

E02-William Liaw

July 9, 2023

Autor: William Liaw (@willfliaw, NUSP: 11834011).

Relatório acadêmico apresentado à Escola Politécnica da Universidade de São Paulo como atividade da Disciplina Introdução à Ciência de Dados para Engenheiros (PME3573).

Este documento é componente da atividade [E02 - BH-3 Ensaio de decaimento em ar](#), do repositório [PME3573](#). Recomenda-se a leitura dos arquivos `README.md` do repositório, e `README.md` específico da atividade, para melhor noção do contexto em que se insere este trabalho acadêmico, dentre outros detalhes preliminares. Além disso, **recomenda-se fortemente** a leitura deste documento em sua versão `.ipynb`, no [GitHub](#), uma vez que lá é possível visualizar em melhor definição as tabelas e gráficos. Ainda assim, vale frisar que todos os gráficos gerados encontram-se disponíveis no diretório [images](#).

Nas seções seguintes, pode-se examinar o código utilizado para a tabulação dos dados, e, ao final, tem-se as constatações passíveis de serem feitas a partir das diversas visualizações.

1 Dados experimentais

Arquivo
<code>./dataBH3/BH_03_AIR_1.csv</code>

Ao longo deste trabalho, apelidar-se este banco de dados de **BH3**.

2 Configurações de ambiente

2.1 Estrutura do diretório

Antes de executar quaisquer células de código, a estrutura do diretório é como a seguir:

```
E02/  
  data/  
    BH_03_AIR_1.csv  
  E02 - BH-3 Ensaio de decaimento em ar.pdf  
  E02-William Liaw.ipynb  
  E02-William Liaw.pdf  
  README.md
```

2.2 Importando bibliotecas

```
[ ]: options(tidyverse.quiet = TRUE)
library(ggplot2)
library(pls, warn.conflicts = FALSE)
```

2.3 Leitura de dados

```
[ ]: dataBH3 <- read.table("./data/BH_03_AIR_1.csv", header = TRUE, sep = "\t")
```

2.4 Hiperparâmetros

```
[ ]: frames.forward = 20
```

3 Análise preliminar

```
[ ]: head(dataBH3)
```

A data.frame: 6 × 188

	Frame <int>	Time <dbl>	A01.X <dbl>	A01.Y <dbl>	A01.Z <dbl>	A02.X <dbl>	A02.Y <dbl>	A02.Z <dbl>	A03.X <dbl>
1	1	0.00	322.989	-187.229	-2600.661	324.028	-188.155	-2556.975	322.989
2	2	0.01	322.893	-187.220	-2600.629	324.130	-188.167	-2557.007	322.893
3	3	0.02	323.079	-187.218	-2600.665	324.068	-188.240	-2556.994	323.079
4	4	0.03	322.884	-187.219	-2600.632	324.176	-188.111	-2557.008	322.884
5	5	0.04	322.653	-187.326	-2600.813	324.009	-188.237	-2557.238	322.653
6	6	0.05	322.525	-187.361	-2600.855	323.760	-188.271	-2557.213	322.525

```
[ ]: class(dataBH3)
```

'data.frame'

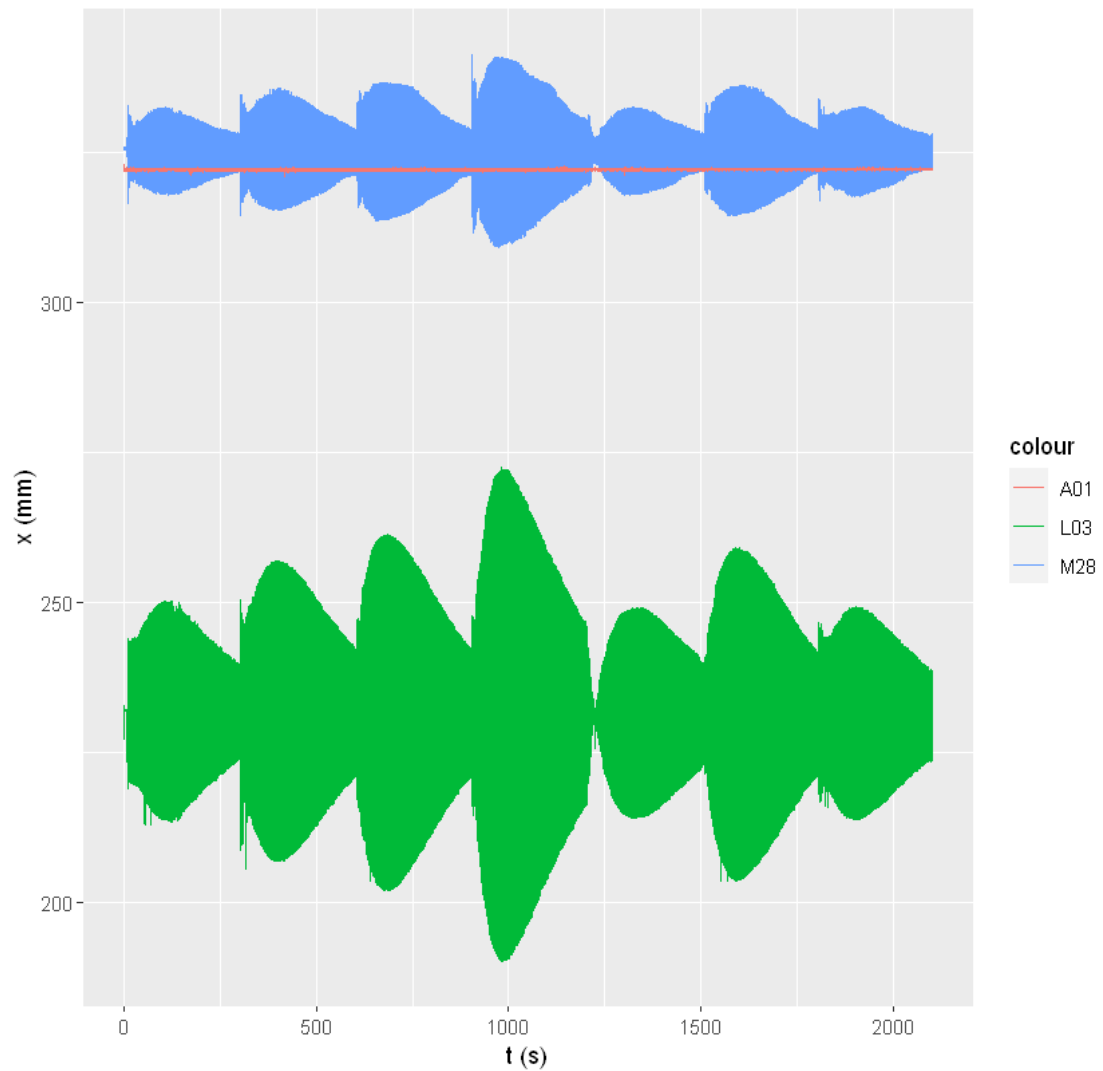
```
[ ]: names(dataBH3)
```

1. 'Frame' 2. 'Time' 3. 'A01.X' 4. 'A01.Y' 5. 'A01.Z' 6. 'A02.X' 7. 'A02.Y' 8. 'A02.Z' 9. 'A03.X'
10. 'A03.Y' 11. 'A03.Z' 12. 'M01.X' 13. 'M01.Y' 14. 'M01.Z' 15. 'M02.X' 16. 'M02.Y' 17. 'M02.Z'
18. 'M03.X' 19. 'M03.Y' 20. 'M03.Z' 21. 'M04.X' 22. 'M04.Y' 23. 'M04.Z' 24. 'M05.X' 25. 'M05.Y'
26. 'M05.Z' 27. 'M06.X' 28. 'M06.Y' 29. 'M06.Z' 30. 'M07.X' 31. 'M07.Y' 32. 'M07.Z' 33. 'M08.X'
34. 'M08.Y' 35. 'M08.Z' 36. 'M09.X' 37. 'M09.Y' 38. 'M09.Z' 39. 'M10.X' 40. 'M10.Y' 41. 'M10.Z'
42. 'M11.X' 43. 'M11.Y' 44. 'M11.Z' 45. 'M12.X' 46. 'M12.Y' 47. 'M12.Z' 48. 'M13.X' 49. 'M13.Y'
50. 'M13.Z' 51. 'M14.X' 52. 'M14.Y' 53. 'M14.Z' 54. 'M15.X' 55. 'M15.Y' 56. 'M15.Z' 57. 'M16.X'
58. 'M16.Y' 59. 'M16.Z' 60. 'M17.X' 61. 'M17.Y' 62. 'M17.Z' 63. 'M18.X' 64. 'M18.Y' 65. 'M18.Z'
66. 'M19.X' 67. 'M19.Y' 68. 'M19.Z' 69. 'M20.X' 70. 'M20.Y' 71. 'M20.Z' 72. 'M21.X' 73. 'M21.Y'
74. 'M21.Z' 75. 'M22.X' 76. 'M22.Y' 77. 'M22.Z' 78. 'M23.X' 79. 'M23.Y' 80. 'M23.Z' 81. 'M24.X'
82. 'M24.Y' 83. 'M24.Z' 84. 'M25.X' 85. 'M25.Y' 86. 'M25.Z' 87. 'M26.X' 88. 'M26.Y' 89. 'M26.Z'
90. 'M27.X' 91. 'M27.Y' 92. 'M27.Z' 93. 'M28.X' 94. 'M28.Y' 95. 'M28.Z' 96. 'M29.X' 97. 'M29.Y'
98. 'M29.Z' 99. 'M30.X' 100. 'M30.Y' 101. 'M30.Z' 102. 'M31.X' 103. 'M31.Y' 104. 'M31.Z'
105. 'M32.X' 106. 'M32.Y' 107. 'M32.Z' 108. 'M33.X' 109. 'M33.Y' 110. 'M33.Z' 111. 'M34.X'
112. 'M34.Y' 113. 'M34.Z' 114. 'M35.X' 115. 'M35.Y' 116. 'M35.Z' 117. 'M36.X' 118. 'M36.Y'

