

ANÁLISE DOS DADOS COMPOSICIONAIS DE Bassoí

Formando uma composição

Carregando o pacote *compositions*

```
> require(compositions)
```

PREPARANDO OS DADOS PARA ANÁLISE

Lendo os dados

```
> dados <- read.table("DadosBassoí.txt",head=TRUE)
> # outra maneira de ler os dados seria
> # dados <- read.csv('DadosBassoí.csv',head=TRUE)
> options(width=80)
> head(dados)
```

	Linha	Coluna	Cota	AGrossa	Silte	Argila	pHAgua	pHKCl	Ca	Mg	K	Al	H
1	1	1	578.295	9	26	43	5.8	4.9	3.6	0.8	0.50	0	3.1
2	1	2	578.460	9	26	42	5.9	4.9	3.4	0.8	0.44	0	3.2
3	1	3	578.491	9	25	41	5.9	4.9	3.7	0.9	0.59	0	2.5
4	1	4	578.699	11	28	40	5.8	4.9	3.7	0.8	0.52	0	3.5
5	1	5	578.749	9	27	41	5.8	5.0	4.2	0.9	0.56	0	3.4
6	1	6	578.726	8	27	43	6.1	5.1	4.1	0.9	0.56	0	3.1

	C	N	CTC	S	V	M	NC	CEC	CN
1	1.2	0.12	8.00	4.90	61.250	0	1.050	4.90	10
2	1.1	0.12	7.84	4.64	59.184	0	1.272	4.64	9
3	1.2	0.13	7.69	5.19	67.490	0	0.289	5.19	9
4	1.3	0.12	8.52	5.02	58.920	0	1.416	5.02	10
5	1.4	0.13	9.06	5.66	62.472	0	1.023	5.66	10
6	1.2	0.13	8.66	5.56	64.203	0	0.753	5.56	9

Criando a coluna Soma que contém a soma das colunas AGrossa Silte e Argila

```
> options(width = 90)
> dadosC <- with(dados, matrix(c(Linha, Coluna, Cota, AGrossa, Silte,
+   Argila, Soma = (AGrossa + Silte + Argila), pHAgua, pHKCl, Ca, Mg,
+   K, Al, H, C, N, CTC, S, V, M, NC, CEC, CN), nrow = 250))
```

Transformando dadosC em data.frame

```

> dadosC <- as.data.frame(dadosC)
> names(dadosC)<-c('Linha','Coluna','Cota','AGrossa','Silte','Argila','Soma',
+                  'pHAgua','pHKCl','Ca','Mg','K','Al','H','C','N','CTC','S',
+                  'V','M','NC','CEC','CN')
> options(width=80)
> head(dadosC)

```

	Linha	Coluna	Cota	AGrossa	Silte	Argila	Soma	pHAgua	pHKCl	Ca	Mg	K	Al
1	1	1	578.295	9	26	43	78	5.8	4.9	3.6	0.8	0.50	0
2	1	2	578.460	9	26	42	77	5.9	4.9	3.4	0.8	0.44	0
3	1	3	578.491	9	25	41	75	5.9	4.9	3.7	0.9	0.59	0
4	1	4	578.699	11	28	40	79	5.8	4.9	3.7	0.8	0.52	0
5	1	5	578.749	9	27	41	77	5.8	5.0	4.2	0.9	0.56	0
6	1	6	578.726	8	27	43	78	6.1	5.1	4.1	0.9	0.56	0

	H	C	N	CTC	S	V	M	NC	CEC	CN
1	3.1	1.2	0.12	8.00	4.90	61.250	0	1.050	4.90	10
2	3.2	1.1	0.12	7.84	4.64	59.184	0	1.272	4.64	9
3	2.5	1.2	0.13	7.69	5.19	67.490	0	0.289	5.19	9
4	3.5	1.3	0.12	8.52	5.02	58.920	0	1.416	5.02	10
5	3.4	1.4	0.13	9.06	5.66	62.472	0	1.023	5.66	10
6	3.1	1.2	0.13	8.66	5.56	64.203	0	0.753	5.56	9

Transformando os dados de AGrossa, Silte e Areia em uma composição

```

> dadosC <-with(dadosC,matrix(c(Linha,Coluna,Cota,
+                               AGrossa=(AGrossa/Soma),Silte=(Silte/Soma),
+                               Argila=(Argila/Soma),pHAgua,pHKCl,Ca,Mg,K,
+                               Al,H,C,N,CTC,S,V,M,NC,CEC,CN),nrow=250))

