Supplementary Table 1. Taxon-level fuzzy coding and supporting source

Genus	Body size			Sediment Position						Living Habit					Feeding guild							Mobility trait			Bioturbation Type				Source
	S	M	L	ATT	BP	EPI	CSS	D3	S3	ATT	BUR	FSC	PC	TD	DF	FF	GS	OM	PR	SC	SSDF	LIM	MOB	SES	BD	BI	NB	SM	
Austruca	0	1	0	0	0	0.5	0	0.5	0	0	0.5	0.5	0	0	1	0	0	0	0	0	0	0	1	0	0.5	0	0	0.5	Mokhlesi et al. (2011); Mokhtari et al. (2015. 2016); Saher and Qureshi (2014).
Cassidula	0	0	1	0	0	1	0	0	0	0	0	1	0	0	0.5	0	0.5	0	0	0	0	0	1	0	0	0	0	1	Ariyanto et al. (2018); Zvonareva et al. (2015); Oktavia et al. (2023).
Ennucula	0	1	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	Lam-Gordillo et al. (2023).
Gelasimus	0	1	0	0	0	0.5	0	0.5	0	0	0.5	0.5	0	0	1	0	0	0	0	0	0	0	1	0	0.5	0	0	0.5	Zeil (1998); Lim (2006); Takeda et al. (2004); Kristensen and Alongi (2006); Min et al. (2023).
Metaplax	0	1	0	0	0	0.5	0	0.5	0	0	0.5	0.5	0	0	1	0	0	0	0	0	0	0	1	0	0.5	0	0	0.5	Kitaura et al. (2002); Nordhaus et al. (2011); Wang et al. (2014); Agusto et al. (2021).
Lingula	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1	Emig (1983); Hammond (1983); Savazzi (1991); Temereva and Tsitrin (2015).
Littoraria	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	1	0	0	0	1	0	Reid (1985); Alfaro (2007. 2008).

Genus	Body size			Sediment Position						Living Habit					Feeding guild							Mobility trait			Bioturbation Type				Source
	S	M	L	ATT	BP	EPI	CSS	D3	S3	ATT	BUR	FSC	PC	TD	DF	FF	GS	OM	PR	SC	SSDF	LIM	MOB	SES	BD	BI	NB	SM	
Nassarius	0	1	0	0	0	1	0	0	0	0	0.5	0.5	0	0	0	0	0	0	0.25	0.75	0	0	1	0	0	0	0	1	Aloy et al. (2011); Morton (2011); Galindo et al. (2016); Vermeij (2017).
Nereis	0	0	1	0	0.5	0.5	0	0	0	0	0	1	0	0	0.25	0	0.25	0	0.25	0.25	0	0	1	0	0.5	0	0	0.5	Lam-Gordillo et al. (2023).
Neritodryas	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	1	0	0	0	1	0	Printrakoon et al. (2008); Mustapha et al. (2021).
Olivella	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0.5	0.5	0	0	1	0	0	0	0	1	Arruda et al. (2003); Morse and Peters (2016); Checon et al. (2020).
Pirenella	0	1	0	0	0	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	Reid and Ozawa (2016); Kamimura and Tsuchiya (2004); Meziane and Tsuchiya (2000); Yadav et al. (2025).
Geloina	0	1	0	0	0	0.5	0	0.5	0	0	1	0	0	0	0	1	0	0	0	0	0	1	0	0	0.5	0.5	0	0	Morton (1976); Clemente and Ingole (2011); Idris et al. (2017); Argente et al. (2024); Alias et al. (2025).

Supplementary Table S2. Raw body-size measurements of benthic macroinvertebrates across six sampling sites in Kampung Blekok, Situbondo, Indonesia.

No	Family	Genus	n-total	Body size*												Average (mm)	Measurement type
				1	2	3	4	5	6	7	8	9	10	11	12		
Site 1																	
1	Nereididae	*Nereis*	26	37	34	32.5	36	28.5	30.5	34.5	35	32	35	36		33.73	total body length
			26														
Site 2																	
1	Ocypodidae	*Gelasimus*	1	20												20.00	Maximum carapace width
2	Nassariidae	*Nassarius*	4	11	10	11	12									11.00	Shell height
3	Nereididae	*Nereis*	2	51	42											46.50	total body length
4	Nuculidae	*Ennucula*	1	10												10.00	Shell length
1	Ellobiidae	*Cassidula*	1	24												24.00	Shell height
2	Ocypodidae	*Austruca*	4	17	21	15	13									16.50	maximum carapace width
3	Potamididae	*Pirenella*	1	32												32.00	Shell height
			14														
Site 3																	
1	Potamididae	*Pirenella*	2	33	32.5											32.75	Shell height
2	Ocypodidae	*Austruca*	5	13	17	15	12	16								14.60	maximum carapace width
3	Varudidae	*Metaplax*	1	6												6.00	Maximum carapace width
4	Ellobiidae	*Cassidula*	8	18.5	22											20.25	Shell height
			16														
Site 4																	
1	Ocypodidae	*Gelasimus*	3	14	5	7										8.67	Maximum carapace width
2	Neritidae	*Neritodryas*	6	8	6	6	6	7	7							6.67	Shell height
3	Littorinidae	*Littoraria*	2	7	4											5.50	Shell height

No	Family	Genus	n-total	Body size*												Average (mm)	Measurement type
				1	2	3	4	5	6	7	8	9	10	11	12		
4	Olividae	*Olivella*	4	8	7	7	8									7.50	Shell height
5	Nassariidae	*Nassarius*	1	8												8.00	Shell height
6	Cyrenoididae	*Geloina*	2	18	16											17.00	Shell length
8	Nereididae	*Nereis*	2	34	36											35.00	total body length
			20														
Site 5																	
1	Potamididae	*Pirenella*	13	12	9	13	14.5	15	13	18	15.5	19	11	16	17	14.42	Shell height
2	Lingulidae	*Lingula*	1	23												23.00	Shell length
3	Littorinidae	*Littoraria*	1	16												16.00	Shell height
			15														
Site 6																	
1	Nassariidae	*Nassarius*	7	13	9	11.5	12	10	12.5	11						11.29	Shell height
2	Potamididae	*Pirenella*	15	17	16	15.5	18	16	19	20	14	15	17	14	17.5	16.58	Shell height
3	Littorinidae	*Littoraria*	1	40												40.00	Shell height
	Total		**23**														

Note: Body-size measurements were obtained only from intact specimens. Individuals damaged during measurement were excluded when reliable measurements could not be obtained.

