*^2 + \sigma_v^2}$

With the result $b = \beta + \frac{\sum_{i=1}^n (x_i - \bar{x})(\epsilon_i - \bar{\epsilon})}{\sum_{i=1}^n (x_i - \bar{x})^2}$ we can divide both numerator and denominator by n so that

$$b = \beta + \frac{\frac{1}{n}\sum_{i=1}^n (x_i - \bar{x})(\epsilon_i - \bar{\epsilon})}{\frac{1}{n}\sum_{i=1}^n (x_i - \bar{x})^2}$$

Notice that with large n, $\frac{1}{n}\sum_{i=1}^n (x_i - \bar{x})(\epsilon_i - \bar{\epsilon}) \approx cov(x_i, \epsilon_i) = -\beta\sigma_v^2$

Also notice that with large n, $\frac{1}{n}\sum_{i=1}^n (x_i - \bar{x})^2 \approx var(x_i) = var(x_i^* + v_i) = \sigma_*^2 + \sigma_v^2$

- Using this last result we can use the SN ratio to simplify the expression for the Bias:

$$b - \beta \approx \frac{-\beta\sigma_v^2}{\sigma_*^2 + \sigma_v^2} = \frac{-\beta}{\frac{\sigma_*^2}{\sigma_v^2} + 1} = \frac{-\beta}{SN + 1}$$

Applications

This last section on simple regression showed you two illustrations. One on the price and sales data discussed before. And another one, on winning times for the Olympic 100 meter in Athletics. Recall the scatter diagram of sales against prices. And recall the model that $Sales = \alpha + \beta \cdot Price + \epsilon$

```r
lm2 <- lm(Sales ~ Price , data = dataset1)
```

Regression Results

```
        Dependent variable:
---------------------------
               Sales
```

Price	-1.750*** (0.107)
Constant	186.507*** (5.767)

Observations 104
R2 0.725
Adjusted R2 0.722
Residual Std. Error 1.189 (df = 102)
F Statistic 268.303*** (df = 1; 102)
=== Note: *p<0.1;* ***p<0.05;***
p<0.01

The least squares estimation results are shown in this table. Coefficient a is estimated as 186. And b as minus 1.75. The R squared is about 0.7, and the standard deviation of the residuals is about 1.2. Clearly, the two t-statistics are larger than 2 in absolute value, and hence the coefficients are significantly different from zero. This can also be seen from the very small p-values. You can find more information on t-values and p-values in the Building Blocks.

The 95 percent confidence interval of beta can be computed using plus and minus two times the estimated standard error of b. And this results in the interval that runs from minus 1.964 to minus 1.536. Clearly, this interval does not contain zero.