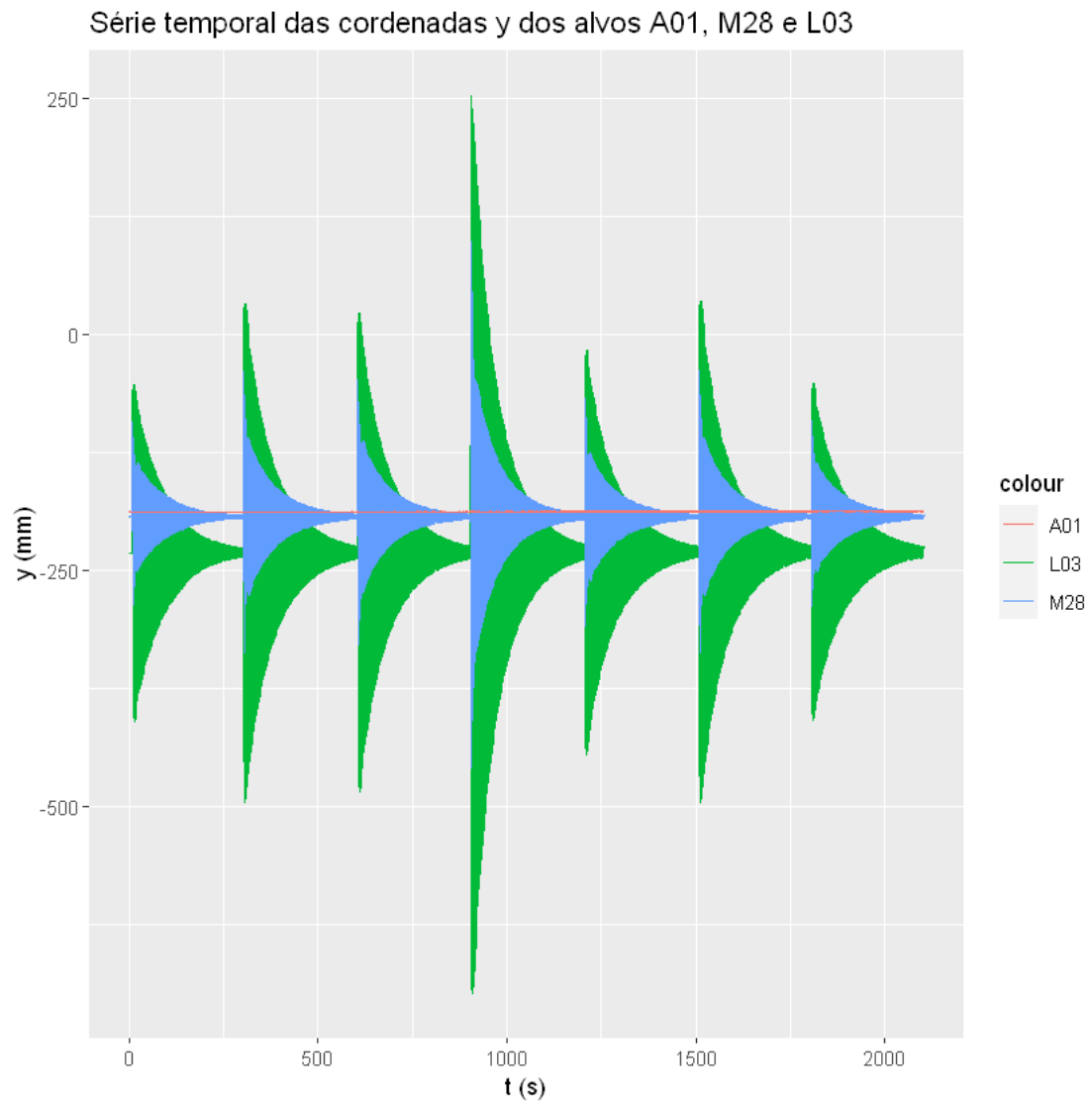
119. 'M36.Z' 120. 'M37.X' 121. 'M37.Y' 122. 'M37.Z' 123. 'M38.X' 124. 'M38.Y' 125. 'M38.Z'
 126. 'M39.X' 127. 'M39.Y' 128. 'M39.Z' 129. 'M40.X' 130. 'M40.Y' 131. 'M40.Z' 132. 'M41.X'
 133. 'M41.Y' 134. 'M41.Z' 135. 'M42.X' 136. 'M42.Y' 137. 'M42.Z' 138. 'M43.X' 139. 'M43.Y'
 140. 'M43.Z' 141. 'M44.X' 142. 'M44.Y' 143. 'M44.Z' 144. 'M45.X' 145. 'M45.Y' 146. 'M45.Z'
 147. 'M46.X' 148. 'M46.Y' 149. 'M46.Z' 150. 'M47.X' 151. 'M47.Y' 152. 'M47.Z' 153. 'M48.X'
 154. 'M48.Y' 155. 'M48.Z' 156. 'M49.X' 157. 'M49.Y' 158. 'M49.Z' 159. 'M50.X' 160. 'M50.Y'
 161. 'M50.Z' 162. 'M51.X' 163. 'M51.Y' 164. 'M51.Z' 165. 'M52.X' 166. 'M52.Y' 167. 'M52.Z'
 168. 'M53.X' 169. 'M53.Y' 170. 'M53.Z' 171. 'M54.X' 172. 'M54.Y' 173. 'M54.Z' 174. 'M55.X'
 175. 'M55.Y' 176. 'M55.Z' 177. 'M56.X' 178. 'M56.Y' 179. 'M56.Z' 180. 'L01.X' 181. 'L01.Y'
 182. 'L01.Z' 183. 'L02.X' 184. 'L02.Y' 185. 'L02.Z' 186. 'L03.X' 187. 'L03.Y' 188. 'L03.Z'

```
[ ]: ggplot(dataBH3) +
  geom_line(aes(x = Time, y = L03.X, color = "L03"), linewidth = 0.10) +
  geom_line(aes(x = Time, y = M28.X, color = "M28"), linewidth = 0.15) +
  geom_line(aes(x = Time, y = A01.X, color = "A01"), linewidth = 0.20) +
  labs(x = "t (s)", y = "x (mm)", title = "Série temporal das coordenadas x dos
↪alvos A01, M28 e L03")
```