```

Transformando dadosC em data.frame

```

> options(width = 90)
> dadosC <- as.data.frame(dadosC)
> names(dadosC) <- c("Linha", "Coluna", "Cota", "AGrossa", "Silte", "Argila",
+                  "pHAgua", "pHKCl", "Ca", "Mg", "K", "Al", "H", "C", "N", "CTC",
+                  "S", "V", "M", "NC", "CEC", "CN")

```

Confirmando que AGrossa, Silte e Areia formam uma composição

```
> dadosC <- with(dadosC, matrix(c(Linha, Coluna, Cota, AGrossa, Silte,
+   Argila, Soma = (AGrossa + Silte + Argila), pHAgua, pHKCl, Ca, Mg,
+   K, Al, H, C, N, CTC, S, V, M, NC, CEC, CN), nrow = 250))
```

Novamente transformando dadosC em data.frame

```
> dados <- as.data.frame(dadosC)
> names(dados) <- c("Linha", "Coluna", "Cota", "AGrossa", "Silte", "Argila",
+   "Soma", "pHAgua", "pHKCl", "Ca", "Mg", "K", "Al", "H", "C", "N",
+   "CTC", "S", "V", "M", "NC", "CEC", "CN")
> options(width = 80)
> head(dados)
```

	Linha	Coluna	Cota	AGrossa	Silte	Argila	Soma	pHAgua	pHKCl	Ca	Mg
1	1	1	578.295	0.1153846	0.3333333	0.5512821	1	5.8	4.9	3.6	0.8
2	1	2	578.460	0.1168831	0.3376623	0.5454545	1	5.9	4.9	3.4	0.8
3	1	3	578.491	0.1200000	0.3333333	0.5466667	1	5.9	4.9	3.7	0.9
4	1	4	578.699	0.1392405	0.3544304	0.5063291	1	5.8	4.9	3.7	0.8
5	1	5	578.749	0.1168831	0.3506494	0.5324675	1	5.8	5.0	4.2	0.9
6	1	6	578.726	0.1025641	0.3461538	0.5512821	1	6.1	5.1	4.1	0.9

	K	Al	H	C	N	CTC	S	V	M	NC	CEC	CN
1	0.50	0	3.1	1.2	0.12	8.00	4.90	61.250	0	1.050	4.90	10
2	0.44	0	3.2	1.1	0.12	7.84	4.64	59.184	0	1.272	4.64	9
3	0.59	0	2.5	1.2	0.13	7.69	5.19	67.490	0	0.289	5.19	9
4	0.52	0	3.5	1.3	0.12	8.52	5.02	58.920	0	1.416	5.02	10
5	0.56	0	3.4	1.4	0.13	9.06	5.66	62.472	0	1.023	5.66	10
6	0.56	0	3.1	1.2	0.13	8.66	5.56	64.203	0	0.753	5.56	9

Criando o arquivo com os dados composicionais

```
> options(width = 90)
> dados <- with(dados, matrix(c(AGrossa = (AGrossa * 100), Silte = (Silte *
+   100), Argila = (Argila * 100)), nrow = 250))
> head(dados)
```

	[,1]	[,2]	[,3]
[1,]	11.53846	33.33333	55.12821
[2,]	11.68831	33.76623	54.54545
[3,]	12.00000	33.33333	54.66667

```
[4,] 13.92405 35.44304 50.63291
[5,] 11.68831 35.06494 53.24675
[6,] 10.25641 34.61538 55.12821
```

INICIANDO ANÁLISE COMPOSICIONAL

Na classe `acomp` utilizada a seguir as quantidades individuais são parte de um todo e os dados são analisados na geometria relativa.

```
> comp <- acomp(dados)
> names(comp) <- c("AGrossa", "Silte", "Argila")
> head(comp)

[1] 0.1153846 0.1168831 0.1200000 0.1392405 0.1168831 0.1025641

> par(mar = c(5, 0.5, 0, 0.5), mgp = c(2, 0.8, 0))
> plot(comp)
```

Vendo os dados como quantidades

```
> quant <- aplus(dados)
> names(quant) <- c("AGrossa", "Silte", "Argila")
> head(quant)

[1] 11.53846 11.68831 12.00000 13.92405 11.68831 10.25641
```

Construindo uma matriz de diagramas de dispersão

```
> plot(quant)
```

Incluindo a média geométrica e a região 2 e 4-sigma no diagrama ternário

```
> plot(comp)
> plot(mean(comp), add = T, pch = 20, col = "red")
> ellipses(mean(comp), var(comp), col = "red", r = 2)
> ellipses(mean(comp), var(comp), col = "red", r = 4)
```