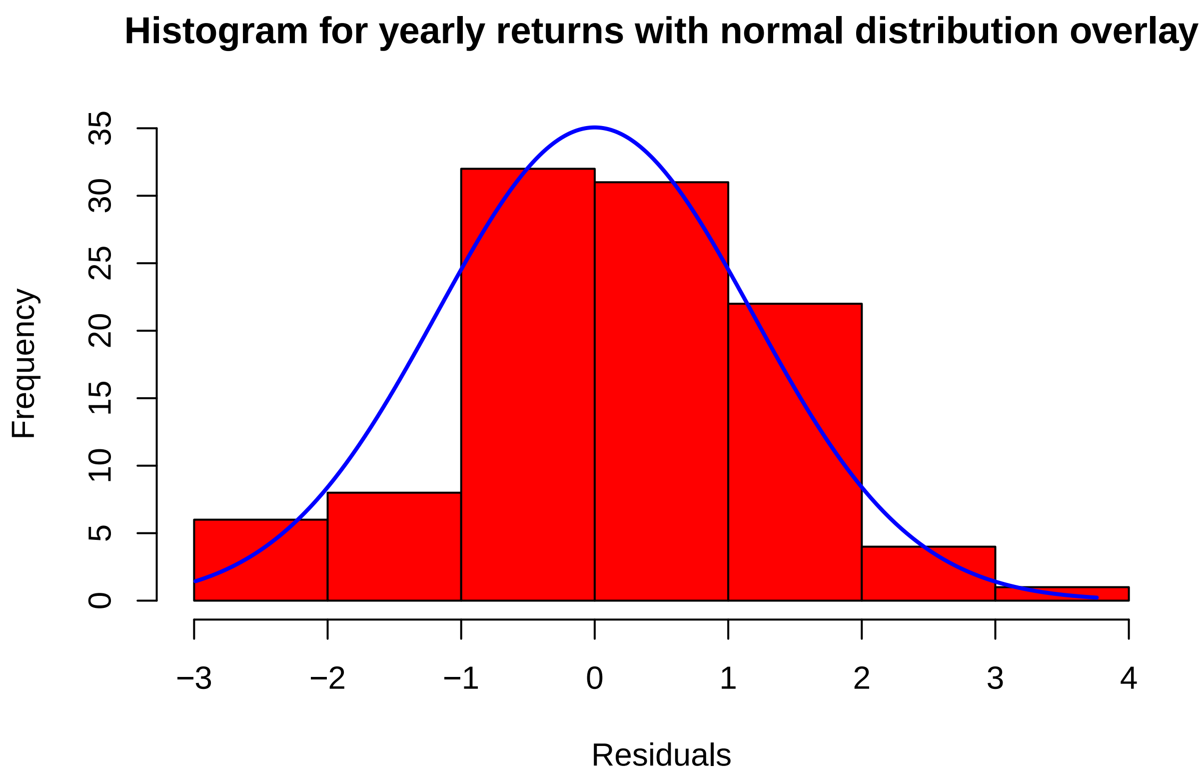
$$-1.75 - 2 \cdot 0.107 \leq \beta \leq -1.75 + 2 \cdot 0.107$$

$$-1.964 \leq \beta \leq -1.536$$

This interval means that we are 95% confident that, when the price goes down by 1 unit, sales will go up by about 1.5 to 2 units. And this is a significant effect.

The histogram of the 104 residuals is given in this graph.

```r
lm2.res = resid(lm2)
plotNormalHistogram(lm2.res, prob = FALSE, col = "red",
                    main = "Histogram for yearly returns with normal distribution overlay",
                    xlab="Residuals", ylab="Frequency",
                    linecol = "blue", lwd = 2)
```




```r
print(paste("The mean of the residuals is ", mean(lm2.res)))
```

```
## [1] "The mean of the residuals is  1.24568574606051e-17"
```

```r
#Expected 0
print(paste("The standard deviation of the residuals is ", sd(lm2.res)))
```

```
## [1] "The standard deviation of the residuals is  1.18338158160322"
```

```r
#Expected 1
print(paste("The skewness of the residuals is ", skewness(lm2.res)))
```

```
## [1] "The skewness of the residuals is  0.0292290449690796"
```

```r
#Expected 0
print(paste("The kurtosis of the residuals is ", kurtosis(lm2.res)))
```

```
## [1] "The kurtosis of the residuals is  3.22530114512523"
```