Supplementary Table S3. Site × taxon abundance matrix

Site	Austruca	*Cassidula*	*Ennucula*	*Geloina*	*Gelasimus*	*Metaplax*	*Lingula*	*Littoraria*	*Nassarius*	*Nereis*	*Neritodryas*	*Olivella*	*Pirenella*	***Total***
Site 1	0	0	0	0	0	0	0	0	0	26	0	0	0	**26**
Site 2	4	1	1	0	1	0	0	0	4	2	0	0	1	**14**
Site 3	5	8	0	0	0	1	0	0	0	0	0	0	2	**16**
Site 4	0	0	0	2	3	0	0	2	1	2	6	4	0	**20**
Site 5	0	0	0	0	0	0	1	1		0	0	0	13	**15**
Site 6	0	0	0	0	0	0	0	1	7	0	0	0	15	**23**

Supplementary Table S4. Relative taxonomic composition of dominant benthic macroinvertebrate genera across six sampling sites in Kampung Blekok

Site	**Austruca**	***Cassidula***	***Littoraria***	***Nassarius***	***Nereis***	***Pirenella***	**Others**	**Total**
Site 1	0,00	0,00	0,00	0,00	100	0,00	0,00	100,00
Site 2	28,57	7,14	0,00	28,57	14,29	7,14	14,29	100,00
Site 3	31,25	50,00	0,00	0,00	0,00	12,50	6,25	100,00
Site 4	0,00	0,00	10,00	5,00	10,00	0,00	75,00	100,00
Site 5	0,00	0,00	6,67	0,00	0,00	86,70	6,67	100,00
Site 6	0,00	0,00	4,35	30,43	0,00	65,22	0,00	100,00

Note: Values represent relative abundance (%). Less abundant genera were grouped as “Others”.

Supplementary Table S5. Abundance-weighted relative contributions of biological trait modalities and feeding guilds across six sampling sites in Kampung Blekok

Angle	Value_pct						Label	Trait_Group
	Site 1	Site 2	Site 3	Site 4	Site 5	Site 6		
35	0,00	0,00	0,00	0,00	0,00	0,00	S	Body Size
45	0,00	71,43	37,50	90,00	93,33	95,65	M	Body Size
55	100,00	28,57	62,50	10,00	6,67	4,35	L	Body Size
75	0,00	0,00	0,00	0,00	0,00	0,00	ATT	Sediment Position
85	50,00	7,14	0,00	5,00	0,00	0,00	BP	Sediment Position
95	50,00	67,86	81,25	62,50	93,33	100,00	EPI	Sediment Position
105	0,00	0,00	0,00	0,00	0,00	0,00	CSS	Sediment Position
115	0,00	17,86	18,75	17,50	0,00	0,00	D3	Sediment Position
125	0,00	7,14	0,00	15,00	6,67	0,00	S3	Sediment Position
145	0,00	0,00	0,00	0,00	0,00	0,00	ATT	Living Habit
155	0,00	39,29	18,75	20,00	6,67	15,22	BUR	Living Habit
165	100,00	60,71	81,25	80,00	93,33	84,78	FSC	Living Habit
175	0,00	0,00	0,00	0,00	0,00	0,00	PC	Living Habit
185	0,00	0,00	0,00	0,00	0,00	0,00	TD	Living Habit
205	25,00	57,14	75,00	20,00	86,67	65,22	DF	Feeding Guild
213	0,00	0,00	0,00	10,00	6,67	0,00	FF	Feeding Guild
221	25,00	7,14	25,00	45,00	6,67	4,35	GS	Feeding Guild
229	0,00	0,00	0,00	0,00	0,00	0,00	OM	Feeding Guild
237	25,00	10,71	0,00	12,50	0,00	7,61	PR	Feeding Guild
245	25,00	25,00	0,00	12,50	0,00	22,83	SC	Feeding Guild
253	0,00	0,00	0,00	0,00	0,00	0,00	SSDF	Feeding Guild
270	0,00	0,00	0,00	0,00	0,00	0,00	LIM	Mobility
280	100,00	100,00	100,00	100,00	100,00	100,00	MOB	Mobility
290	0,00	0,00	0,00	0,00	0,00	0,00	SES	Mobility
310	50,00	25,00	18,75	12,50	0,00	0,00	BD	Bioturbation
320	0,00	0,00	0,00	5,00	0,00	0,00	BI	Bioturbation
330	0,00	0,00	0,00	40,00	6,67	4,35	NB	Bioturbation
340	50,00	75,00	81,25	42,50	93,33	95,65	SM	Bioturbation

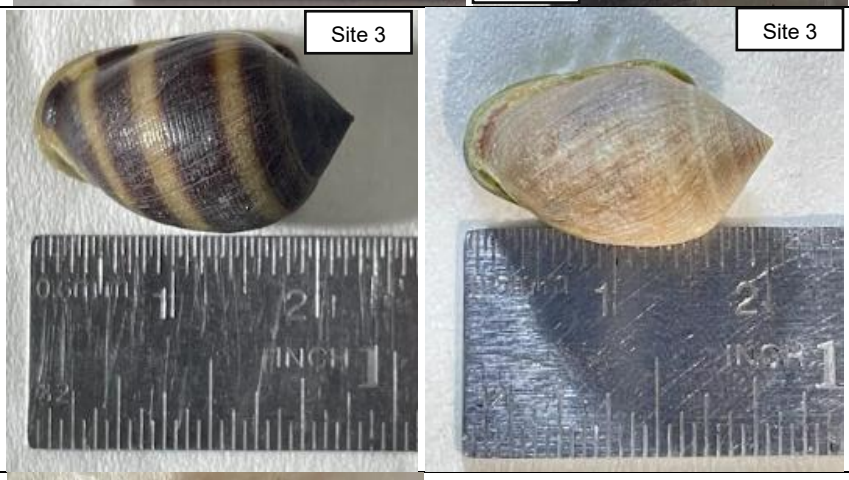
Note: Values represent the relative contribution (%) of each modality within its corresponding trait group. The Angle column specifies the plotting position, while Label and Trait_Group identify the trait modalities and their respective groups.