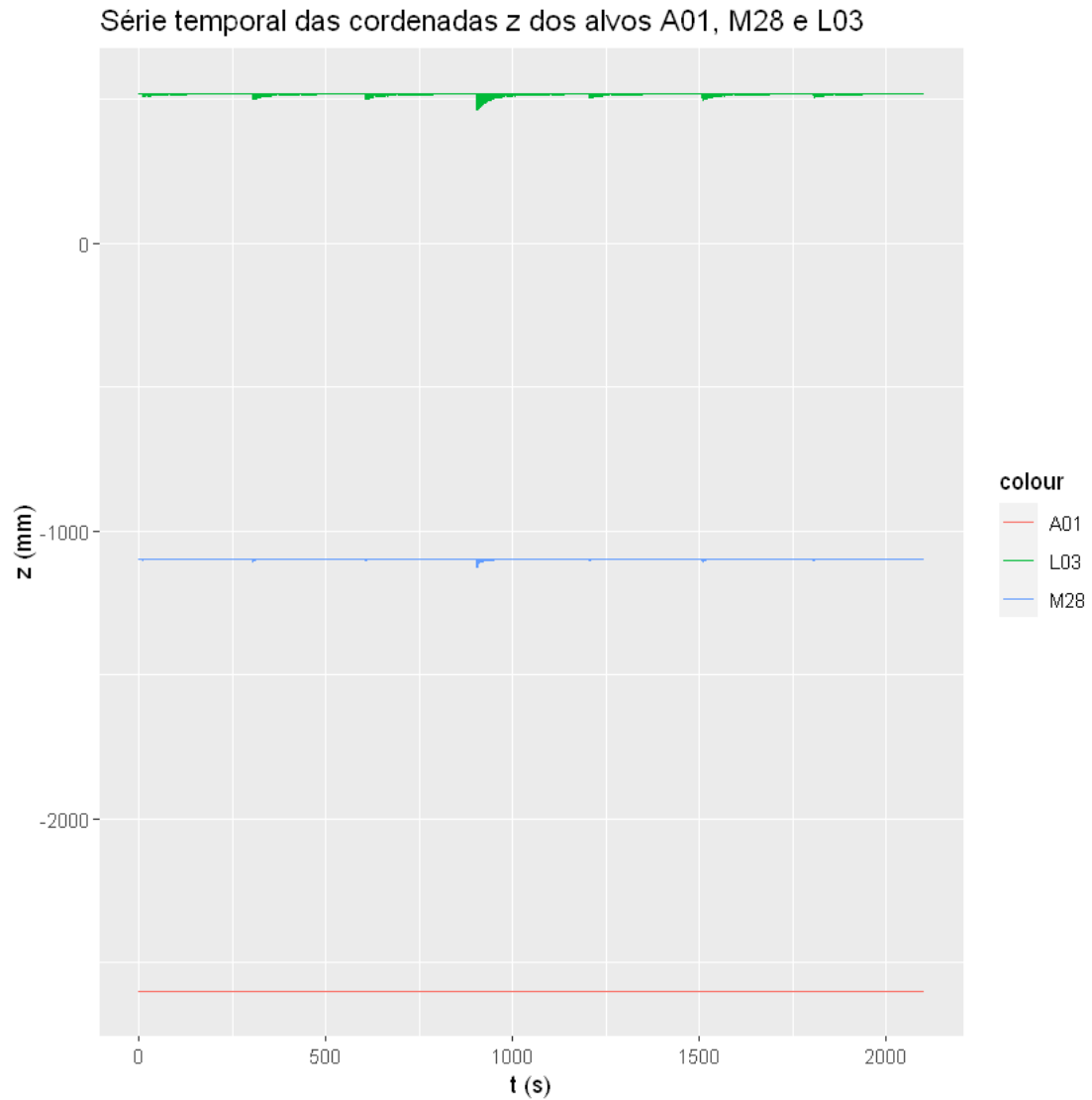
Série temporal das coordenadas x dos alvos A01, M28 e L03



```
[ ]: ggplot(dataBH3) +
  geom_line(aes(x = Time, y = L03.Y, color = "L03"), linewidth = 0.10) +
  geom_line(aes(x = Time, y = M28.Y, color = "M28"), linewidth = 0.15) +
  geom_line(aes(x = Time, y = A01.Y, color = "A01"), linewidth = 0.20) +
  labs(x = "t (s)", y = "y (mm)", title = "Série temporal das coordenadas y dos
  ↪ alvos A01, M28 e L03")
```



```
[ ]: ggplot(dataBH3) +
  geom_line(aes(x = Time, y = L03.Z, color = "L03"), linewidth = 0.10) +
  geom_line(aes(x = Time, y = M28.Z, color = "M28"), linewidth = 0.15) +
  geom_line(aes(x = Time, y = A01.Z, color = "A01"), linewidth = 0.20) +
  labs(x = "t (s)", y = "z (mm)", title = "Série temporal das coordenadas z dos alvos A01, M28 e L03")
```



4 Parte 1

Conjunto de treino simples (sem validação cruzada).

```
[ ]: n.t = 50000
      TrainingSet = 1:n.t

[ ]: x.t <- as.matrix(cbind(dataBH3[TrainingSet[-n.t], !colnames(dataBH3) %in%
  ↪("Frame", "Time")], dataBH3[TrainingSet[-1], !colnames(dataBH3) %in%
  ↪("Frame", "Time")]))
      y.t <- dataBH3[TrainingSet[-n.t] + frames.forward, "L03.Y"]
```

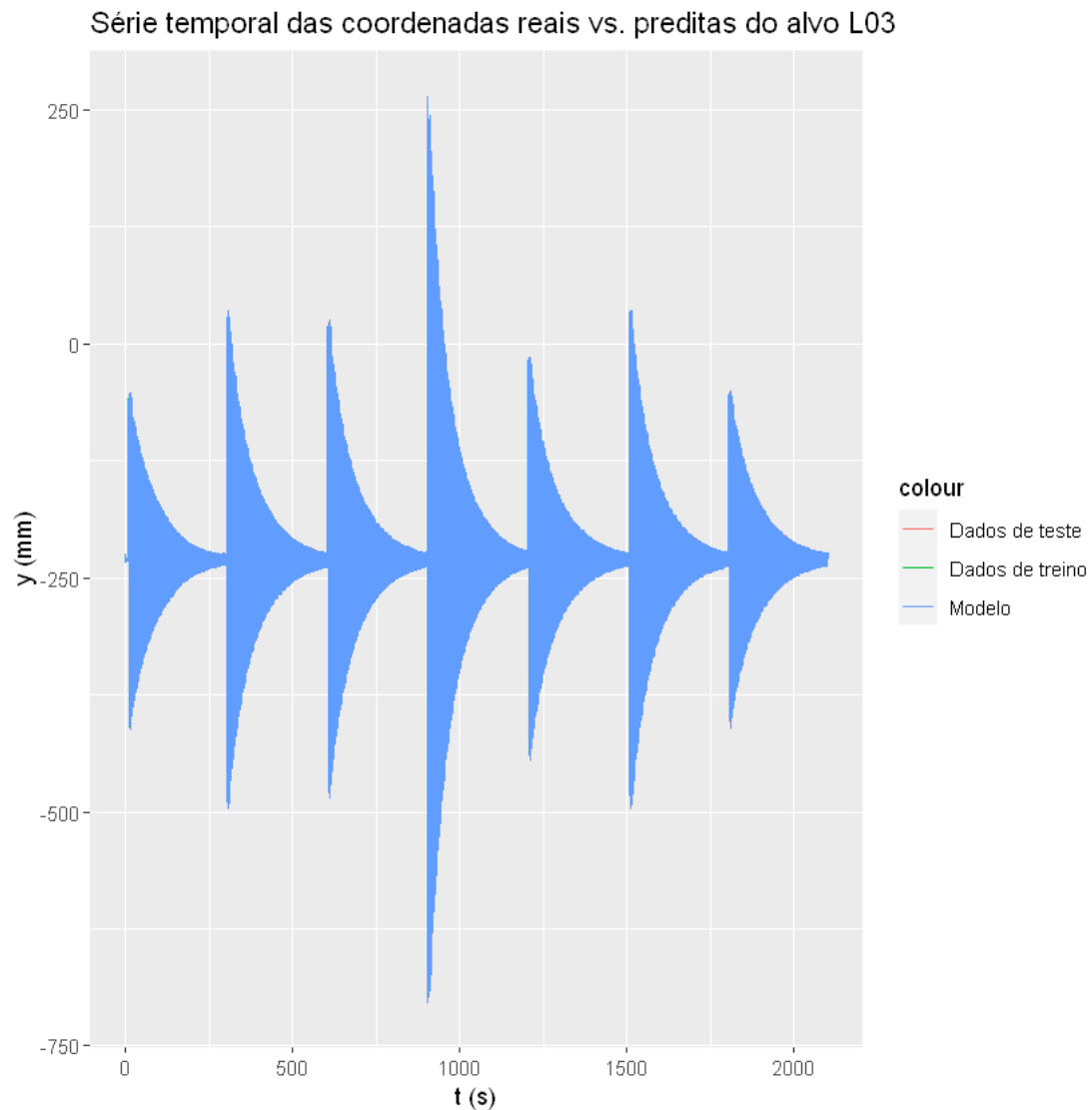
4.1 a. Regressão Linear Multivariada (MLR)

```
[ ]: L03.Y.r <- lm(y.t ~ x.t)

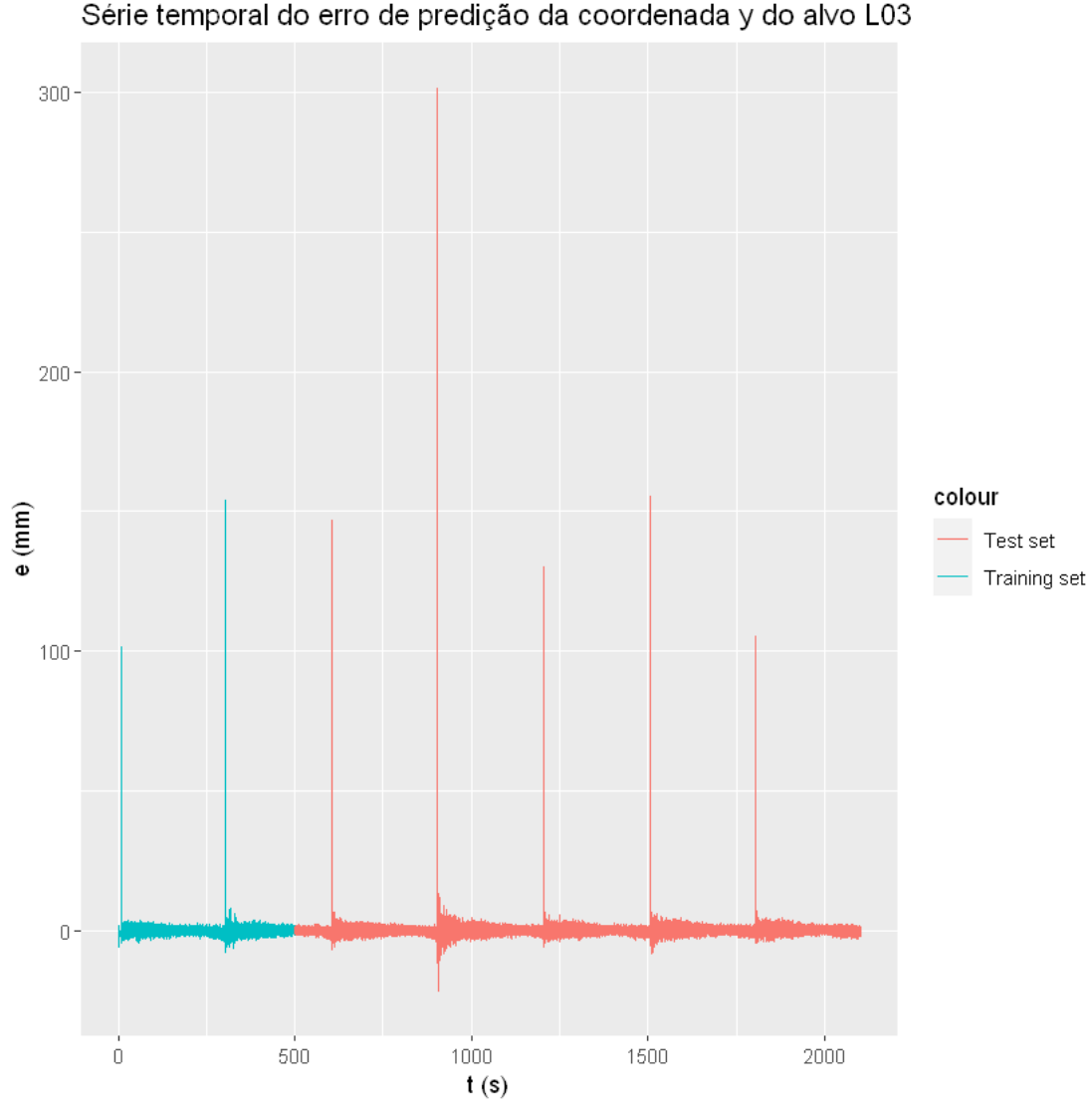
[ ]: coef <- summary(L03.Y.r)$coefficients[, "Estimate"]

L03.Y.p <- as.matrix(cbind(dataBH3[1:(nrow(dataBH3) - frames.forward), !
  ↪ colnames(dataBH3) %in% c("Frame", "Time")], dataBH3[1:(nrow(dataBH3) -
  ↪ frames.forward) + 1L, !colnames(dataBH3) %in% c("Frame",
  ↪ "Time")]))%*%coef[-1] + coef[1]
L03.Y.e <- (dataBH3[(1 + frames.forward):nrow(dataBH3), "L03.Y"] - L03.Y.p)

[ ]: ggplot() +
  geom_line(aes(x = dataBH3$Time[TrainingSet], y = dataBH3[TrainingSet, "L03.
  ↪ Y"], color = "Dados de treino"), linewidth = 0.15) +
  geom_line(aes(x = dataBH3$Time[-TrainingSet], y = dataBH3[-TrainingSet, "L03.
  ↪ Y"], color = "Dados de teste"), linewidth = 0.12) +
  geom_line(aes(x = dataBH3$Time[(1 + frames.forward):nrow(dataBH3)], y = L03.
  ↪ Y.p, color = "Modelo"), linewidth = 0.10) +
  labs(x = "t (s)", y = "y (mm)", title = "Série temporal das coordenadas
  ↪ reais vs. preditas do alvo L03")
```



```
[ ]: ggplot() +
  geom_line(aes(x = dataBH3$Time[TrainingSet + frames.forward], y = L03.Y.
    ↪e[TrainingSet], color = "Training set"), linewidth = 0.10) +
  geom_line(aes(x = dataBH3$Time[(n.t + 1 + frames.forward):nrow(dataBH3)], y =
    ↪L03.Y.e[-TrainingSet], color = "Test set"), linewidth = 0.10) +
  labs(x = "t (s)", y = "e (mm)", title = "Série temporal do erro de predição
    ↪da coordenada y do alvo L03")
```