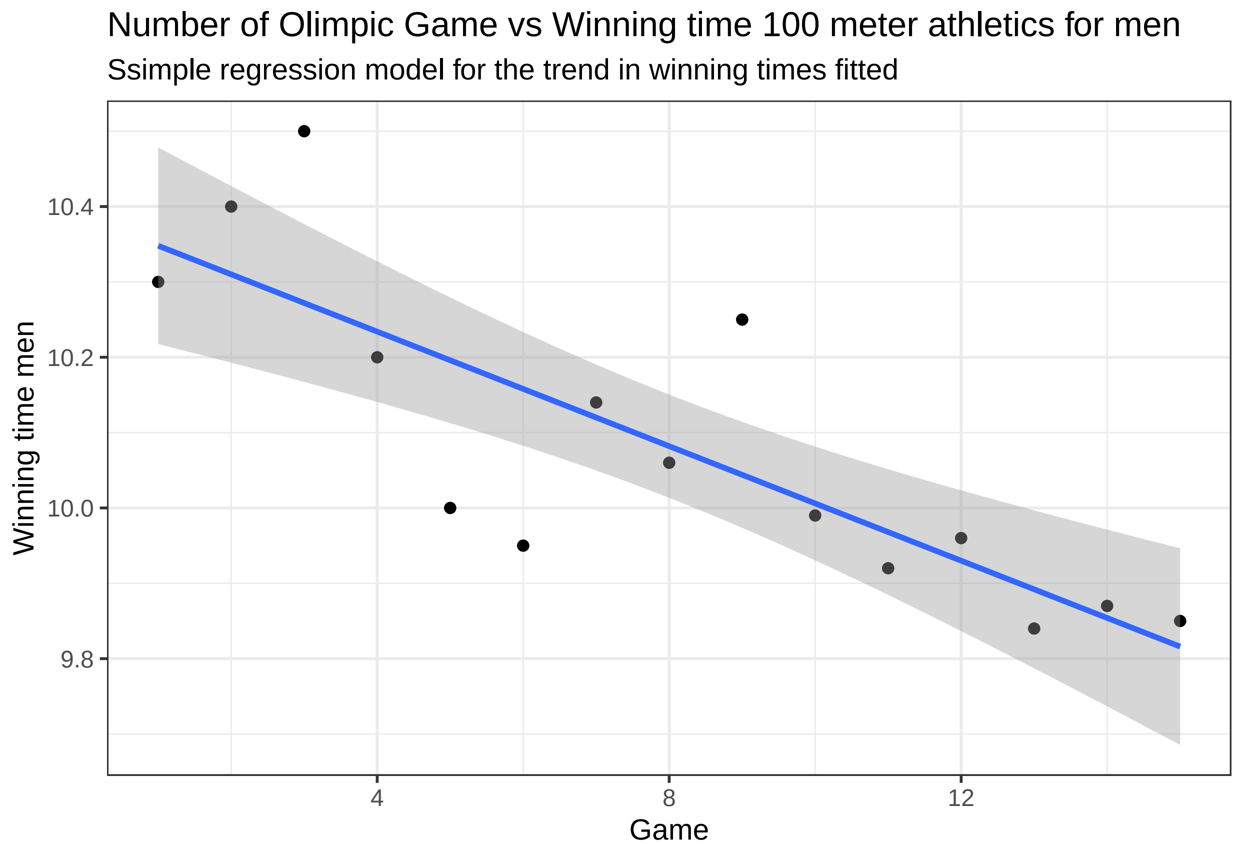
```r
#Expected 3
```

These values come close to the standard normal distribution. This concludes our first illustration of the simple regression model.

We now turn to our second illustration, where we consider the winning times of the men on the 100 meter Olympics in athletics from 1948 onwards.

In the graph, you see the winning times. The line seem to slope downwards. Consider the following simple regression model, with winning time as the dependent variable and the game number, measured as 1, 2, 3, to 15, as the explanatory factor. This model corresponds to a linear trend in winning times.

```r
plot3 + theme_bw()
```



Recall our regression

Regression Results

```
    Dependent variable:
---------------------------
    `Winning time men`
```

Game -0.038*** (0.007)
Constant 10.386*** (0.067)

Observations 15
R2 0.673
Adjusted R2 0.648
Residual Std. Error 0.123 (df = 13)
F Statistic 26.801*** (df = 1; 13)
== Note: *p<0.1;* ***p<0.05;***
p<0.01

You may now wonder whether a linear trend is the best explanatory variable for these winning times. The linear trend implies that in the very long run, the winning times could become zero. And this seems quite strange. Perhaps a better way to describe the winning times data is provided by a non-linear trend, for instance, an exponentially decaying function.

$$W_i = \gamma e^{\beta G_i}$$

then $\frac{W_{i+1}}{W_i} = e^{\beta(G_{i+1}-G_i)} = e^{\beta}$ So e^{β} is fixed. So we could transform this function by applying logs

$$log(W_i) = \alpha + \beta G_i + \epsilon_i$$

With $G_i = i$ and $\alpha = log(\gamma)$

This non-linear relation is transformed into the simple regression model by taking the log of winning time as a dependent variable.

```
lm1a <- lm(log(`Winning time men`) ~ Game, data = dataset3)
```

