转录组年龄指数法在动物发育进化研究中的应用

Application of Transcriptome Age Index in Animal Evo-devo Studies

王静^{1,3}, 王师^{1,2,3,*}

¹海洋生物遗传学与育种教育部重点实验室, 中国海洋大学, 山东青岛;

²方宗熙-萨斯海洋分子生物学研究中心, 中国海洋大学, 山东青岛;

³海洋生物学与生物技术功能实验室, 青岛海洋科学与技术国家试点实验室, 山东青岛

*通讯作者: swang@ouc.edu.cn

引用格式: 王静, 王师. (2021). 转录组年龄指数法在动物发育进化研究中的应用. *Bio-101* e1010642. Doi: 10.21769/BioProtoc.1010642.

How to cite: Wang, J., and Wang, S. (2021). Application of Transcriptome Age Index in Animal Evo-devo Studies. *Bio-101* e1010642. Doi: 10.21769/BioProtoc.1010642. (in Chinese)

摘要: 随着生命树各主要进化分支的基因组数据迅速增长, 使得从分子水平解析宏观进化生物学问题成为可能。转录组年龄指数 (TAI) 分析方法是 Domazet-Lošo 等开发的一种结合基因年龄与基因表达量的加权算法。TAI 指数可以表征转录组年龄, 其值越高代表转录组越年轻。本团队提出一种改进的 TAI 计算方法, 根据基因的系统发生起源和表达谱信息来重演宏观进化趋势, 对解决发育进化生物学问题具有重要价值。研究方案主要包括: (1) 收集足够完整的表达谱数据; (2) 根据 NCBI 网站 Taxonomy 数据库中的种系发生树划分系统进化层级, 定义基因年龄; (3) 用 *myTAI* 软件包进行 TAI 相关分析。

关键词: 转录组年龄指数, 基因年龄, 进化层级, 发育进化

研究背景: 随着高通量测序技术的迅猛发展, 生命树各主要进化分支组学数据不断完善, 使得从分子水平解析宏观进化生物学问题成为可能。通过转录组测序获得的基因表达变化的差异往往蕴含着表型进化的信息, 因而追踪并量化转录组在进化过程中的变化规律, 将为我们理解转录组变化如何驱动表型进化提供新的认知 (Domazet-Lošo and Tautz, 2010)。Domazet-Lošo 等 (2007) 提出了系统进化层级理论, 认为基因组中每个基因都有其系统进化层级, 并可通过与已知基因组进行相似性比对来确定基因的“进化年龄”。

基于这一理论所建立的转录组年龄指数 (TAI) 分析方法, 采用基因表达量的基因年龄加权总和除以基因表达量的总和得到的 TAI 指数来表征转录组年龄, 即 TAI 值越高代表转录组越年轻 (Domazet-Lošo and Tautz, 2010)。该方法已广泛应用于动物、植物、微生物等类群, 在发育沙漏模型进化推演、干细胞起源进化、海洋幼虫起源进化等重要科学问题解析中取得突破性发现 (Domazet-Lošo and Tautz, 2010; Hemmrich, *et al.*, 2012; Futo *et al.*, 2020; Wang *et al.*, 2020)。在 Domazet-Lošo 等的研究基础上, Hajk-Georg 等 (2018) 开发了可用于 TAI 分析的 R 包, 进一步为研究者提供了一个标准化、自动化的 TAI 分析流程。但现有的 TAI 分析方法在解析宏观进化生物学问题时仍存有局限性, 如 Domazet-Lošo 等开发的算法仅计算各系统进化层级对进化终端 (即现存物种) 转录组 TAI 的贡献, 不能计算各祖先状态转录组的 TAI, 故无法实现对祖先转录组变化规律的推演分析。针对已有 TAI 算法的分析局限性, 本方案在 Domazet-Lošo 等开发的原始算法的基础上, 提出了一种改进的 TAI 分析方法 (Wang *et al.*, 2020), 通过逐层移除相对更年轻的基因来推演各祖先进化节点的转录组状态, 并利用该方法实现了对转录组宏观进化规律的推演。

应用程序和软件

1. Perl (<https://www.perl.org/>)
2. fqtrim (<https://ccb.jhu.edu/software/fqtrim/>)
3. STAR (<https://github.com/alexdobin/STAR>)
4. Samtools (<http://www.htslib.org/>)
5. featureCounts (<http://bioinf.wehi.edu.au/featureCounts/>)
6. Trinity (<https://github.com/trinityrnaseq/trinityrnaseq>)
7. TransDecoder (<https://github.com/TransDecoder/TransDecoder>)
8. BLAST+ (<ftp://ftp.ncbi.nlm.nih.gov/blast/executables/blast+/>)
9. R (<https://www.r-project.org/>)
10. myTAI (<https://github.com/drostlab/myTAI>)