Nota-se que o erro apresenta 7 picos, de ordem de grandeza muito superior àquela do erro médio. Infere-se que esses picos coincidem com os instantes em que se fornece estímulos externos, impactos ao sistema. É natural e esperado que o modelo não seja capaz de prever a coordenada y do alvo refletivo L03 aceitavelmente nesses instantes, pois os impactos não foram incorporados na modelagem feita, a qual utiliza informações decorrentes dos instantes n e $n + 1$, como posição e velocidade.

Para além dos 7 picos, a ordem de grandeza do erro, no geral, é cerca de poucos milímetros, o que se considera um resultado apreciável.

4.2 b. Método recursivo de mínimos quadrados (RLS)

1. $z_{k+1} \leftarrow P_k \phi(x^{(k+1)})$
2. $\gamma_{k+1} \leftarrow \frac{1}{\phi(x^{(k+1)})^* z_{k+1} + \sigma_{k+1}^2}$

3. $\theta_{k+1} \leftarrow \theta_k + \gamma_{k+1}(y^{(k+1)} - \phi(x^{(k+1)})^* \theta_k) z_{k+1}$
4. $P_{k+1} \leftarrow P_k - \gamma_{k+1} z_{k+1} z_{k+1}^*$

```
[ ]: phi <- cbind(matrix(1L, nrow = nrow(x.t)), x.t)
P0 <- diag(ncol(x.t) + 1) # m×m
theta0 <- matrix(1e-2L, nrow = ncol(x.t) + 1) # m
```