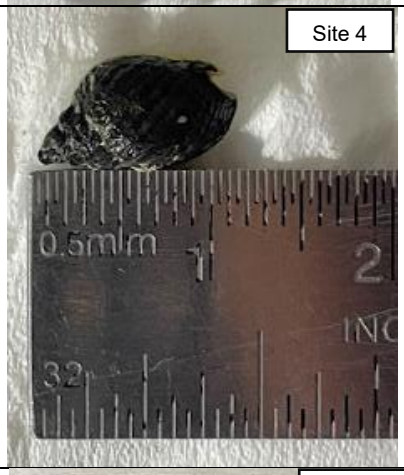
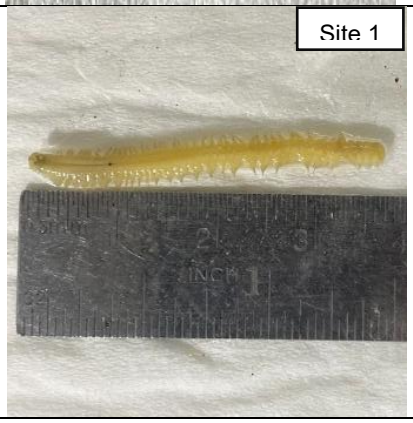
Supplementary Table S6. Sediment grain-size composition and organic carbon content across six sampling sites in Kampung Blekok

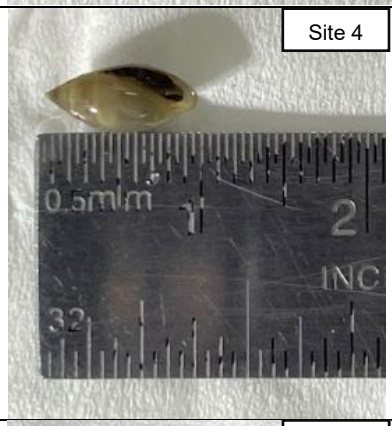
Site	Gravel (%)	Sand (%)	Fine fraction (%)	C. Organik (%)
1	11.75406	64.63021	23.61573	1
2	9.221048	49.42739	41.35156	0.51
3	4.47426	86.02681	9.498929	0.76
4	20.85794	73.87035	5.271707	3.84
5	8.84917	81.35684	9.79399	1.03
6	9.455727	57.1297	33.41457	1.34

Note: Grain-size composition is expressed as the percentage of gravel, sand, and fine fraction. Sediment organic carbon was analyzed by an external laboratory using the Walkley–Black method, as specified in the laboratory report. Organic carbon values represent the final concentrations reported by the laboratory.

Suplementary Table S7. Representative photographs of body-size measurements

Genus	Photographs
Austruca	
Cassidula	
Gelasimus	
Metaplax	

Genus	Photographs
Lingula	
Littoraria	
Nassarius	
Nereis	

Genus	Photographs
Neritodryas	
Olivella	
Pirenella	
Geloina	

Note: Representative photographs of selected benthic macroinvertebrate taxa collected during the survey. Sampling-site information was assigned based on the original station-specific photographic archives. The photographs represent selected specimens rather than all individuals or all occurrences of each taxon. Formal specimen voucher codes were not available

Supplementary File S8. Reproducible R Scripts, Statistical Outputs, and Graphical Results

S8.1. Software Environment

Script

```
# ================================================================
# S4.1. SOFTWARE ENVIRONMENT
# ================================================================

# Install packages if necessary (run only once)
# install.packages(c("readxl", "vegan"))

# Load packages
library(readxl)
library(vegan)

# R version
R.version.string

# Package versions
packageVersion("readxl")
packageVersion("vegan")

# Complete session information
sessionInfo()
```

Console

```
> # R version
> R.version.string
[1] "R version 4.4.1 (2024-06-14 ucrt)"
> # Package versions
> packageVersion("readxl")
[1] '1.4.5'
> packageVersion("vegan")
[1] '2.7.3'
> # Complete session information
> sessionInfo()
R version 4.4.1 (2024-06-14 ucrt)
Platform: x86_64-w64-mingw32/x64
Running under: Windows 11 x64 (build 26200)

Matrix products: default


locale:
[1] LC_COLLATE=English_Indonesia.utf8  LC_CTYPE=English_Indonesia.utf8
[3] LC_MONETARY=English_Indonesia.utf8 LC_NUMERIC=C
[5] LC_TIME=English_Indonesia.utf8

time zone: Asia/Jakarta
tzcode source: internal

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
[1] vegan_2.7-3    permute_0.9-10 readxl_1.4.5

loaded via a namespace (and not attached):
 [1] utf8_1.2.4        cellranger_1.1.0  mgcv_1.9-1
 [4] Matrix_1.7-0      lattice_0.22-6    magrittr_2.0.3
 [7] splines_4.4.1     glue_1.7.0        tibble_3.2.1
[10] parallel_4.4.1    pkgconfig_2.0.3   lifecycle_1.0.5
[13] cli_3.6.3         fansi_1.0.6       vctrs_0.7.3
[16] grid_4.4.1        compiler_4.4.1    rstudioapi_0.16.0
[19] tools_4.4.1       cluster_2.1.6     nlme_3.1-164
[22] pillar_1.9.0      rlang_1.2.0       MASS_7.3-60.2
```

S8.2. Taxonomic Clustering

Script

```
#
=============================================================
=
# A. TAXONOMIC CLUSTERING
#
=============================================================
=


# chose file R_Taxonomic.xlsx
tax_raw <- read_excel("R_Taxonomic.xlsx")
View(R_Taxonomic)

# data check
print(tax_raw)

# get the site name
tax_site <- tax_raw$Site

# Select only the taxa column
tax <- as.data.frame(tax_raw[, -1])

# empty el = taxa not found, change to 0
tax[is.na(tax)] <- 0

# Make Site the row names
rownames(tax) <- tax_site

# Check the final matrix
print(tax)

# cCheck number of abundance per site
rowSums(tax)



# ------------------------------------------------------------
# Bray-Curtis dissimilarity
# ------------------------------------------------------------
tax_bray <- vegdist(
  tax,
  method = "bray"
```