Regression Results

```
    Dependent variable:
---------------------------
  log(`Winning time men`)
```

Game -0.004*** (0.001)
Constant 2.341*** (0.007)

Observations 15
R2 0.677
Adjusted R2 0.652
Residual Std. Error 0.012 (df = 13)
F Statistic 27.215*** (df = 1; 13)
=== Note: *p<0.1;* ***p<0.05;*** p<0.01

The forecasts across the two models are very similar, so that the linear trend does not perform worse than the non-linear trend, at least, for the short run.

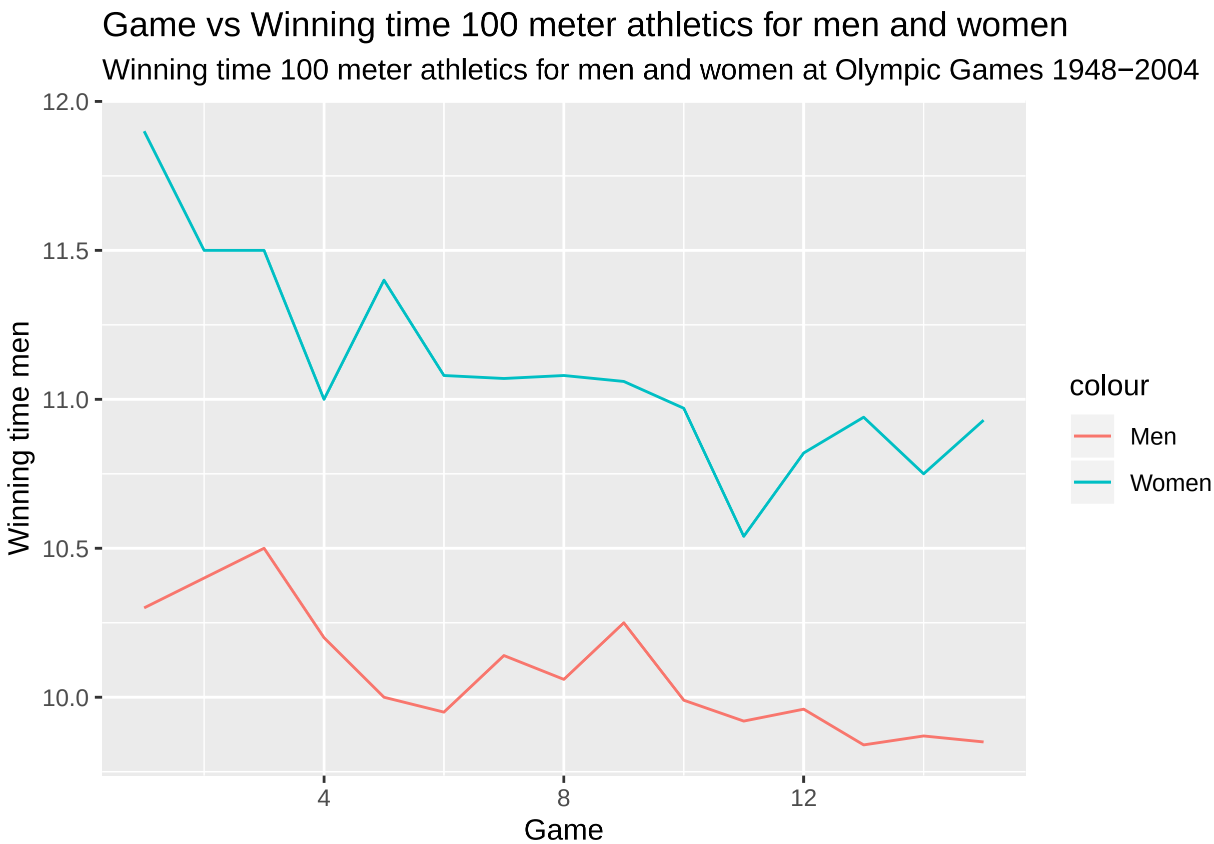
Example:

