Introdução

O intuito prende-se em aplicar a simulação de Monte Carlo ao modelo de regressão linear simples calculando intervalos de confiança para os coeficientes da regressão. Usa-se dados gerados para que estes exibam propriedades favoraveis à análise e contrapõem-se os coeficientes simulados com os do modelo estimado à priori com betas e x pré-definidos. Inicia-se com uma contextualização em forma de sintese sobre o modelo linear simples, as suas propriedades e hipoteses subjacentes e aborda-se os metodos de Monte Carlo. Prosseguindo com as várias estapas até à simulação em si. A ferramenta utilizada é a linguagem de progamção R e não é usado o package monte carlo ou forecast +ara esta exposição. Tentou-se reduzir ao essencial os pacotes para a elaboração da simulação.

Regressão Linear Simples

Um dos objetivos principais na análise de regressão é estudar simultaneamente vários atributos. O objetivos fundamental é saber o grau de dependência existente. Isto é, explicar as relações existentes entre determinadas variaveis atraves da construção de modelos matemáticos. Consiste assim em medir o impacto de variações de uma variavel x sobre o comportamento da variavel y. Um dos modelos mais usados e que permite estabalecer uma relacão linear fucnional entre duas variaveis é o modelo linear simples. Duas variaveis x e y relacionam-se atraves da equação

$$Y_i = \beta_0 + \beta_1 x_i + \epsilon_i, \quad \epsilon_i \sim N(0, \sigma^2)$$

em que x diz-se variavel independente,Y diz-se variavel dependente, ϵ denomina-se de termo de erro e representa todos os outros factores casuais que afectam y mas que não são explicados pela relação linear entre a varivael dependente e independente e, β_0 e β_1 que são os coeficientes do modelo.

O modelo de regressão linear simples define-se pelo seguinte conjunto de pressupostos:

- o valor médio de cada pertubação aleatória é nulo

$$E(\epsilon_i) = 0, \ \ \forall i$$

- a variância das perturbações aleatória é constante, isto é, existe homoscedasticidade

$$Var(\epsilon_i) = \sigma^2, \ \ \forall i$$

- a covariância entre perturbações aleatórias é nula. Ausência de autocorrelação.

$$Cov(\epsilon_i \epsilon_j) = 0, \ \ \forall i, j \quad i \neq j$$

Resulta daqui

$$E(Y_i) = \beta_0 + \beta_1 X_i$$

e que

$$Var(Y_i) = \sigma^2$$

isto não significa que qualquer indivíduo com valor x da variavel independente tenha o valor de variavel dependente igual a $\beta_0 + \beta_1 x$. A expressão $\beta_0 + \beta_1 x$ representa o valor médio de y dado o valor de x da variável independente.

A estimação dos coeficientes β_0 e β_1 pelo critério do Ordinary Least Square vem:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n} x_i Y_i - n\overline{x}\overline{Y}}{\sum_{i=1}^{n} x_i^2 - n\overline{x}^2}$$