Console

```
> # chose file R_Taxonomic.xlsx
> tax_raw <- read_excel("R_Taxonomic.xlsx")
> View(R_Taxonomic)
> # data check
> print(tax_raw)
# A tibble: 6 x 14
  Site   Austruca Cassidula Ennucula Geloina Gelasimus Metaplax Lingula Littoraria Nassarius Nereis Neritodryas Olivella Pirenella
  <chr>     <db1>     <db1>    <db1>   <db1>     <db1>    <db1>   <db1>      <db1>     <db1>  <db1>      <db1>    <db1>     <db1>
1 Site 1        0         0        0       0         0        0       0          0         0     26          0        0         0
2 Site 2        4         1        1       0         1        0       0          0         4      2          0        0         1
3 Site 3        5         8        0       0         0        1       0          0         0      0          0        0         2
4 Site 4        0         0        0       2         3        0       0          3         1      2          6        4        NA
5 Site 5        0         0        0       0         0        0       1          1        NA      0          0        0        13
6 Site 6        0         0        0       0         0        0       0          1         7      0          0        0        15
> # get the site name
> tax_site <- tax_raw$Site
> # Select only the taxa column
> tax <- as.data.frame(tax_raw[, -1])
> # empty el = taxa not found, change to 0
> tax[is.na(tax)] <- 0
> # Make Site the row names
> rownames(tax) <- tax_site
> # Check the final matrix
> print(tax)
       Austruca Cassidula Ennucula Geloina Gelasimus Metaplax Lingula Littoraria Nassarius Nereis Neritodryas Olivella Pirenella
Site 1        0         0        0       0         0        0       0          0         0     26          0        0         0
Site 2        4         1        1       0         1        0       0          0         4      2          0        0         1
Site 3        5         8        0       0         0        1       0          0         0      0          0        0         2
Site 4        0         0        0       2         3        0       0          3         1      2          6        4         0
Site 5        0         0        0       0         0        0       1          1         0      0          0        0        13
Site 6        0         0        0       0         0        0       0          1         7      0          0        0        15
> # cCheck number of abundance per site
> rowSums(tax)
Site 1 Site 2 Site 3 Site 4 Site 5 Site 6
    26     14     16     21     15     23
> # ------------------------------------------------------------
> # Bray-Curtis dissimilarity
> # ------------------------------------------------------------
> tax_bray <- vegdist(
+   tax,
+   method = "bray"
+ )
> # Matriks Bray-Curtis
> tax_bray_matrix <- as.matrix(tax_bray)
> print(round(tax_bray_matrix, 3))
       Site 1 Site 2 Site 3 Site 4 Site 5 Site 6
Site 1  0.000  0.900  1.000  0.915  1.000  1.000
Site 2  0.900  0.000  0.600  0.771  0.931  0.730
Site 3  1.000  0.600  0.000  1.000  0.871  0.897
Site 4  0.915  0.771  1.000  0.000  0.944  0.909
Site 5  1.000  0.931  0.871  0.944  0.000  0.263
Site 6  1.000  0.730  0.897  0.909  0.263  0.000
```

Taxonomic Bray-Curtis distance Matrix

Script	Console

```
)

# Matriks Bray-Curtis
tax_bray_matrix <- as.matrix(tax_bray)

print(round(tax_bray_matrix, 3))

sessionInfo()
# ------------------------------------------------------------
# Hierarchical clustering
# Average linkage = UPGMA
# ------------------------------------------------------------
tax_cluster <- hclust(
  tax_bray,
  method = "average"
)

# clustering result
print(tax_cluster)

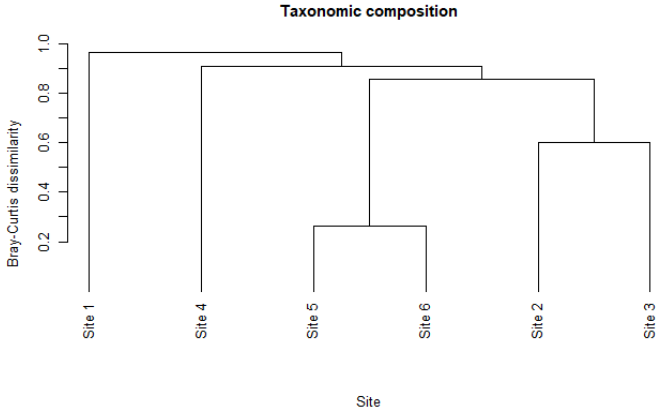
# ------------------------------------------------------------
# Plot dendrogram taxonomic
# ------------------------------------------------------------
plot(
  tax_cluster,
  hang = -1,
  main = "Taxonomic composition",
  xlab = "Site",
  ylab = "Bray-Curtis dissimilarity",
  sub = ""
)
```

```
> # ------------------------------------------------------------
> # Hierarchical clustering
> # Average linkage = UPGMA
> # ------------------------------------------------------------
> tax_cluster <- hclust(
+   tax_bray,
+   method = "average"
+ )
> # clustering result
> print(tax_cluster)

Call:
hclust(d = tax_bray, method = "average")

Cluster method   : average
Distance         : bray
Number of objects: 6

> # ------------------------------------------------------------
> # Plot dendrogram taxonomic
> # ------------------------------------------------------------
> plot(
+   tax_cluster,
+   hang = -1,
+   main = "Taxonomic composition",
+   xlab = "Site",
+   ylab = "Bray-Curtis dissimilarity",
+   sub = ""
+ )
```



S8.3. Functional Clustering

Script

```
# ================================================================
# B. FUNCTIONAL CLUSTERING
# ================================================================

# Chose file NMDS_Functional_Kampung_Blekok_6_Site.xlsx
func_raw <- read_excel("NMDS_Functional_Kampung_Blekok_6_Site.xlsx")
View(func_raw)

# Data check
print(func_raw)

# Get site name
func_site <- func_raw$Site

# Select only the functional modalities column
func <- as.data.frame(func_raw[, -1])

# For safety
func[is.na(func)] <- 0

# Make Site the row names
rownames(func) <- func_site

# Cek data final
print(func)

# Cek jumlah nilai per site
rowSums(func)



# ----------------------------------------------------------------
# Bray-Curtis dissimilarity
# ----------------------------------------------------------------
func_bray <- vegdist(
  func,
  method = "bray"
)

# Matriks Bray-Curtis
```