数据分析

一、基因表达谱计算

1. 从 NCBI 网站的 SRA 数据库(<https://www.ncbi.nlm.nih.gov/sra/>)中下载所需的高通量测序数据（表 1）。

表 1 虾夷扇贝不同发育时期的转录组数据

Samples (full name)	Samples (short name)	Accession number
zygote	Zyg	SRR10055417
2-8cell	2-8cell	SRR2025018
blastula	Bla	SRR4415805
gastrula	Ga	SRR4415808
trochophore	Tro	SRR4415810
D-stage larvae	Dst	SRR4415807
early umbo larvae	E-umb	SRR10055402
middle umbo larvae	M-umb	SRR10055401
post umbo larvae	P-umb	SRR10055400
pedi-veliger larvae	Ped	SRR4415806
spat	Spa	SRR4415809
juvenile-1	Juv-1	SRR10055399
juvenile-2	Juv-2	SRR10055398
1year-old-adult-1	Adu-1y-1	SRR10055397
1year-old-adult-2	Adu-1y-2	SRR10055396
2year-old-adult-1	Adu-2y-1	SRR10055414
2year-old-adult-2	Adu-2y-2	SRR10055415
3year-old-adult-1	Adu-3y-1	SRR10055413
3year-old-adult-2	Adu-3y-2	SRR10055412
4year-old-adult-1	Adu-4y-1	SRR10055410
4year-old-adult-2	Adu-4y-2	SRR10055411

2. 测序数据预处理

对高通量测序平台测序获得的原始数据进行质量过滤，用 **fqtrim** 软件去掉含不确定碱基 N 的序列以及低质量 **reads**（超过 20% 的碱基质量值小于 20 的 **reads**），获得的 **clean reads** 用于后续分析，输入文件为 **FASTQ** 格式。

```
$ fqtrim -A -q 20 -w 80 -m 0 -p 8 -o trimed.fastq sample.R1.fastq, sample.R2.fastq
-P 33
```

3. 表达谱计算（无参照基因组）

3.1 对于没有参照基因组的物种，首先需要用 **Trinity** 软件将预处理获得的 **clean reads** 进行 **de novo** 拼接。

```
$ Trinity.pl --seqType fq --max_memory 40G --left sample1.R1.clean.fq.gz,
sample2.R1.clean.fq.gz --right sample1.R2.clean.fq.gz,
sample2.R2.clean.fq.gz --CPU 16 --full_cleanup > test.log
```

注：--left : left reads，多个文件以逗号隔开

--right : right reads，多个文件以逗号隔开

Trinity 运行过程中会自动检测之前输入，如果中途发生报错，进行修改之后，会在原有输出文件的基础上继续运行，不会对之前结果进行覆盖

3.2 对拼接获得的 **isoform** 聚类，每条 **unigene** 保留最长的转录本，以此作为转录组比对的参照数据库

```
$ perl trinity_unigene.pl -i trinity.fasta
```

3.3 用 **TransDecoder** 软件对 3.2 获得的数据库进行结构注释

```
$ TransDecoder.LongOrfs -t trinity_unigene.fasta
```

```
$ TransDecoder.Predict -t trinity_unigene.fasta --single_best_only
```

4. 表达谱计算（有参照基因组）

4.1 用 **STAR** 软件建立参照基因组的索引文件，将结果文件夹命名为 **genome_index**（图 1）。

```
4.2 $ STAR --runThreadN 8 --runMode genomeGenerate --
genomeDir ./genome_index/ --genomeFastaFiles genome.fasta --
sjdbOverhang 149 --genomeChrBinNbits 14 > starindex.log
```

注：--genomeDir：需要预先建立一个有可读写权限的文件夹，用于存放生成的索引文件

--sjdbOverhang: 该数值设定为单端 reads 的长度减 1，是在注释可变剪切序列时使用的最大长度值

--genomeChrBinNbits: 默认值为 18，如需节约运行内存，可根据参考基因组的大小重新定义， $=\min(18, \log_2(\text{GenomeLength}/\text{NumberOfReferences}))$

Remote Name	Size	Type
chrLength.txt	4,735	TXT 文件
chrName.txt	10,230	TXT 文件
chrNameLength.txt	14,965	TXT 文件
chrStart.txt	9,094	TXT 文件
Genome	1178,89...	文件
genomeParameters.txt	712	TXT 文件
SA	9664,56...	文件
SAindex	1565,87...	文件