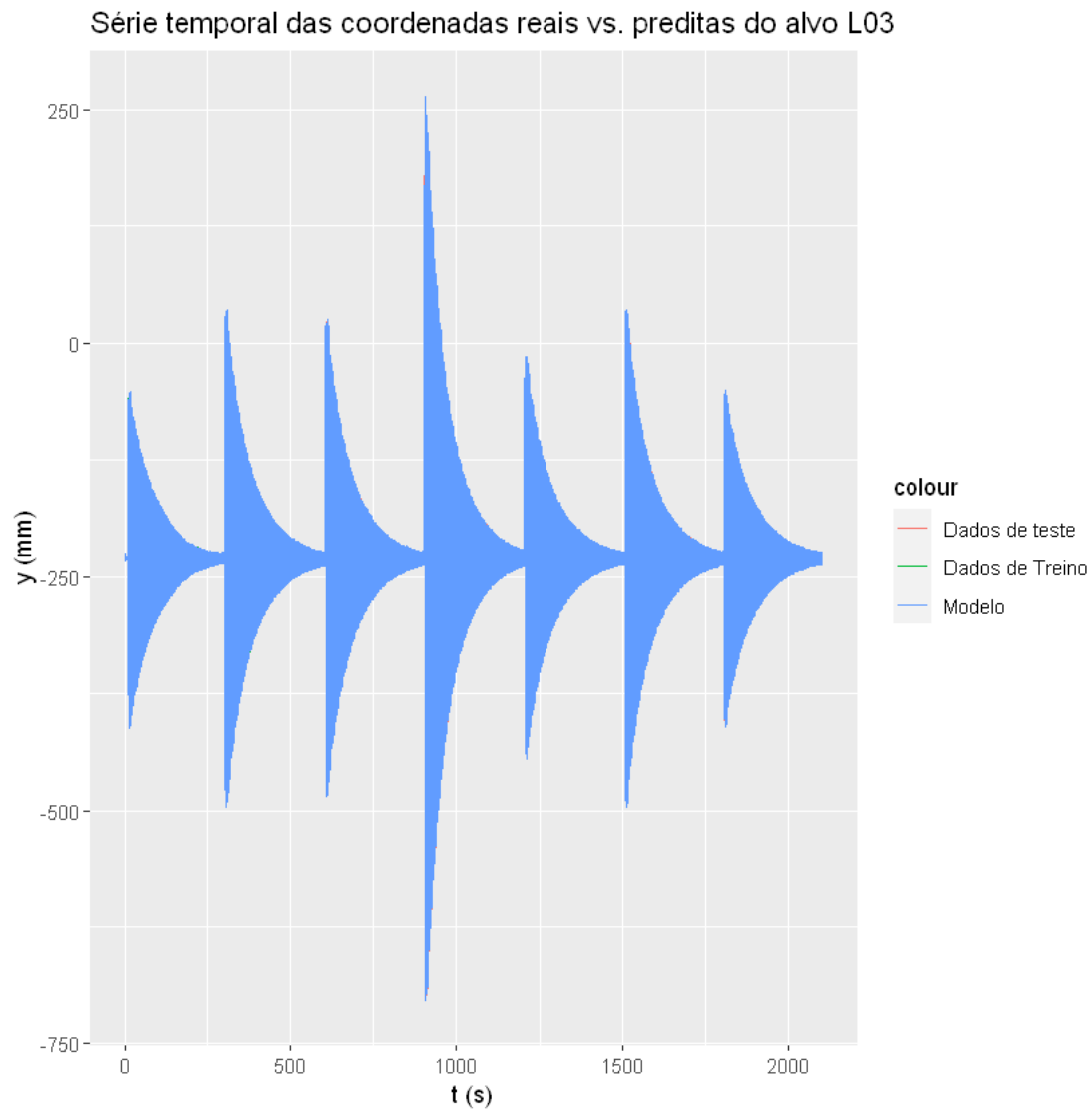
```
[ ]: z <- P0%%phi[1,]
gamma <- 1/(t(phi[1,])%*%z + 1)[1]
theta <- theta0 + gamma*(y.t[1] - t(phi[1,])%*%theta0)[1]*z
P <- P0 - gamma*z%*%t(z)

for (k in 1:(n.t - 2)) {
  z <- P%*%phi[k + 1,]
  gamma <- 1/(t(phi[k + 1,])%*%z + 1)[1]
  theta <- theta + gamma*(y.t[k + 1] - t(phi[k + 1,])%*%theta)[1]*z
  P <- P - gamma*z%*%t(z)
}

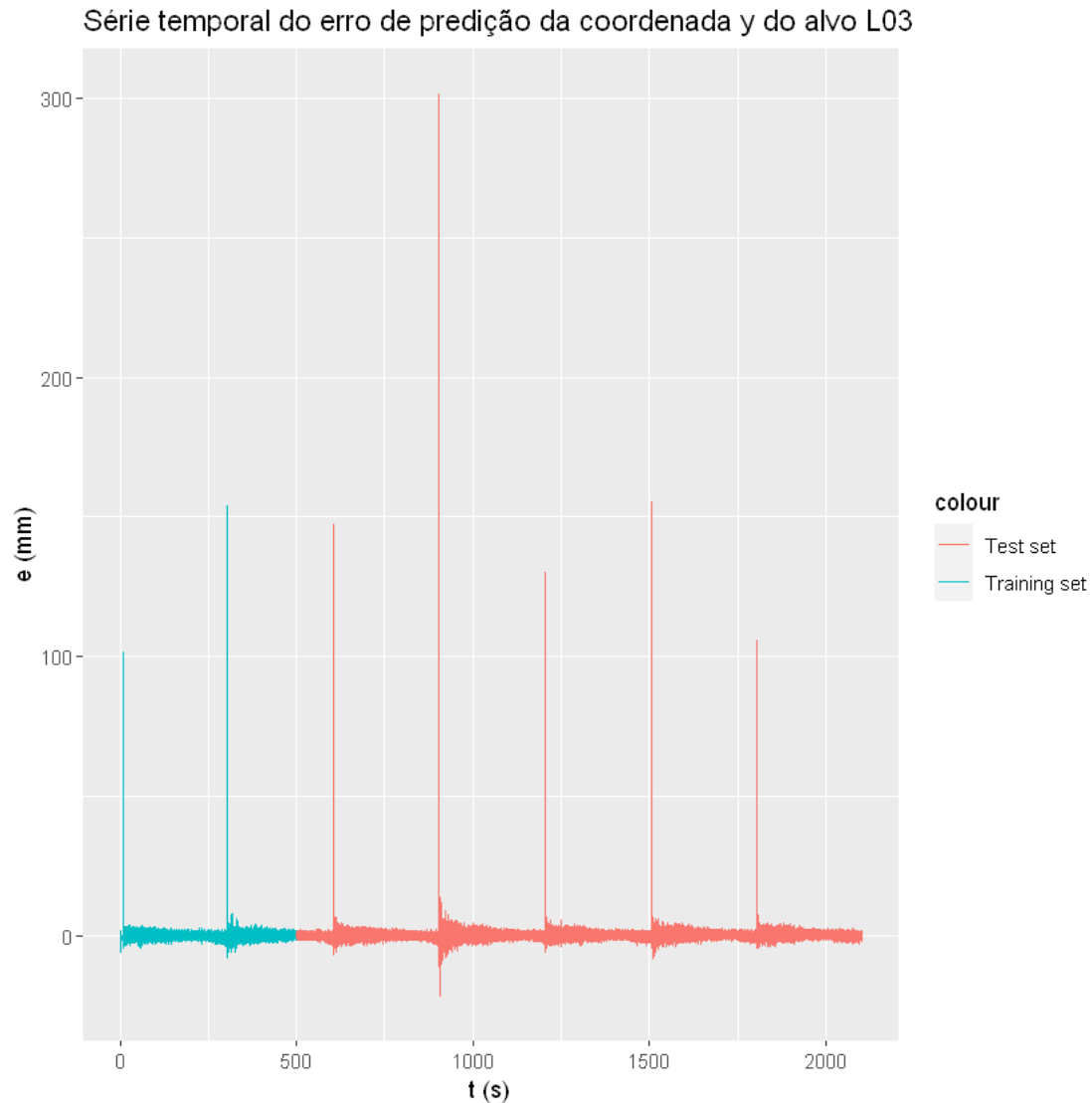
rm(phi, P0, theta0, z, gamma, P)
```

```
[ ]: L03.Y.p <- as.matrix(cbind(dataBH3[1:(nrow(dataBH3) - frames.forward), !
  ↪ colnames(dataBH3) %in% c("Frame", "Time")], dataBH3[1:(nrow(dataBH3) -
  ↪ frames.forward) + 1, !colnames(dataBH3) %in% c("Frame",
  ↪ "Time")]))%*%theta[-1] + theta[1]
L03.Y.e <- (dataBH3[(1 + frames.forward):nrow(dataBH3), "L03.Y"] - L03.Y.p)
```

```
[ ]: ggplot() +
  geom_line(aes(x = dataBH3$Time[TrainingSet], y = dataBH3[TrainingSet, "L03.
  ↪ Y"], color = "Dados de Treino"), linewidth = 0.15) +
  geom_line(aes(x = dataBH3$Time[-TrainingSet], y = dataBH3[-TrainingSet, "L03.
  ↪ Y"], color = "Dados de teste"), linewidth = 0.12) +
  geom_line(aes(x = dataBH3$Time[(1 + frames.forward):nrow(dataBH3)], y = L03.
  ↪ Y.p, color = "Modelo"), linewidth = 0.10) +
  labs(x = "t (s)", y = "y (mm)", title = "Série temporal das coordenadas
  ↪ reais vs. preditas do alvo L03")
```



```
[ ]: ggplot() +
  geom_line(aes(x = dataBH3$Time[TrainingSet + frames.forward], y = L03.Y.
    ↪e[TrainingSet], color = "Training set"), linewidth = 0.10) +
  geom_line(aes(x = dataBH3$Time[(n.t + 1 + frames.forward):nrow(dataBH3)], y =
    ↪L03.Y.e[-TrainingSet], color = "Test set"), linewidth = 0.10) +
  labs(x = "t (s)", y = "e (mm)", title = "Série temporal do erro de predição
    ↪da coordenada y do alvo L03")
```



Analogamente ao item a.: o erro apresenta 7 picos, de ordem de grandeza muito superior àquela do erro médio. Infere-se que esses picos coincidem com os instantes em que se fornece estímulos externos, impactos ao sistema. É natural e esperado que o modelo não seja capaz de prever a coordenada y do alvo refletivo L03 aceitavelmente nesses instantes, pois os impactos não foram incorporados na modelagem feita, a qual utiliza informações decorrentes dos instantes n e $n + 1$, como posição e velocidade.

Para além dos 7 picos, a ordem de grandeza do erro, no geral, é cerca de poucos milímetros, o que se considera um resultado apreciável.

```
[ ]: coefs <- cbind(coef, theta)
      colnames(coefs) <- c("MLR", "RLS")

      head(coefs)
```

	MLR	RLS
(Intercept)	-1.309705e+03	0.008329324
x.tA01.X	-4.266303e-01	-0.411316325
x.tA01.Y	3.505316e-02	-0.015762368
x.tA01.Z	-3.619487e-01	-0.282309667
x.tA02.X	4.472529e-01	0.444140561
x.tA02.Y	3.159419e-01	0.309058925

A matrix: 6 × 2 of type dbl

```
[ ]: rm(coefs)
```

Constata-se diferença mínima entre os coeficientes advindos dos métodos MLR e RLS.