Console

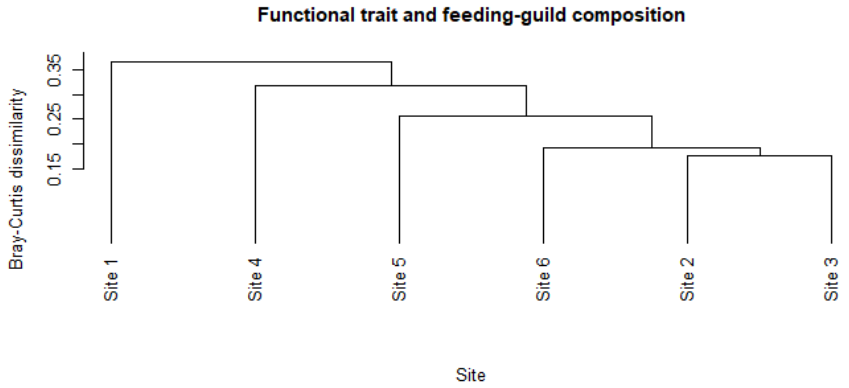
```
> # Chose file NMDS_Functional_Kampung_Blekok_6_Site.xlsx
> func_raw <- read_excel("NMDS_Functional_Kampung_Blekok_6_Site.xlsx")
> View(func_raw)
> # Data check
> print(func_raw)
# A tibble: 6 x 29
  Site    BS_S  BS_M  BS_L SP_ATT SP_BP SP_EPI SP_CSS SP_D3 SP_S3 LH_ATT LH_BUR LH_FSC LH_PC LH_TD
  <chr>  <dbl> <dbl> <dbl>  <dbl> <dbl>  <dbl>  <dbl> <dbl> <dbl>  <dbl>  <dbl>  <dbl> <dbl> <dbl>
1 Site 1     0     0   100      0    50     50      0     0     0      0      0    100      0     0
2 Site 2     0  78.6    21.4     0     7.14   67.9     0  17.9   7.14     0   39.3    60.7     0     0
3 Site 3     0  50      50      0     0      81.2     0  18.8   0        0   18.8    81.2     0     0
4 Site 4     0  90      10      0     5      62.5     0  12.5  20        0   20      80      0     0
5 Site 5     0   6.67   93.3     0     0      93.3     0   0     6.67     0    6.67   93.3     0     0
6 Site 6     0 100      0      0     0     100      0     0     0        0   15.2    84.8     0     0
# i 14 more variables: FG_DF <dbl>, FG_FF <dbl>, FG_GS <dbl>, FG_OM <dbl>, FG_PR <dbl>,
#   FG_SC <dbl>, FG_SSDF <dbl>, MO_LIM <dbl>, MO_MOB <dbl>, MO_SES <dbl>, BT_BD <dbl>, BT_BI <dbl>,
#   BT_NB <dbl>, BT_SM <dbl>
> # Get site name
> func_site <- func_raw$Site
> # Select only the functional modalities column
> func <- as.data.frame(func_raw[, -1])
> # Select only the functional modalities column
> func <- as.data.frame(func_raw[, -1])
> # For safety
> func[is.na(func)] <- 0
> # Make Site the row names
> rownames(func) <- func_site
> # Cek data final
> print(func)
       BS_S       BS_M      BS_L SP_ATT      SP_BP     SP_EPI SP_CSS     SP_D3      SP_S3 LH_ATT
Site 1    0   0.000000 100.00000      0 50.000000  50.00000      0  0.00000  0.000000      0
Site 2    0  78.571429  21.42857      0  7.142857  67.85714      0 17.85714  7.142857      0
Site 3    0  50.000000  50.00000      0  0.000000  81.25000      0 18.75000  0.000000      0
Site 4    0  90.000000  10.00000      0  5.000000  62.50000      0 12.50000 20.000000      0
Site 5    0   6.666667  93.33333      0  0.000000  93.33333      0  0.00000  6.666667      0
Site 6    0 100.000000   0.00000      0  0.000000 100.00000      0  0.00000  0.000000      0
         LH_BUR    LH_FSC LH_PC LH_TD     FG_DF     FG_FF     FG_GS FG_OM     FG_PR     FG_SC
Site 1  0.000000 100.00000     0     0 25.00000  0.000000 25.000000     0 25.000000 25.00000
Site 2 39.285714  60.71429     0     0 57.14286  0.000000  7.142857     0 10.714286 25.00000
Site 3 18.750000  81.25000     0     0 75.00000  0.000000 25.000000     0  0.000000  0.00000
Site 4 20.000000  80.00000     0     0 17.50000 10.000000 42.500000     0 13.750000 16.25000
Site 5  6.666667  93.33333     0     0 86.66667  6.666667  6.666667     0  0.000000  0.00000
Site 6 15.217391  84.78261     0     0 65.21739  0.000000  4.347826     0  7.608696 22.82609
       FG_SSDF MO_LIM    MO_MOB   MO_SES BT_BD BT_BI     BT_NB     BT_SM
Site 1       0      0 100.00000 0.000000 50.00     0  0.000000 50.00000
Site 2       0      0 100.00000 0.000000 25.00     0  0.000000 75.00000
Site 3       0      0 100.00000 0.000000 18.75     0  0.000000 81.25000
Site 4       0     10  90.00000 0.000000 17.50     5 40.000000 37.50000
Site 5       0      0  93.33333 6.666667  0.00     0  6.666667 93.33333
Site 6       0      0 100.00000 0.000000  0.00     0  4.347826 95.65217
> # Cek jumlah nilai per site
> rowSums(func)
Site 1 Site 2 Site 3 Site 4 Site 5 Site 6
   600    600    600    600    600    600
```

Script	Console

func_bray_matrix <- as.matrix(func_bray)
 	
print(round(func_bray_matrix, 3))
 	
# ---------- 	
# Hierarchical clustering 	
# Average linkage = UPGMA 	
# ---------- 	
func_cluster <- hclust(
  func_bray,
  method = "average"
)
 	
print(func_cluster)
 	