图 1 genome_index 文件夹中的索引文件

4.3 将预处理获得的 clean reads 用 STAR 软件进行比对，将结果文件夹命名为 STAR_out（图 2）

```
$ STAR --runThreadN 8 --genomeDir ./genome_index / --readFilesCommand
zcat --readFilesIn R1.clean.fastq.gz R2.clean.fastq.gz --
sjdbGTFtagExonParentGene genome.gff3 --outFilterMultimapNmax 100000 -
-outFileNamePrefix ./STAR_out/ --outReadsUnmapped Fasta > star.log
```

注: **--outFilterMultimapNmax:** 默认值为 100000，限制 SAM 文件的大小，可根据数据量大小适当调高该参数。

--FileNamePrefix: 需要事先建立一个有可读写权限的文件夹，用于存放输出文件

Remote Name	Size	Type
Aligned.out.sam	13075,6...	SAM 文件
Log.final.out	1,696	OUT 文件
Log.out	2,974,066	OUT 文件
Log.progress.out	1,426	OUT 文件
SJ.out.tab	11,139,8...	TAB 文件

图 2. STAR_out 文件夹中的结果文件

4.4 将结果文件夹下的 sam 文件转换为 bam 文件

```
$ samtools view -bS ./STAR_out/Aligned.out.sam > sample.bam
```

4.5 根据基因组结构注释文件，用 featureCounts 软件统计 map 到各基因外显子区的 reads 数

```
$ featureCounts -p -t exon -g Parent -a genome.gff3 -o sample.txt sample.bam
```

注：-p：只能用在双端测序 *paired-end* 的情况中，会统计 *fragment* 而不统计 *read*

-t：根据 *gff* 文件设置 *feature-type*，同时 *read* 只有落到这些 *feature* 上才会被统计到，默认是“*exon*”。

-g：根据 *gff* 文件设置 *id identifier*，要确保-g 参数指定的值是 *gff* 文件中的唯一标识符，默认是“*gene_id*”。

4.6 将基因的 count 值转换为 TPM 值

```
$ perl count2TPM.pl -l exon-length -i count.txt -o TPM.txt
```

注：-l：步骤 3.4 中定义的 *feature* 的长度，输入文件包括两列，第一列为基因 ID (*geneID*)，第二列为 *feature* 长度 (*geneL*)，两列数据以 *tab* 键分隔。

-i：步骤 3.4 中统计获得的 *count* 值，输入文件第一列为基因 ID (*geneID*)，第二列及之后为 *count* 值 (*sample1, sample2...*)，每列数据以 *tab* 键分隔。

二、定义基因年龄

基因年龄的定义方法参照 Domazet-Lošo *et al.* (2010)开发的系统层级法 (Phylostratigraphy)。

1. 定义系统进化层级

根据 NCBI 网站 Taxonomy 数据库 (<https://www.ncbi.nlm.nih.gov/taxonomy>) 中的种系发生树，将全基因组的基因划分为不同的系统进化层级 (图 3, 图 4)。以虾夷扇贝为例，本研究中将虾夷扇贝的基因划分为 11 个进化层级 (图 5)，其中将细胞生物 (*cellular organisms*) 定义为系统进化层级的第一级 (PS1)，真核生物 (*Eukaryota*) 定义为系统进化层级的第二级 (PS2)，依次类推，将虾夷扇贝属 (*Mizuhopecten*) 定义为系统进化层级的最后一级 (PS11)。

NCBI Taxonomy Browser

Search for as ☒ lock

Display levels using filter:

Mizuhopecten yessoensis

Taxonomy ID: 6573 (for references in articles please use NCBI:txid6573)

current name

Mizuhopecten yessoensis (Jay, 1956)

basionym: ***Pecten yessoensis*** Jay, 1857

homotypic synonym: ☐ ***Patiopecten yessoensis***
Patiopecten yessoensis

Genbank common name: **Yesso scallop**

NCBI BLAST name: **bivalves**

Rank: **species**

Genetic code: [Translation table 1 \(Standard\)](#)

Mitochondrial genetic code: [Translation table 5 \(Invertebrate Mitochondrial\)](#)

Other names:

common name(s)

ezo giant scallop

[Lineage \(full\)](#)