e

$$\hat{\beta}_0 = \overline{Y} - \hat{\beta}_1 \overline{x}$$

Métodos de Monte Carlo

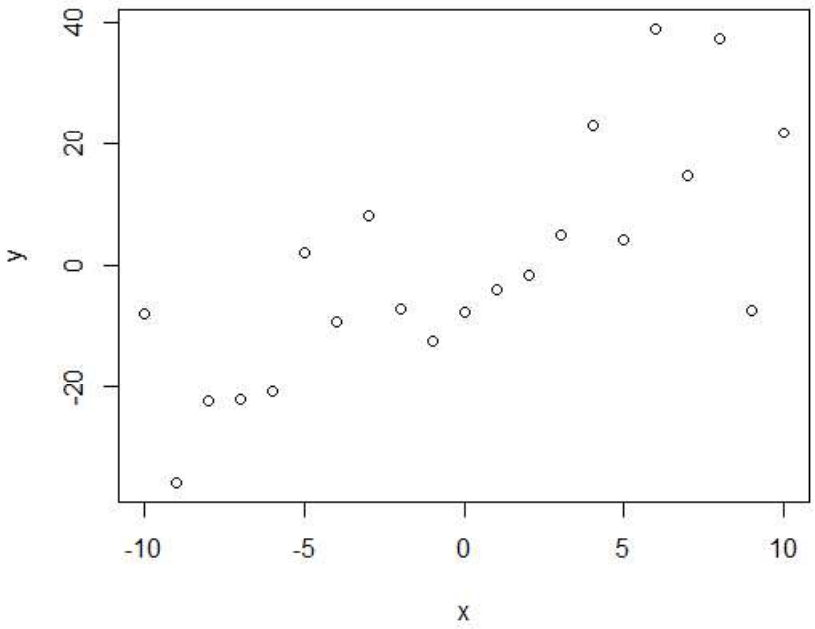
O método de simulação Monte Carlo encontra-se hoje espalhado por vários domínios das ciências exactas e sociais, podendo ser usado para descrever coisas tão díspares o comportamento da bolsa de valores ou modelagem molecular. De uma forma simplista podemos ver o método de simulação Monte Carlo como uma forma de resolver problemas recorrendo a variáveis aleatórias, cujo valor é amostrado de forma repetida ao longo de um número elevado de iterações, até obtermos um resultado com uma precisão aceitável. O método é especialmente bem adaptado para a resolução de problemas de natureza estatística, mas pode também ser usado noutros tipos de problema, especialmente quando a obtenção de uma solução exacta atraves de metodos analíticos é difícil ou até impossível de obter. A simulação Monte Carlo é sobretudo um método computacional de resolução de problemas, tendo o seu desenvolvimento e campo de aplicação, acompanhado a evolução dos computadores e poder de processamento. O aparecimento do método Monte Carlo vem muitas vezes relacionado com o projecto Manhattan de desenvolvimento da primeira bomba atómica. Um método de cálculo do número π utilizando uma experiência estocástica é talvez dos exmplos e a introdução ao método mais comum. O nome "Monte Carlo" terá sido proposto por Stanisław Ulam, um dos pais modernos do método e, é uma referência directa ao tipo de jogo efectuado no famoso casino da cidade do Mónaco.
Em termos computacionais é Iremos dessa forma tirar partido do método de Monte Carlo para simluar os coeficientes do modelo de regressão linear.
Atraves da linguagem de programação R usamos dados simulados para exemplficar de que forma é possivel simular a estimação dos coeficientes.

Especifica-se à piori β_0 e β_1 e estima-se os respetivos coeficientes atraves OLS e contrapõe-se o quão precisa a estimação é com coeficientes da população (pré definidos). Usa-se dados simulados ou gerados pelo facto das propriedades obtidas serem boas. Perante dados reais existentem problemas ou correções necessárias a ser feitas aos dados antes de iniciar a simulação. Nomeadamente é necessário verificar os pressupostos anteriormente referidos e a existente de normalidade. Não é o foco deste documento passar pelos vários passos de descrever os dados, pré-processamento dos dados e análise exploratória até chegar ao que se pretende.

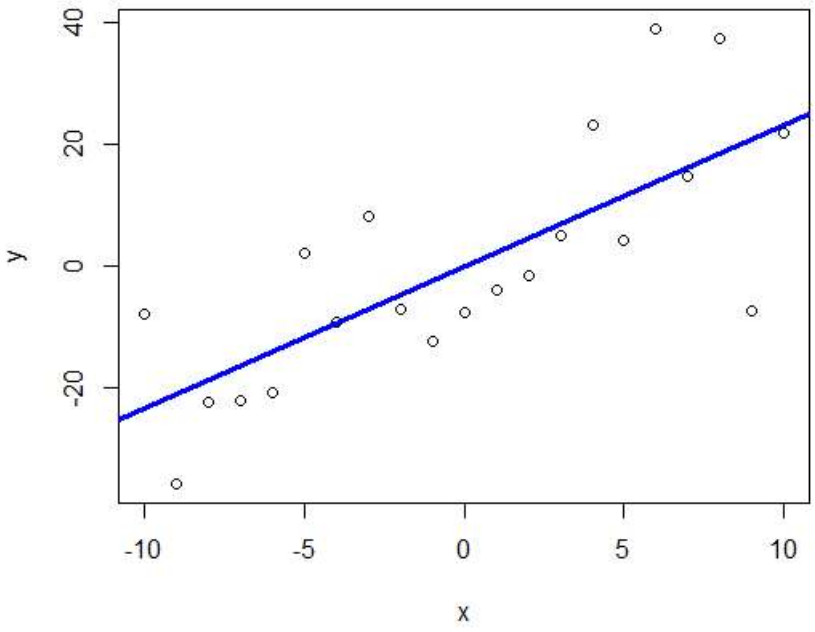
Inicia-se atribuindo a x valores entre -10 e 10, $\beta_0 = 0$, $\beta_1 = 1.5$ e y gerado por uma distribuição normal. Isto é, utiliza-se a função rnorm do R para gerar valores pseudo-aleatórios.

```
y <- rnorm(length(x), b_0 + b_1*x, 15)
```