# ---------- 	
# Plot dendrogram functional 	
# ---------- 	
plot(
  func_cluster,
  hang = -1,
  main = "Functional trait and feeding-guild composition",
  xlab = "Site",
  ylab = "Bray-Curtis dissimilarity",
  sub = ""
)
	> # ----------
	> # Bray-Curtis dissimilarity
	> # ----------
	> func_bray <- vegdist(
	+ func,
	+ method = "bray"
	+)
	> # Matriks Bray-Curtis
	> func_bray_matrix <- as.matrix(func_bray)
	> print(round(func_bray_matrix, 3))
	  Site 1 Site 2 Site 3 Site 4 Site 5 Site 6
	Site 1 0.000 0.363 0.333 0.396 0.314 0.426
	Site 2 0.363 0.000 0.176 0.245 0.330 0.185
	Site 3 0.333 0.176 0.000 0.298 0.197 0.202
	Site 4 0.396 0.245 0.298 0.000 0.438 0.291
	Site 5 0.314 0.330 0.197 0.438 0.000 0.247
	Site 6 0.426 0.185 0.202 0.291 0.247 0.000
	> # ----------
	> # Hierarchical clustering
	> # Average linkage = UPGMA
	> # ----------
	> func_cluster <- hclust(
	+ func_bray,
	+ method = "average"
	+)
	> print(func_cluster)
	
	Call:
	hclust(d = func_bray, method = "average")
	
	Cluster method : average
	Distance : bray
	Number of objects: 6

Functional Bray-Curtis distance Matrix

Script	Console

	`> # ------------------------------------------------------------`
	`> # Plot dendrogram functional`
	`> # ------------------------------------------------------------`
	`> plot(`
	`+   func_cluster,`
	`+   hang = -1,`
	`+   main = "Functional trait and feeding-guild composition",`
	`+   xlab = "Site",`
	`+   ylab = "Bray-Curtis dissimilarity",`
	`+   sub = ""`
	`+ )`
	

S8.4. Cophenetic Correlation of Taxonomic and Functional

Script

```
# ============================================================
# E. COPHENETIC CORRELATION
# ============================================================

# Cophenetic distances dari dendrogram
tax_coph <- cophenetic(tax_cluster)
func_coph <- cophenetic(func_cluster)

# Correlation between original dissimilarity and cophenetic distance
tax_cophenetic_r <- cor(
  as.vector(tax_bray),
  as.vector(tax_coph),
  method = "pearson"
)

# Taxonomic cophenetic matrix
tax_coph_matrix <- as.matrix(
  cophenetic(tax_cluster)
)

print(round(tax_coph_matrix, 6))
cat("Taxonomic cophenetic correlation =", round(tax_cophenetic_r, 3), "\n")

func_cophenetic_r <- cor(
  as.vector(func_bray),
  as.vector(func_coph),
  method = "pearson"
)
# Functional cophenetic matrix
func_coph_matrix <- as.matrix(
  cophenetic(func_cluster)
)

print(round(func_coph_matrix, 6))

cat("Functional cophenetic correlation =", round(func_cophenetic_r, 3), "\n")
```

Console

```
> # Cophenetic distances dari dendrogram
> tax_coph <- cophenetic(tax_cluster)
> func_coph <- cophenetic(func_cluster)
> # Correlation between original dissimilarity and cophenetic distance
> tax_cophenetic_r <- cor(
+   as.vector(tax_bray),
+   as.vector(tax_coph),
+   method = "pearson"
+ )
> # Taxonomic cophenetic matrix
> tax_coph_matrix <- as.matrix(
+   cophenetic(tax_cluster)
+ )
> print(round(tax_coph_matrix, 6))
         Site 1   Site 2   Site 3   Site 4   Site 5   Site 6
Site 1 0.000000 0.962979 0.962979 0.962979 0.962979 0.962979
Site 2 0.962979 0.000000 0.600000 0.906241 0.857292 0.857292
Site 3 0.962979 0.600000 0.000000 0.906241 0.857292 0.857292
Site 4 0.962979 0.906241 0.906241 0.000000 0.906241 0.906241
Site 5 0.962979 0.857292 0.857292 0.906241 0.000000 0.263158
Site 6 0.962979 0.857292 0.857292 0.906241 0.263158 0.000000
> cat("Taxonomic cophenetic correlation =", round(tax_cophenetic_r, 3), "\n")
Taxonomic cophenetic correlation = 0.941
> func_cophenetic_r <- cor(
+   as.vector(func_bray),
+   as.vector(func_coph),
+   method = "pearson"
+ )
> # Functional cophenetic matrix
> func_coph_matrix <- as.matrix(
+   cophenetic(func_cluster)
+ )
> print(round(func_coph_matrix, 6))
         Site 1   Site 2   Site 3   Site 4   Site 5   Site 6
Site 1 0.000000 0.366375 0.366375 0.366375 0.366375 0.366375
Site 2 0.366375 0.000000 0.175595 0.317893 0.257636 0.193485
Site 3 0.366375 0.175595 0.000000 0.317893 0.257636 0.193485
Site 4 0.366375 0.317893 0.317893 0.000000 0.317893 0.317893
Site 5 0.366375 0.257636 0.257636 0.317893 0.000000 0.257636
Site 6 0.366375 0.193485 0.193485 0.317893 0.257636 0.000000
> cat("Functional cophenetic correlation =", round(func_cophenetic_r, 3), "\n")
Functional cophenetic correlation = 0.797
>
```

Taxonomic Cophenetic Matrix

Functional Cophenetic Matrix

S8.5. Mantel Test of Taxonomic vs Functional

Script

```
#
================================================================
# F. MANTEL TEST
# Taxonomic vs Functional Bray-Curtis matrices
#
================================================================

set.seed(123)

mantel_result <- mantel(
  tax_bray,
  func_bray,
  method = "pearson",
  permutations = 999
)

print(mantel_result)
```

Console

```
> set.seed(123)
> mantel_result <- mantel(
+   tax_bray,
+   func_bray,
+   method = "pearson",
+   permutations = 999
+ )
'nperm' >= set of all permutations: complete enumeration.
Set of permutations < 'minperm'. Generating entire set.
> print(mantel_result)

Mantel statistic based on Pearson's product-moment correlation

Call:
mantel(xdis = tax_bray, ydis = func_bray, method = "pearson",      permutations = 999)

Mantel statistic r: 0.5169
      Significance: 0.022222

Upper quantiles of permutations (null model):
  90%   95% 97.5%   99%
0.346 0.433 0.496 0.559
Permutation: free
Number of permutations: 719
```