[cellular organisms](#); [Eukaryota](#); [Opisthokonta](#); [Metazoa](#); [Eumetazoa](#); [Bilateria](#); [Protostomia](#); [Spiralia](#); [Lophotrochozoa](#); [Mollusca](#); [Bivalvia](#); [Autobranchia](#); [Pteriormorphia](#); [Pectinida](#); [Pectinoidea](#); [Pectinidae](#); [Mizuhopecten](#)

图 3. 虾夷扇贝在 Taxonomy 数据库中的进化层级

NCBI Taxonomy Browser

Search for as ☒ lock

Display levels using filter:

☐ Nucleotide ☒ Protein ☐ Structure ☐ Genome ☐ Popset ☐ SNP ☐ Conserved Domains ☐ GEO Datasets ☐ PubMed Central

☐ Gene ☐ HomoloGene ☐ SRA Experiments ☐ LinkOut ☐ BLAST ☐ GEO Profiles ☐ Protein Clusters ☐ Identical Protein Groups ☐ SPARCLE

☐ Bio Project ☐ Bio Sample ☐ Bio Systems ☐ Assembly ☐ dbVar ☐ Genetic Testing Registry ☐ Host ☐ Viral Host ☐ Probe

☐ PubChem BioAssay

[Lineage \(full\)](#): [cellular organisms](#); [Eukaryota](#); [Opisthokonta](#); [Metazoa](#); [Eumetazoa](#); [Bilateria](#); [Protostomia](#); [Spiralia](#); [Lophotrochozoa](#)

- o [Mollusca](#) (mollusks) [1,065,795](#) [Click on organism name to get more information.](#)
 - o [Aplacophora](#) [486](#)
 - o [Bivalvia](#) (bivalves) [459,773](#)
 - o [Cephalopoda](#) (cephalopods) [169,947](#)
 - o [Gastropoda](#) (gastropods) [431,163](#)
 - o [Monoplacophora](#) [62](#)
 - o [Polyplacophora](#) (chitons) [4,190](#)
 - o [Scaphopoda](#) (tusk shells) [127](#)
 - o [unclassified Mollusca](#) [16](#)
 - o [environmental samples](#) [31](#)

图 4. NCBI 的 Protein 数据库中属于 Mollusca 进化层级的蛋白序列

图 5. 本研究中将虾夷基因划分为 11 个进化层级 (PS1-PS11)

2. 定义基因年龄

将虾夷扇贝基因组的全部基因与 NCBI 的 Protein 数据库做同源比对, 以基因能比对到的最古老的物种所在的进化层级定义基因年龄, 在任何物种中都找不到的定义为虾夷扇贝特有的新基因, 划为 PS11。同源比对的命令如下:

```
$ blastp -db database.fasta -query genome_protein.fasta -evalue 1e-3 -out output.txt -max_target_seqs 1 -num_threads 8 -outfmt 6
```

三、转录组年龄指数 (TAI) 分析

参考 Wang *et al.*, (2020)发表的文章中关于 TAI 分析的计算公式, 使用 R (version 3.5.0) 软件中的 *myTAI* 软件包进行 TAI 相关分析。

1. 通过 CRAN 镜像安装 *myTAI* (0.8.0) 软件包

```
>source("http://bioconductor.org/biocLite.R")
>biocLite('myTAI')
```