Dataset: TrainExer15 Winning time 100 meter athletics for men and women at Olympic Games 1948-2004. - Year: calendar year of Olympic Game (1948-2004) - Game: order number of game (1-15) - Winmen: winning time 100 meter athletics for men (in seconds) - Winwomen: winning time 100 meter athletics for women (in seconds)

```
dataset4 <- read_csv(
  "https://raw.githubusercontent.com/diego-eco/diego-eco.github.io/master/downloads/trainexer15.csv")
```

Previously we computed the regression coefficients a and b for two trend models, one with a linear trend and one with a nonlinear trend. In a test question, you created forecasts of the winning times men in 2008 and 2012. Of course, you can also forecast further ahead in the future. In fact, it is even possible to predict when men and women would run equally fast, if the current trends persist.

```
plot3b <- ggplot(data=dataset4, aes(x=Game,y=`Winning time men`,color='Men')) + geom_line() +
  geom_line(aes(y = `Winning time women`, color = "Women")) +
  labs(title="Game vs Winning time 100 meter athletics for men and women ",
       subtitle="Winning time 100 meter athletics for men and women at Olympic Games 1948-2004")
plot3b
```



Lets compute our linear and non-linear models:

```
lm_men_linear <- lm(`Winning time men` ~ Game, data = dataset4)
lm_men_log <- lm(log(`Winning time men`) ~ Game, data = dataset4)
lm_women_linear <- lm(`Winning time women` ~ Game, data = dataset4)
lm_women_log <- lm(log(`Winning time women`) ~ Game, data = dataset4)
```

Regression Results

```
                                   Dependent variable:
                    -------------------------------------------------------
                    Men linear Men non-linear Women linear Women non-linear
                       (1)           (2)            (3)              (4)
```

Game -0.038*** -0.004*** -0.063*** -0.006*** (0.007) (0.001) (0.012) (0.001)
Constant 10.386*** 2.341*** 11.606*** 2.452*** (0.067) (0.007) (0.111) (0.010)

Observations 15 15 15 15
R2 0.673 0.677 0.672 0.673
Adjusted R2 0.648 0.652 0.647 0.647
Residual Std. Error (df = 13) 0.123 0.012 0.204 0.018
F Statistic (df = 1; 13) 26.801*** 27.215*** 26.679*** 26.701***
==
Note: *p<0.1;* ***p<0.05;*** p<0.01

.

For example, we can also calculate some predictions with our non-linear models, remeber to exp() your results to remove the log()

```r
new.games <- data.frame(Game = c(16, 17, 18, 20, 30, 40, 50))
exp(predict(lm_men_log,new.games))
```

```
##        1        2        3        4        5        6        7
## 9.781655 9.744985 9.708452 9.635796 9.280593 8.938483 8.608985
```

```r
predict(lm_men_linear,new.games)
```

```
##     1     2     3     4     5     6     7
## 9.778 9.740 9.702 9.626 9.246 8.866 8.486
```

- Show that the linear trend model predicts equal winning times at around 2140.

From our linear models we know that for men : $W_i = 10.386 - 0.0386G_i$ and for women $W_i = 11.606 - 0.0636G_i$.

We can equialize both models and solve for G_i

$$10.386 - 0.0386G_i = 11.606 - 0.0636G_i$$

$$G_i = 48.8$$

Recall G_i counts to calendar year $1948 + (i - 1)4$ thus equal times will ocur around 2140.

- Show that the nonlinear trend model predicts equal winning times at around 2192.

Same process yields:

$$2.341 - 0.038G_i = 2.452 - 0.0056G_i$$

$$G_i = 61.7$$

Thus equal times will ocur around 2192.

- Show that the linear trend model predicts equal winning times of approximately 8.53 seconds.

In the linear time model we plug $G_i = 48.8$ resulting in $W_i = 8.53$

Both models behave “similar” in the short run, different in the long run.