Mantel test taxonomic vs functional

S8.6. Clustering of Sediment Characteristics

Script

```
# ============================================================
# Clustering of Sediment Characteristics
# Gravel + Fine + C-Organik
# ============================================================

# Install sekali saja jika belum ada:
# install.packages("readxl")

library(readxl)

# ============================================================
# 1. IMPORT DATA EXCEL
# ============================================================

sed <- read_excel("Sediment.xlsx")

# Data check
sed
names(sed)


# ============================================================
# 2. Clean up column names
# ============================================================

# As per the headers in the Excel file:
colnames(sed) <- c(
  "Site",
  "Gravel",
  "Fine",
  "C_Organik"
)

sed

# ============================================================
# 3. SELECT SEDIMENT VARIABLE
# ============================================================

sed_use <- sed[, c(
```

Console

```
> View(sed_cluster)
> sed <- read_excel("Sediment.xlsx")
Warning messages:
1: In doTryCatch(return(expr), name, parentenv, handler) :
  display list redraw incomplete
2: In doTryCatch(return(expr), name, parentenv, handler) :
  invalid graphics state
3: In doTryCatch(return(expr), name, parentenv, handler) :
  invalid graphics state
> # Data check
> sed
# A tibble: 6 × 4
  Site   Gravel  Fine C_Organik
  <chr>   <db1> <db1>     <db1>
1 Site 1  11.8  23.6      1
2 Site 2   9.22 41.4      0.51
3 Site 3   4.47  9.50     0.757
4 Site 4  20.9   5.27     3.84
5 Site 5   8.85  9.79     1.03
6 Site 6   9.46 33.4      1.34
> names(sed)
[1] "Site"      "Gravel"    "Fine"      "C_Organik"
> # As per the headers in the Excel file:
> colnames(sed) <- c(
+   "Site",
+   "Gravel",
+   "Fine",
+   "C_Organik"
+ )
> sed
# A tibble: 6 × 4
  Site   Gravel  Fine C_Organik
  <chr>   <db1> <db1>     <db1>
1 Site 1  11.8  23.6      1
2 Site 2   9.22 41.4      0.51
3 Site 3   4.47  9.50     0.757
4 Site 4  20.9   5.27     3.84
5 Site 5   8.85  9.79     1.03
6 Site 6   9.46 33.4      1.34
```

Script	Console

```
  "Gravel",
  "Fine",
  "C_Organik"
)]

# Convert to a standard data.frame
sed_use <- as.data.frame(sed_use)

# Use the site name as the row names
rownames(sed_use) <- sed$Site

sed_use


# ============================================================
# 4. Ensure all variables are numeric.
# ============================================================

sed_use$Gravel    <- as.numeric(sed_use$Gravel)
sed_use$Fine      <- as.numeric(sed_use$Fine)
sed_use$C_Organik <- as.numeric(sed_use$C_Organik)

str(sed_use)

# Cek missing value
sum(is.na(sed_use))


# ============================================================
# 5. DATA STANDARDIZATION
# Z-score
#
# It is important because gravel, fines, and organic carbon
# have different value ranges.
# ============================================================

sed_z <- scale(
  sed_use,
  center = TRUE,
  scale = TRUE
)
```

```
> sed_use <- sed[, c(
+   "Gravel",
+   "Fine",
+   "C_Organik"
+ )]
> # Convert to a standard data.frame
> sed_use <- as.data.frame(sed_use)
> # Use the site name as the row names
> rownames(sed_use) <- sed$Site
> sed_use
          Gravel      Fine C_Organik
Site 1 11.754060 23.615731 1.0000000
Site 2  9.221048 41.351565 0.5100000
Site 3  4.474260  9.498929 0.7566667
Site 4 20.857944  5.271707 3.8400000
Site 5  8.849170  9.793990 1.0300000
Site 6  9.455727 33.414568 1.3400000
> sed_use$Gravel    <- as.numeric(sed_use$Gravel)
> sed_use$Fine      <- as.numeric(sed_use$Fine)
> sed_use$C_Organik <- as.numeric(sed_use$C_Organik)
> str(sed_use)
'data.frame':   6 obs. of  3 variables:
 $ Gravel   : num  11.75 9.22 4.47 20.86 8.85 ...
 $ Fine     : num  23.62 41.35 9.5 5.27 9.79 ...
 $ C_Organik: num  1 0.51 0.757 3.84 1.03 ...
> # Cek missing value
> sum(is.na(sed_use))
[1] 0
> sed_z <- scale(
+   sed_use,
+   center = TRUE,
+   scale = TRUE
+ )
> sed_z
```

Script

```
sed_z


# ============================================================
# 6. Calculate the distance between sites.
# Standardized Euclidean distance
# ============================================================

sed_dist <- dist(
  sed_z,
  method = "euclidean"
)

# View the distance matrix
as.matrix(sed_dist)

# ============================================================
# 7. HIERARCHICAL CLUSTERING
# UPGMA / average linkage
# ============================================================

sed_cluster <- hclust(
  sed_dist,
  method = "average"
)


# ============================================================
# 8. PLOT DENDROGRAM
# ============================================================

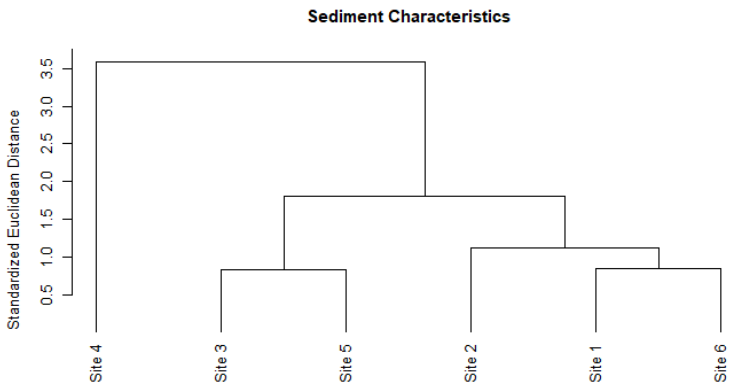
plot(
  sed_cluster,
  main = "Sediment Characteristics",
  xlab = "",
  sub = "",
  ylab = "Standardized Euclidean Distance",
  hang = -1
)
```