O centro da distribuição é dado por

```
> mean(comp)
```

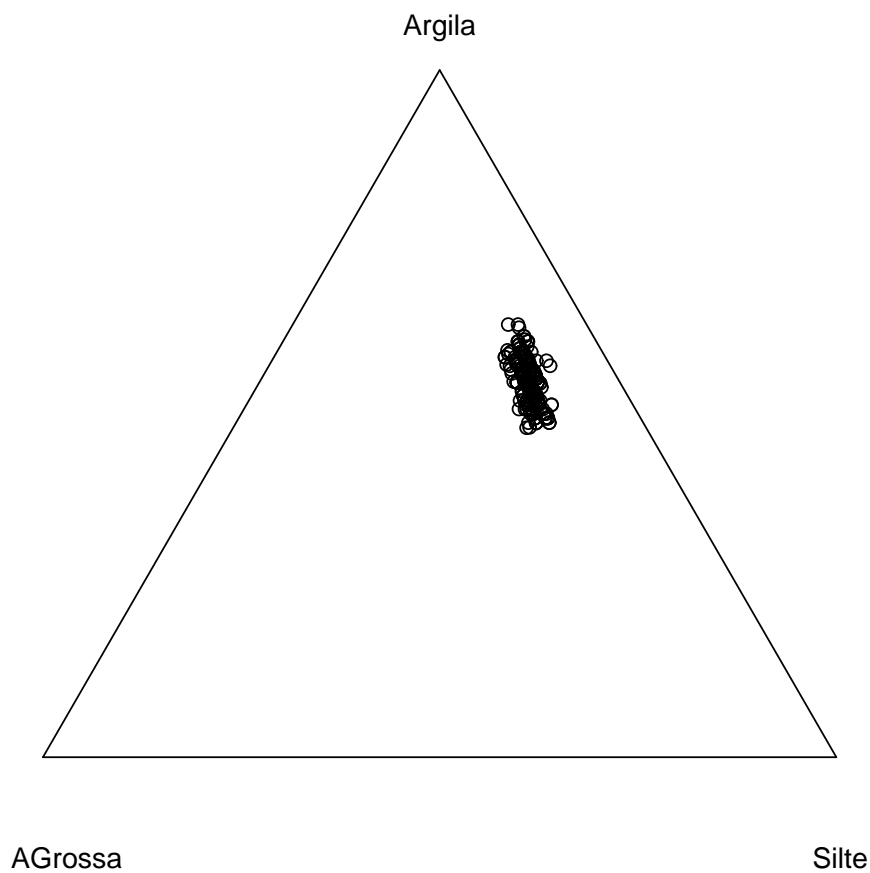


Figura 1: Diagrama Ternário para AGrossa, Silte e Argila.

```
AGrossa    Silte    Argila
0.1129220  0.3371904 0.5498876
attr("class")
[1] "acomp"
```

A variância na estrutura CLR do espaço euclidiano é

```
> var(comp)
```

```
          AGrossa          Silte          Argila
AGrossa  0.0067548139 -0.0007540116 -0.006000802
Silte    -0.0007540116  0.0032778222 -0.002523811
Argila   -0.0060008023 -0.0025238106  0.008524613
```

Obtém-se a variância métrica que nada mais é do que o traço da matriz CLR, fazendo-se

```
> mvar(comp)
```