Em termos gráficos temos



portanto graficamente a reta de regressão linear simples de y é



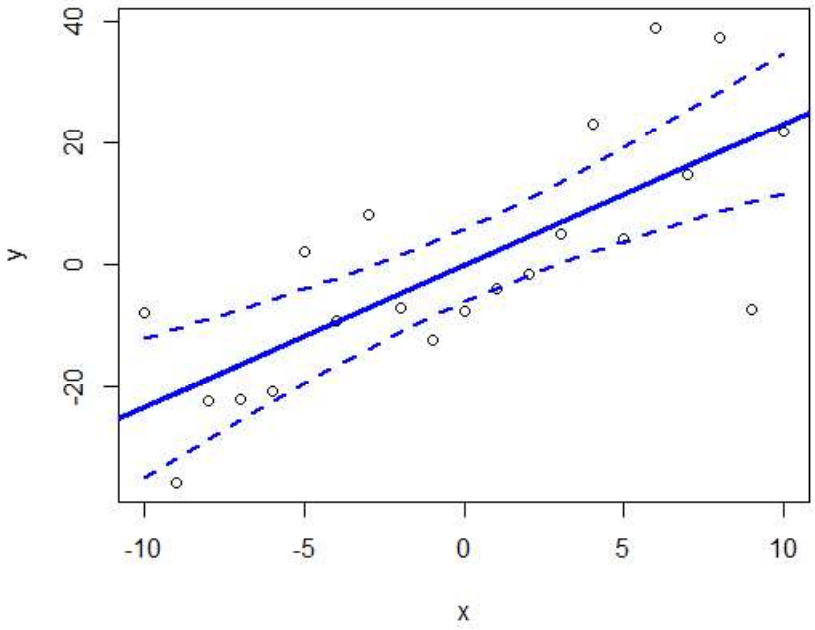
```
Call:
lm(formula = y ~ x)

Residuals:
    Min      1Q  Median      3Q     Max
-28.343  -6.547  -2.222  13.827  25.146

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -0.2224     2.8546  -0.078    0.939
x             2.3358     0.4714   4.955 8.79e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 13.08 on 19 degrees of freedom
Multiple R-squared:  0.5637,    Adjusted R-squared:  0.5408
F-statistic: 24.55 on 1 and 19 DF,  p-value: 8.795e-05
```

Torna-se possível verificar no quadro acima que as estimativas para os coeficientes em questão afastam-se bastante do que seria prevesivel ou do que se pretenderia. Devido ao reduzido número de observações é possivel deperar com este tipo de descrepância.Porém a estimativa de β_1 tem significãncia e portanto parece ajustar-se bem às observações

Segue-se com a previsão intervalar do modelo linear. Desta forma, obtem-se um treshold de uma banda superior e inferior onde os coeficientes simulados poderão eventual pertencer.

```
        fit       lwr       upr
1 -23.58010 -35.11488 -12.045323
2 -21.24433 -31.94731 -10.541347
3 -18.90855 -28.80822  -9.008879
4 -16.57278 -25.70515  -7.440398
5 -14.23700 -22.64796  -5.826041
6 -11.90123 -19.64947  -4.152984
```

graficamente temos



É agora possivel proceder à simulação de Monte Carlo dos coeficientes. Ajustamos $N = 1000$, isto é, a simulação irá iterar mil vezes a reta da regressão linear para que possamos obter os diversos betas. Seria plausivel este número de iteração ser superior, contudo para o efeito este valor parece ser razoável. Observa-se abaixo as seis primeiras simulações dos coeficientes.