Console

```
           Gravel       Fine   C_Organik
Site 1  0.1797887  0.2127213 -0.33795443
Site 2 -0.2823851  1.4201493 -0.73913319
Site 3 -1.1484851 -0.7483287 -0.53717926
Site 4  1.8408852 -1.0361114  1.98724485
Site 5 -0.3502382 -0.7282414 -0.31339247
Site 6 -0.2395656  0.8798109 -0.05958551
attr(,"scaled:center")
   Gravel      Fine C_Organik
10.768702 20.491082  1.412778
attr(,"scaled:scale")
   Gravel      Fine C_Organik
 5.480648 14.688936  1.221401
> sed_dist <- dist(
+   sed_z,
+   method = "euclidean"
+ )
> # View the distance matrix
> as.matrix(sed_dist)
          Site 1    Site 2   Site 3   Site 4   Site 5    Site 6
Site 1 0.0000000 1.3536733 1.651550 3.118554 1.080251 0.8356769
Site 2 1.3536733 0.0000000 2.343760 4.239650 2.191219 0.8692434
Site 3 1.6515504 2.3437601 0.000000 3.923247 0.829266 1.9248556
Site 4 3.1185536 4.2396499 3.923247 0.000000 3.191980 3.4912102
Site 5 1.0802511 2.1912193 0.829266 3.191980 0.000000 1.6317165
Site 6 0.8356769 0.8692434 1.924856 3.491210 1.631716 0.0000000
> sed_cluster <- hclust(
+   sed_dist,
+   method = "average"
+ )
> plot(
+   sed_cluster,
+   main = "Sediment Characteristics",
+   xlab = "",
+   sub = "",
+   ylab = "Standardized Euclidean Distance",
+   hang = -1
+ )
```

Z-score for sediment characteristics

Sediment euclidean distance matrix

Script	Console
	

S8.7. Cophenetic Correlation of Sediment Characteristics

Script

```
# ============================================================
# 9. COPHENETIC CORRELATION
# ============================================================

cophen_sed <- cor(
  sed_dist,
  cophenetic(sed_cluster),
  method = "pearson"
)
# Sediment cophenetic matrix
sed_coph_matrix <- as.matrix(
  cophenetic(sed_cluster)
)

print(round(sed_coph_matrix, 6))

cat(
  "Sediment cophenetic correlation =",
  round(cophen_sed, 3),
  "\n"
)
```

Console

```
> cophen_sed <- cor(
+   sed_dist,
+   cophenetic(sed_cluster),
+   method = "pearson"
+ )
> # Sediment cophenetic matrix
> sed_coph_matrix <- as.matrix(
+   cophenetic(sed_cluster)
+ )
> print(round(sed_coph_matrix, 6))
         Site 1   Site 2   Site 3   Site 4   Site 5   Site 6
Site 1 0.000000 1.111458 1.803892 3.592928 1.803892 0.835677
Site 2 1.111458 0.000000 1.803892 3.592928 1.803892 1.111458
Site 3 1.803892 1.803892 0.000000 3.592928 0.829266 1.803892
Site 4 3.592928 3.592928 3.592928 0.000000 3.592928 3.592928
Site 5 1.803892 1.803892 0.829266 3.592928 0.000000 1.803892
Site 6 0.835677 1.111458 1.803892 3.592928 1.803892 0.000000
> cat(
+   "Sediment cophenetic correlation =",
+   round(cophen_sed, 3),
+   "\n"
+ )
Sediment cophenetic correlation = 0.943
>
```

Sediment Cophenetic Matrix

S8.8. Mantel Test of Sediment Characteristics

Script

```
# ============================================================
# 10. CHECK SITE LABEL
# ============================================================

labels(sed_dist)

# Cophenetic correlation sediment
cophen_sed <- cor(
  sed_dist,
  cophenetic(sed_cluster),
  method = "pearson"
)

cophen_sed

library(vegan)

# Taxonomy vs Sediment
set.seed(123)
mantel_tax_sed <- mantel(
  tax_bray,
  sed_dist,
  method = "pearson",
  permutations = 999
)

mantel_tax_sed


# Functional vs Sediment
set.seed(123)
mantel_func_sed <- mantel(
  func_bray,
  sed_dist,
  method = "pearson",
  permutations = 999
)

mantel_func_sed
```

Console

```
> labels(sed_dist)
[1] "Site 1" "Site 2" "Site 3" "Site 4" "Site 5" "Site 6"
> # Cophenetic correlation sediment
> cophen_sed <- cor(
+   sed_dist,
+   cophenetic(sed_cluster),
+   method = "pearson"
+ )
> cophen_sed
[1] 0.9433041
> library(vegan)
> # Taxonomy vs Sediment
> set.seed(123)
> mantel_tax_sed <- mantel(
+   tax_bray,
+   sed_dist,
+   method = "pearson",
+   permutations = 999
+ )
+ )
'nperm' >= set of all permutations: complete enumeration.
Set of permutations < 'minperm'. Generating entire set.
> mantel_tax_sed

Mantel statistic based on Pearson's product-moment correlation

Call:
mantel(xdis = tax_bray, ydis = sed_dist, method = "pearson",      permutations = 999)

Mantel statistic r: 0.07655
      Significance: 0.39583

Upper quantiles of permutations (null model):
  90%   95% 97.5%   99%
0.374 0.463 0.520 0.564
Permutation: free
Number of permutations: 719
```

Mantel test taxonomic-sediment

Script	Console

	`> # Functional vs Sediment`
	`> set.seed(123)`
	`> mantel_func_sed <- mantel(`
	`+  func_bray,`
	`+  sed_dist,`
	`+  method = "pearson",`
	`+  permutations = 999`
	`+ )`
	`'nperm' >= set of all permutations: complete enumeration.`
	`Set of permutations < 'minperm'. Generating entire set.`
	`> mantel_func_sed`
	
	`Mantel statistic based on Pearson's product-moment correlation`
	
	`Call:`
	`mantel(xdis = func_bray, ydis = sed_dist, method = "pearson",  permutations = 999)`
	
	`Mantel statistic r: 0.1222`
	`  Significance: 0.38611`
	
	`Upper quantiles of permutations (null model):`
	`  90%  95% 97.5%  99%`
	`0.512 0.575 0.625 0.689`
	`Permutation: free`
	`Number of permutations: 719`
	
	Mantel test functional-sediment