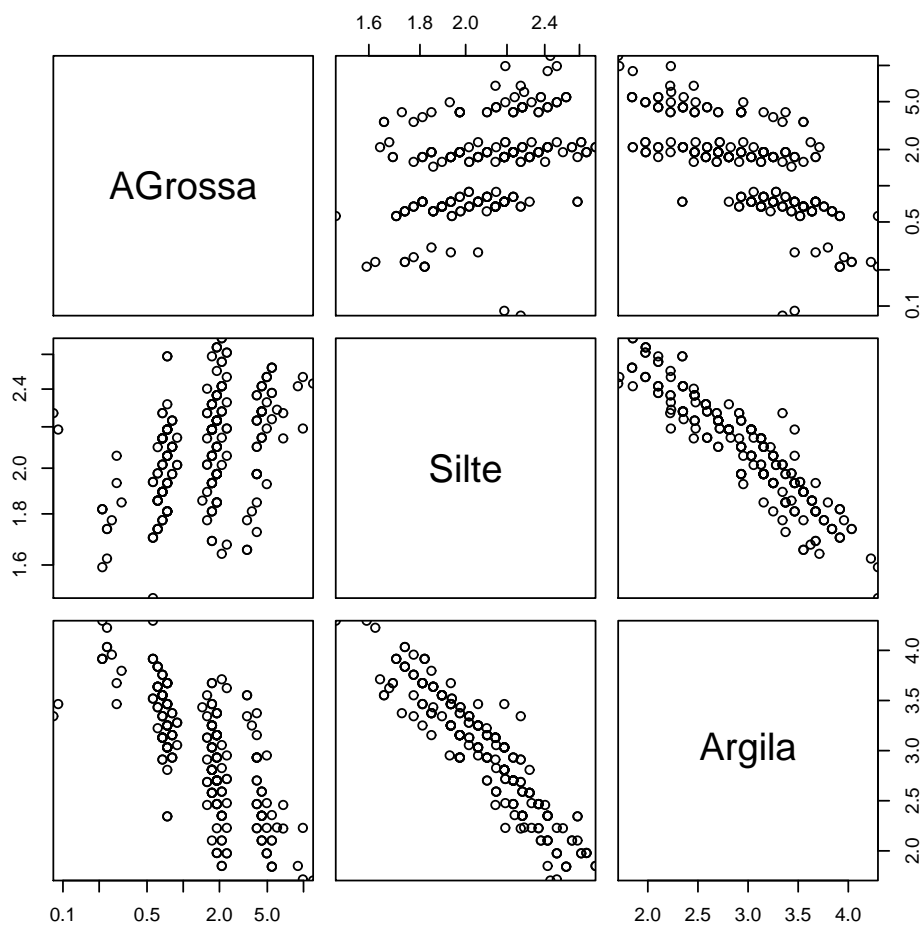


Figura 2: $AGrossa \times Silte$, $AGrossa \times Argila$ e $Silte \times Argila$.

[1] 0.01855725

O desvio padrão métrico - $\sqrt{mvar/(D-1)}$ é dado por

```
> msd(comp)
```

[1] 0.09632562

Já o desvio padrão clássico é obtido através de

```
> sd(comp)
```

```
AGrossa    Silte    Argila
0.01308933 0.02448595 0.03241931
```

Matriz variação - $var[\log(x_i/x_j)]$:

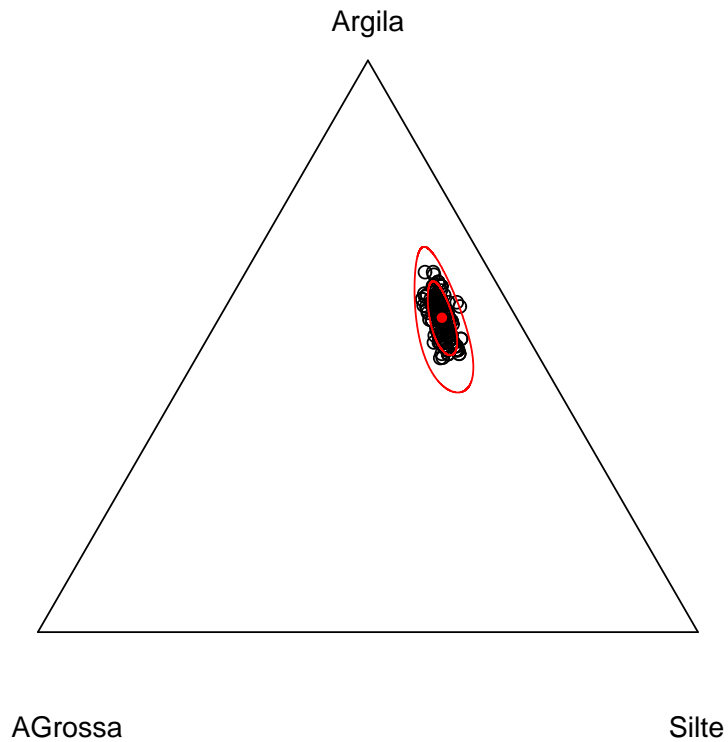


Figura 3: Diagrama ternário com região de confiança.

```
> variation(comp)
```

	AGrossa	Silte	Argila
AGrossa	0.00000000	0.01154066	0.02728103
Silte	0.01154066	0.00000000	0.01685006
Argila	0.02728103	0.01685006	0.00000000

Por fim, a variância total, a partir da variância na estrutura CLR é obtida. Para isto cria-se um vetor com 3 elementos iguais a 1 e o transposto deste

```
> x <- c(1, 1, 1)
```

```
> x
```

```
[1] 1 1 1
```

```
> xt <- t(x)
```

```
> xt
```

```
      [,1] [,2] [,3]
[1,]    1    1    1
```

e, então:

```
> options(width = 80)
> varTot <- mvar(comp) - (1/3) * (xt %*% var(comp) %*% x)
> varTot
```

```
      [,1]
[1,] 0.01855725
```