5 Parte 2

Normalização dos dados:

```
[ ]: ColRange <- grep("X|Y|Z", names(dataBH3))
A. <- max(abs(dataBH3$L03.Y - mean(dataBH3$L03.Y)))
mean_vec <- c()

for (col in ColRange){
  m. <- mean(dataBH3[, col])
  dataBH3[,col] <- ((dataBH3[, col] - m.)/A.)
  mean_vec <- c(mean_vec, m.)
}

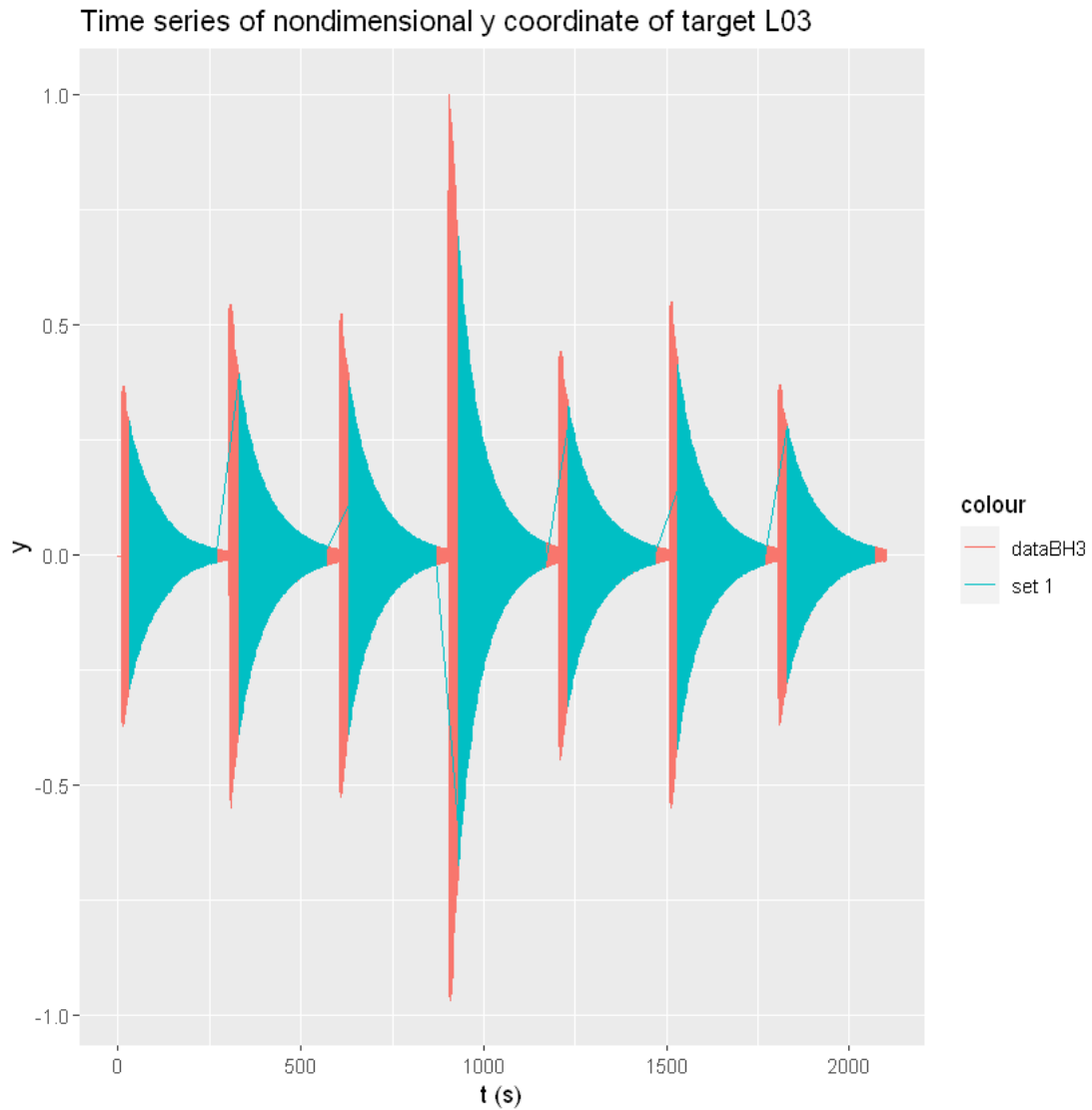
rm(ColRange, A., m.)
```

Subconjuntos para validação cruzada:

```
[ ]: subsets <- list(3000:27000, 33000:57000, 63000:87000, 93000:117000, 123000:
  ↪147000, 153000:177000, 183000:207000)
```

```
[ ]: subsets.all <- c(subsets[[1]], subsets[[2]], subsets[[3]], subsets[[4]],
  ↪subsets[[5]], subsets[[6]], subsets[[7]])
```

```
[ ]: ggplot() +
  geom_line(aes(x = dataBH3$Time, y = dataBH3$L03.Y, color = "dataBH3"),
  ↪linewidth = 0.05) +
  geom_line(aes(x = dataBH3$Time[subsets.all], y = dataBH3$L03.Y[subsets.all],
  ↪color = "set 1"), linewidth = 0.08) +
  labs(x = "t (s)", y = "y", title = "Time series of nondimensional y
  ↪coordinate of target L03")
```



μ máximo a ser avaliado:

```
[ ]: mu.max <- 10
```

5.1 Parte I: PCR

```
[ ]: MSEs <- matrix(, nrow = 0, ncol = 7)
for (mu in 1:mu.max) {
  MSEs.mu <- c()
  for (i in 1:7) {
    TrSet <- subsets.all[!subsets.all %in% subsets[[i]]]
    TsSet <- subsets[[i]]
```

```

    Ut <- as.matrix(cbind(dataBH3[TrSet, !colnames(dataBH3) %in% c("Frame", "Time")], dataBH3[TrSet + 1, !colnames(dataBH3) %in% c("Frame", "Time")]))
    K <- (t(Ut)%*%Ut)/length(TrSet)
    rm(Ut)
    evp.K <- eigen(K)

    ROM.PCA.L03.Y <- lm(dataBH3[TsSet + frames.forward, "L03.Y"] ~ as.matrix(cbind(dataBH3[TsSet, !colnames(dataBH3) %in% c("Frame", "Time")], dataBH3[TsSet + 1, !colnames(dataBH3) %in% c("Frame", "Time")]))%*%as.matrix(evp.K$vectors[, 1:mu]))

    MSE <- mean((dataBH3[TsSet + frames.forward, "L03.Y"] - predict(ROM.PCA.L03.Y))^2)

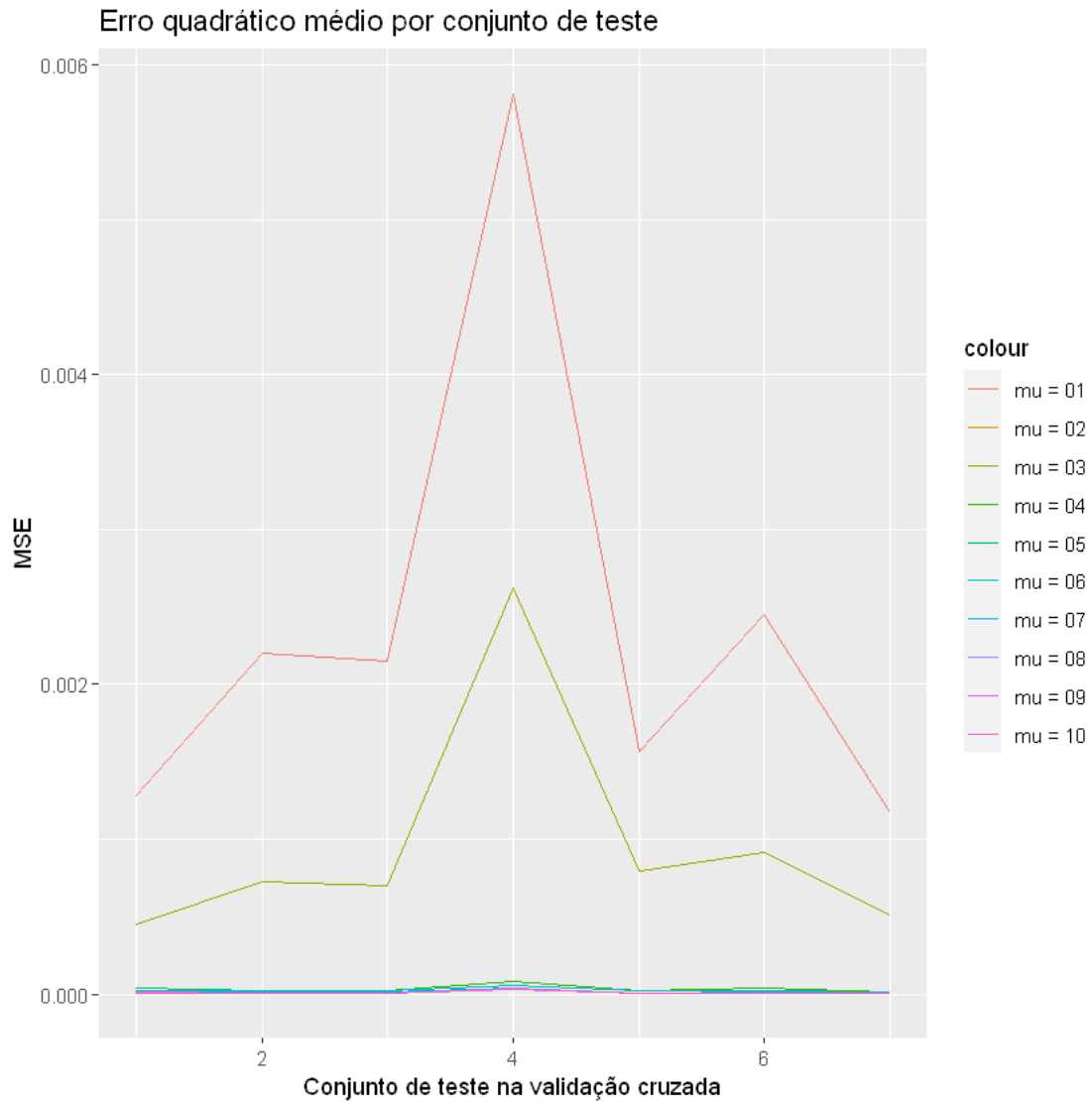
    MSEs.mu <- c(MSEs.mu, MSE)
  }
  MSEs <- rbind(MSEs, t(MSEs.mu))
}