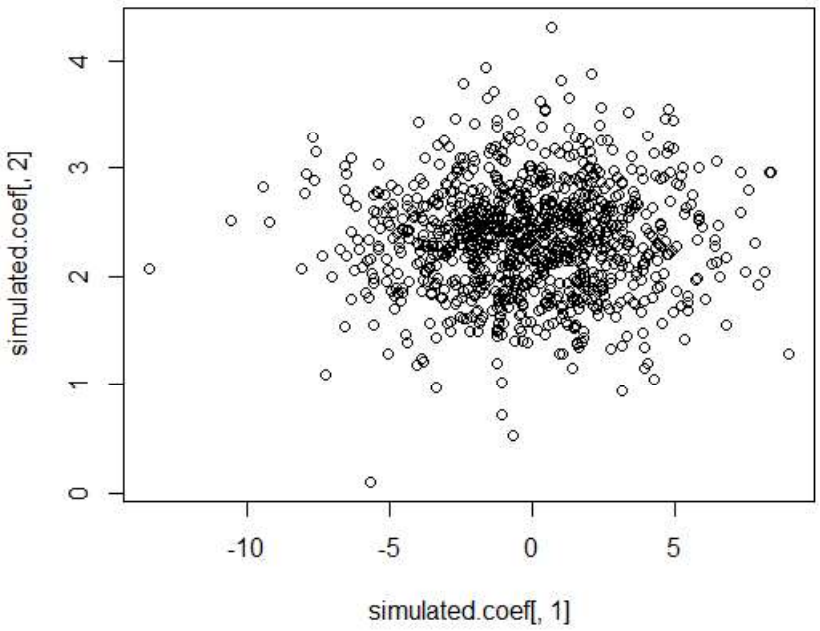
```
     (Intercept)        x
[1,]   -2.365172 2.432611
[2,]   -2.382844 1.870186
[3,]    2.736220 2.853695
[4,]   -1.945283 2.368829
[5,]    5.255348 2.934101
[6,]    8.380272 2.957854
```

Análisa-se o intervalo de confiança a 95% para o modelo e para a simulação do coeficiente β_1, respetivamente. Torna-se visivel que o beta inicialmente pré-definido está contido no intervalo.

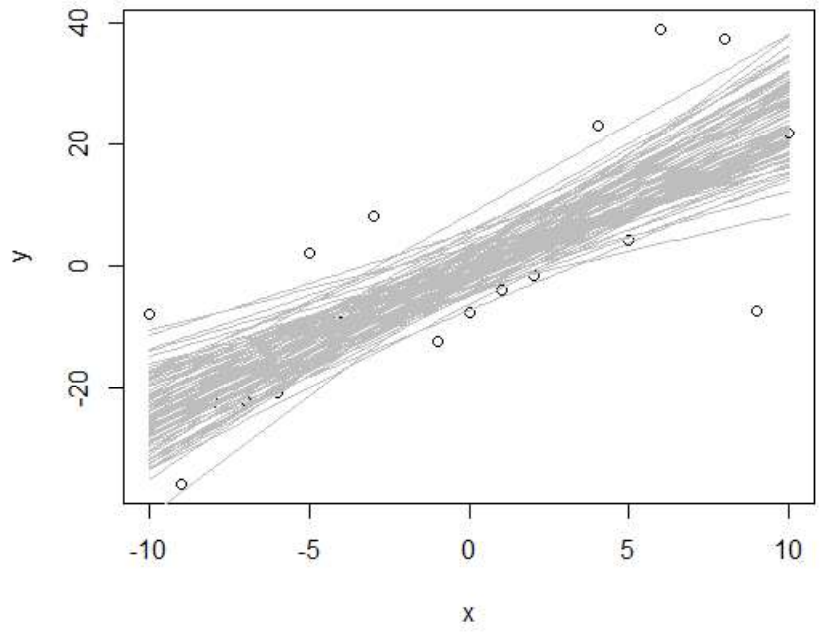
```
    2.5 %   97.5 %
1.349091 3.322459
```

```
    2.5%    97.5%
1.388414 3.348151
```