```

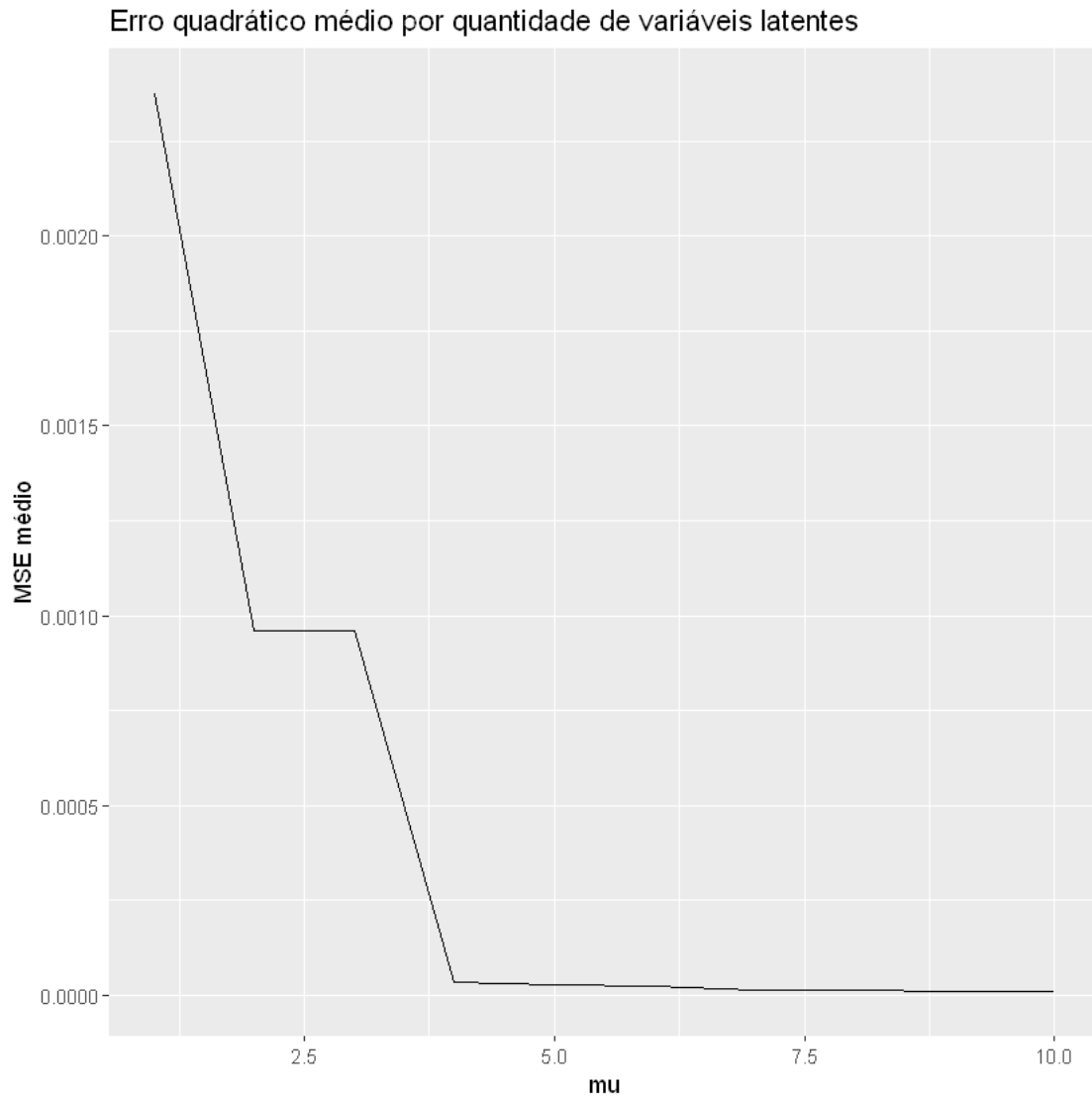
```

[ ]: ggplot() +
  geom_line(aes(x = 1:7, y = MSEs[1,], color = "mu = 01"), linewidth = 0.5) +
  geom_line(aes(x = 1:7, y = MSEs[2,], color = "mu = 02"), linewidth = 0.5) +
  geom_line(aes(x = 1:7, y = MSEs[3,], color = "mu = 03"), linewidth = 0.5) +
  geom_line(aes(x = 1:7, y = MSEs[4,], color = "mu = 04"), linewidth = 0.5) +
  geom_line(aes(x = 1:7, y = MSEs[5,], color = "mu = 05"), linewidth = 0.5) +
  geom_line(aes(x = 1:7, y = MSEs[6,], color = "mu = 06"), linewidth = 0.5) +
  geom_line(aes(x = 1:7, y = MSEs[7,], color = "mu = 07"), linewidth = 0.5) +
  geom_line(aes(x = 1:7, y = MSEs[8,], color = "mu = 08"), linewidth = 0.5) +
  geom_line(aes(x = 1:7, y = MSEs[9,], color = "mu = 09"), linewidth = 0.5) +
  geom_line(aes(x = 1:7, y = MSEs[10,], color = "mu = 10"), linewidth = 0.5) +
  labs(x = "Conjunto de teste na validação cruzada", y = "MSE", title = "Erro quadrático médio por conjunto de teste")

```



```
[ ]: ggplot() +
  geom_line(aes(x = 1:10, y = rowMeans(MSEs)), linewidth = 0.5) +
  labs(x = "mu", y = "MSE médio", title = "Erro quadrático médio por quantidade_
  de variáveis latentes")
```

Constata-se que o μ ótimo é 4, uma vez que para $\mu \geq 4$, tem-se uma média dos MSEs aproximadamente estacionária.

```
[ ]: l.eb <- c(1.8751, 4.69409, 7.85476, 10.9955, 14.1372)
EulerBernoulli <- function(i, s){
  cosh(s*l.eb[i]) - cos(s*l.eb[i]) - (sinh(l.eb[i]) - sin(l.eb[i])) / (cosh(l.
  ↪eb[i]) + cos(l.eb[i])) * (sinh(s*l.eb[i]) - sin(s*l.eb[i]))
}
```

5.2 Parte II: PLSR

```
[ ]: MSEs <- matrix(, nrow = mu.max, ncol = 0)
for (i in 1:7) {
  TrSet <- subsets.all[!subsets.all %in% subsets[[i]]]
  TsSet <- subsets[[i]]

  Ut <- as.matrix(cbind(dataBH3[TrSet, !colnames(dataBH3) %in% c("Frame", "Time")], dataBH3[TrSet + 1, !colnames(dataBH3) %in% c("Frame", "Time")]))
  St <- as.matrix(dataBH3[TrSet + frames.forward, !colnames(dataBH3) %in% c("Frame", "Time")])
  K <- (t(Ut)%*%Ut)/length(TrSet)
  C <- (t(Ut)%*%St)/length(TrSet)
  rm(Ut, St)
  R_0 <- sum(diag(K))

  m <- dim(C)[1]
  l <- dim(C)[2]
  P <- matrix(nrow = m, ncol = mu.max)
  Q <- matrix(nrow = l, ncol = mu.max)
  V_1 <- matrix(nrow = m, ncol = mu.max)
  kp <- vector(length = mu.max)

  for (k in 1:mu.max){
    V_1[,k] <- svd(C)$u[,1]
    g <- 1 / (V_1[,k] %*% K %*% V_1[,k])
    P[,k] <- g[1] * K %*% V_1[,k]
    Q[,k] <- g[1] * t(C) %*% V_1[,k]
    K[,] <- K[,] - P[,k] %*% t(V_1[,k]) %*% K[,]
    C[,] <- C[,] - P[,k] %*% t(V_1[,k]) %*% C[,]
    kp[k] <- sum(diag(K))/R_0
  }

  U <- t(as.matrix(cbind(dataBH3[TsSet, !colnames(dataBH3) %in% c("Frame", "Time")], dataBH3[TsSet + 1, !colnames(dataBH3) %in% c("Frame", "Time")]))))
  S <- t(as.matrix(dataBH3[TsSet + frames.forward, !colnames(dataBH3) %in% c("Frame", "Time")]))
  Z <- matrix(nrow = mu.max, ncol = length(TsSet))

  ROM.myPLSR.L03.Y <- S[1,]
  MSE <- mean((ROM.myPLSR.L03.Y)^2)
  MSEs.tsSet <- c(MSE)

  for (k in 1:(mu.max - 1)) {
    Z[k,] <- t(V_1[,k]) %*% U
    U[,] <- U[,] - P[,k] %*% t(Z[k,])
    S[,] <- S[,] - Q[,k] %*% t(Z[k,])
  }
}
```

```

ROM.myPLSR.L03.Y <- S[1,]
MSE <- mean((ROM.myPLSR.L03.Y)^2)
MSEs.tsSet <- c(MSEs.tsSet, MSE)
}

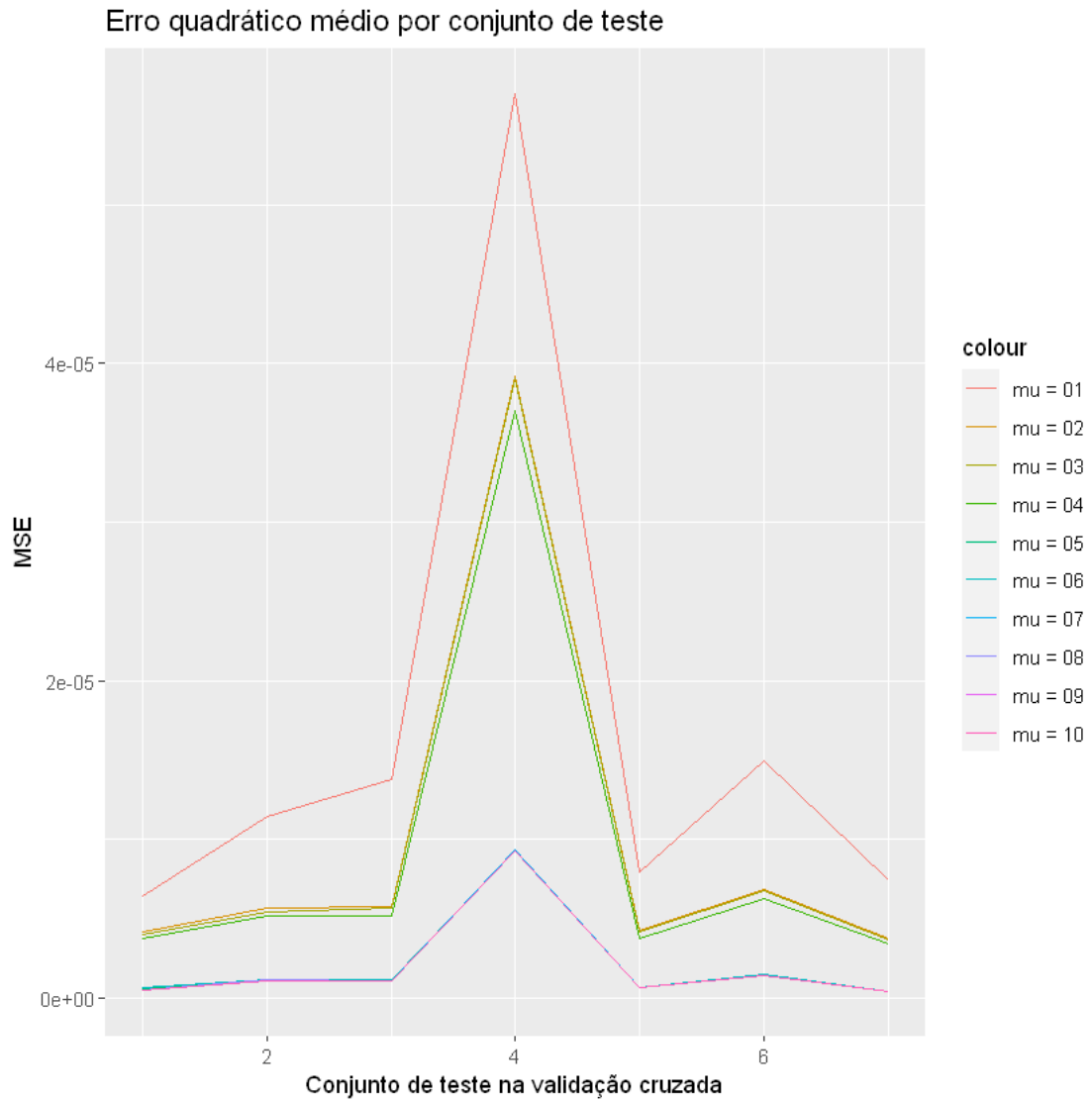
MSEs <- cbind(MSEs, MSEs.tsSet)
}

```

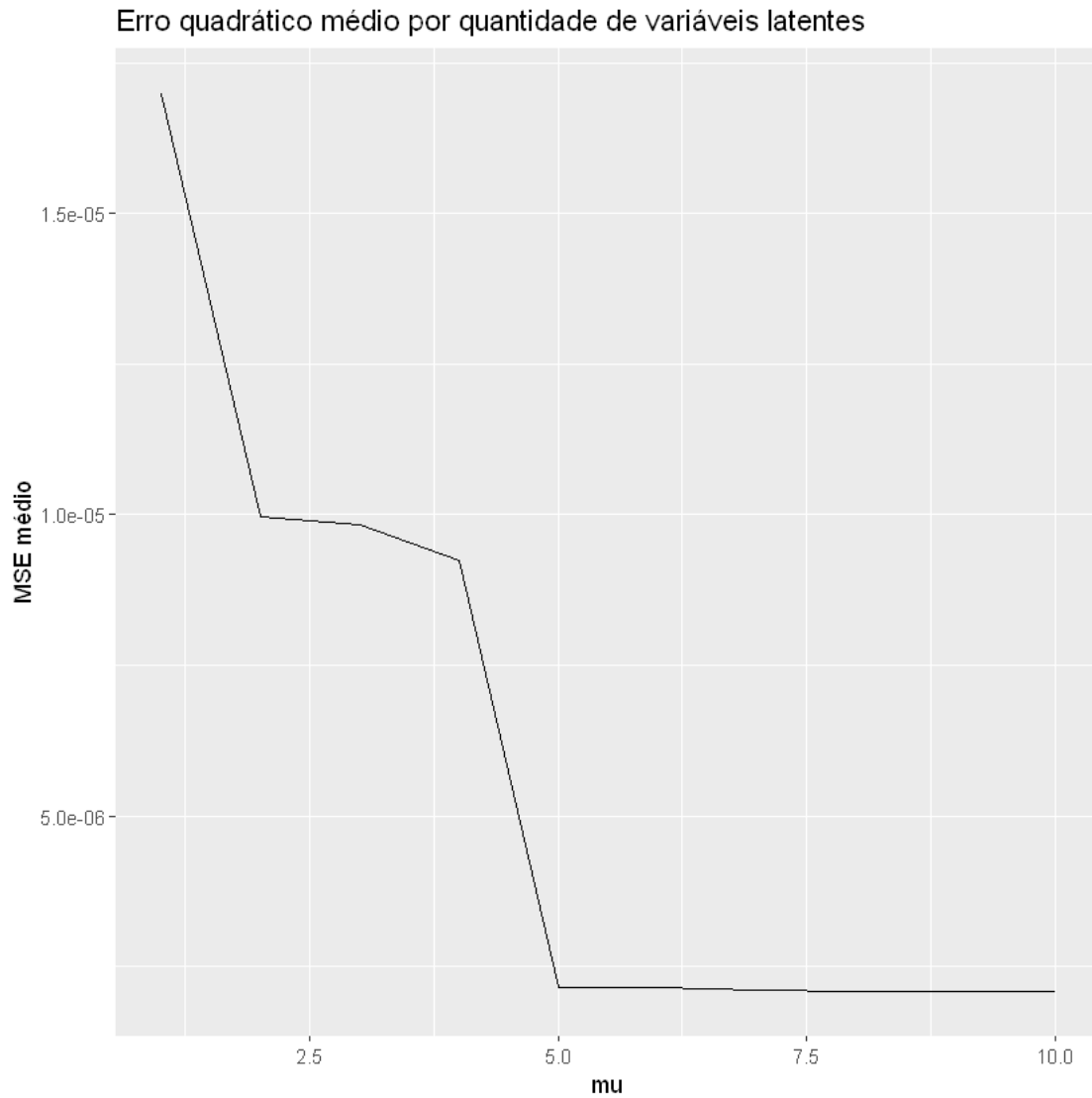
```

[ ]: ggplot() +
  geom_line(aes(x = 1:7, y = MSEs[1,], color = "mu = 01"), linewidth = 0.5) +
  geom_line(aes(x = 1:7, y = MSEs[2,], color = "mu = 02"), linewidth = 0.5) +
  geom_line(aes(x = 1:7, y = MSEs[3,], color = "mu = 03"), linewidth = 0.5) +
  geom_line(aes(x = 1:7, y = MSEs[4,], color = "mu = 04"), linewidth = 0.5) +
  geom_line(aes(x = 1:7, y = MSEs[5,], color = "mu = 05"), linewidth = 0.5) +
  geom_line(aes(x = 1:7, y = MSEs[6,], color = "mu = 06"), linewidth = 0.5) +
  geom_line(aes(x = 1:7, y = MSEs[7,], color = "mu = 07"), linewidth = 0.5) +
  geom_line(aes(x = 1:7, y = MSEs[8,], color = "mu = 08"), linewidth = 0.5) +
  geom_line(aes(x = 1:7, y = MSEs[9,], color = "mu = 09"), linewidth = 0.5) +
  geom_line(aes(x = 1:7, y = MSEs[10,], color = "mu = 10"), linewidth = 0.5) +
  labs(x = "Conjunto de teste na validação cruzada", y = "MSE", title = "Erro_
  quadrático médio por conjunto de teste")

```



```
[ ]: ggplot() +
  geom_line(aes(x = 1:10, y = rowMeans(MSEs)), linewidth = 0.5) +
  labs(x = "mu", y = "MSE médio", title = "Erro quadrático médio por quantidade_
  de variáveis latentes")
```



Constata-se que o μ ótimo é 5, uma vez que para $\mu \geq 5$, tem-se uma média dos MSEs aproximadamente estacionária.

Pelos resultados, conseguiu-se um modelo com menor MSE médio, ao longo da validação cruzada, pelo método de PLSR.