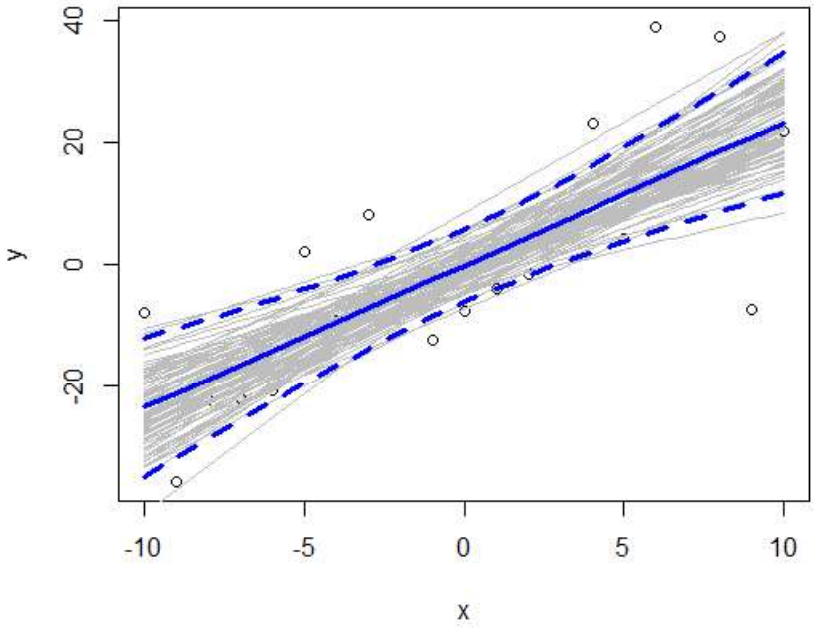
Verfica-se que os coeficientes simulados têm uma correlação bastante baixa de 0.0224 e é possivel visualizar isso mesmo graficamente.



Visualiza-se as primeiras 95 simulações dos coeficientes



Inclui-se seguidamente os Intervalos de confiança de previsão efetuados antes da simulação.



É possível obter neste intervalo um conjunto de retas de regressão linear ajustadas às observações em questão. Em que os estimadores exibissem boas propriedades, como consistência e eficiência. Mesmo com uma amostra pequena foi possivel passar a ideia de que a simualção de monte carlo é uma ferramenta poderosa, possibilita-nos obter uma perspectiva das diversas retas de regressão que podem ser geradas e ajustadas aos dados.

R code

```
    x <- -10:10
b_0 <- 0
b_1 <- 1.5


y <- rnorm(length(x), b_0 + b_1*x, 15)


plot(y~x)

lm_1 <- lm(y~x)
coef(lm_1)
confint(lm_1)
vcov(lm_1)
summary(lm_1)
abline(lm_1, lwd= 3, col="blue")


x.frame <- data.frame(x=x)

conf.int<-predict(lm_1, interval = "conf", newdata = x.frame)
head(conf.int)

matlines(x = x.frame$x, y = conf.int, lty = c(1,2,2), lwd = 2, col = "blue")

SimReg <- function(mod.input = lm_1){
  a = coef(mod.input)[1]
  b = coef(mod.input)[2]
  df.sim <- mod.input$df
  rse =summary(mod.input)$sigma
```

```
  rse.sim <- rse*sqrt(df.sim/rchisq(1, df=df.sim))
  y.sim <- rnorm(n = length(x), mean = a + b*x, sd=rse.sim)
  lm_sim <- lm(y.sim ~ x)
  coef(lm_sim)}

SimReg()



N <- 1000

simulated.coef <- replicate(N, SimReg())
simulated.coef <- t(simulated.coef)
head(simulated.coef)

sd(simulated.coef[,1])
sd(simulated.coef[,2])

summary(lm_1)$coef[,1:2]
quantile(simulated.coef[,2], c(0.025, 0.975))
confint(lm_1)[2,]

cor(simulated.coef[,1], simulated.coef[,2])
plot(simulated.coef[,1], simulated.coef[,2])

plot(y~x)
for(i in 1:95){
  curve(simulated.coef[i,1] + simulated.coef[i,2] * x, add=T, col="grey")
}
matlines(x= x.frame$x, y= conf.int, lty=c(1,2,2), lwd=3, col="blue")
```

Bibliografia

[1] Gonçalves, E.; Nogueira, E.; Rosa, A.C., *Probabilidades e Estatística para Ciências e Tecbologia*, Edições Almedina, 2016.

[2] Reis, E.; Melo, P.; Andrade, R.; Calapez, T., *Estatística Aplicada*, Volume 2, 6ª Edição, Edições Sílabo, 2018.