2. 整理 TAI 分析所需的输入文件

输入文件的第一列是基因对应的进化层级, 第二列是 GeneID, 第三列之后是基因在各发育时期的表达量, 如下图所示。输入文件保存为 csv 格式 (图 6)。

Phylostratum	geneID	Zyg	2-8cell	Bla	Gas	Tro	Dst	E.umbo	M.umbo	P.umbo	Ped	Spa	Juv	A-1y	A-2y	A-3y	A-4y
1	scaffold11149.11	28.23	18.6	8.38	10.47	15.4	11.15	4.65	5.87	5.67	12.79	3.45	3.18	9.03	5.265	7.975	7.67
2	scaffold11149.13	34.78	24.64	15.58	17.65	15.85	13.1	7.54	11.97	11.13	10.75	10.36	3.035	9.21	12.27	7.57	9.44
4	scaffold11149.16	57.74	62.74	49.55	45.56	22	16.92	8.82	8.56	10.6	12.16	14.42	15.34	24.95	16.545	18.675	16.245
1	scaffold11149.21	235.61	178.61	230.73	230.61	170.53	127.26	51.4	57.84	51.79	36.15	65.95	24.485	18.44	20.74	14.915	30.555
1	scaffold11149.22	7.12	4.11	12.61	7.51	4.98	2.85	0.53	0.88	1.07	1.7	1.27	2.67	3.8	6.035	5.225	2.515
1	scaffold11149.23	24.78	12.2	2.24	2.1	1.45	0.81	0.16	0.43	0.07	0.74	0.16	4.135	3.245	4.075	5.645	5.375
2	scaffold11149.24	0.21	0	0.25	0.23	0.36	0.17	0.18	0.03	0.09	0	0.11	0.43	1.1	0.885	1.44	1.195
3	scaffold11149.25	78.89	83.87	46.49	40.42	24.94	26.37	9.05	32.22	23.92	59.32	28.29	56.85	56.315	43.705	62.045	51.285
1	scaffold11149.26	55.26	39.25	95.12	112.56	133.16	140.72	19.1	71.12	34.12	154.22	50.82	286.725	379.37	219.59	388.415	319.32
2	scaffold11149.27	1.33	0.97	1.86	1.35	0.51	0.56	0.55	0.37	0.13	0.33	0.24	8.795	6.1	11.23	14.285	12.96
8	scaffold11149.28	31.79	32.58	129.39	141.92	104.34	81.19	23	43.69	34.74	43.61	45.81	66.075	59.135	56.83	78	74.97
11	scaffold11149.29	19.16	16.4	74.12	69.85	54.28	37.56	13.05	34.83	17.26	15.31	54.36	19.48	23.84	31.125	37.675	24.87
1	scaffold11149.30	2.32	1.48	0.22	0.4	2.02	3.95	1.81	3.48	3.83	5.39	5.87	12	6.415	3.985	6.605	6.385
10	scaffold11149.31	12.84	8.88	9.98	12.85	12.46	7.41	1.41	2.55	1.89	3.46	2.84	11.475	11.695	11.605	19.02	16.315
3	scaffold11149.32	0.84	0.5	0.68	0.98	1.4	1.45	0.71	0.75	0.65	1.16	0.72	3.415	2.18	2.26	3.44	3.405
2	scaffold11149.33	122.06	124.74	111.47	112.81	124.55	117.41	145.53	112.83	138.01	74.67	161.48	71.285	58.84	47.87	45.645	40.85
6	scaffold11149.34	0.11	0.28	0	0	0	0.3	3.47	4.9	1.82	4.49	0.75	2.865	0.435	0.885	0.715	0.07
2	scaffold11149.35	0	0	0	0	0.51	0.72	0	0	0	0.52	0	4.025	16.88	5.935	14.46	12.985
3	scaffold11149.36	0	0	0	1.02	18.45	49.72	17.41	23.79	8.88	34.18	2.18	21.875	9.965	7.37	8.19	10.965
7	scaffold11149.37	0	0	0.34	0	0	6.33	71.57	64.72	51.56	59.92	20.31	66.78	8.19	10.63	17.785	32.89
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...

图 6. TAI 输入文件格式

3. 调用 *myTAI* 软件包进行 TAI 计算（图 7）

```
>library(myTAI)
>data<-read.csv('TAI 输入文件.csv', header=T)
>dataset<-as.data.frame(data)
>TAI(dataset)
```

```
> library(myTAI)
> data<-read.csv('TAI输入文件.csv', header=T)
> dataset<-as.data.frame(data)
> TAI(dataset)

      Zyg X2.8cell      Bla      Gas      Tro      Dst      E.umbo      M.umbo
2.055495 2.502822 2.233624 2.532967 3.892229 3.545192 2.995443 2.517783
      P.umbo      Ped      Spa      Juv      A.1y      A.2y      A.3y      A.4y
2.422162 2.389675 2.587175 2.497997 2.474285 2.406495 2.431323 2.601544
```

图 7. 虾夷扇贝 16 个发育时期的 global TAI 值

注：这里需要着重强调的是，根据 TAI 计算公式，

$$TAI_s = \frac{\sum_{i=1}^n p_{si} e_{is}}{\sum_{i=1}^n e_{is}} = p_{s1} \frac{e_{1s}}{e_{1s} + e_{2s} + \dots + e_{ns}} + p_{s2} \frac{e_{2s}}{e_{1s} + e_{2s} + \dots + e_{ns}} + \dots + p_{sn} \frac{e_{ns}}{e_{1s} + e_{2s} + \dots + e_{ns}}$$

p_{si} 表示基因 i 的进化层级， e_{is} 表示基因 i 在发育时期 s 时的表达量， n 表示用于 TAI 计算的基因总数。

当计算虾夷扇贝某一发育阶段的 global TAI 时，输入文件中包含 PS1-PS11 的全部基因。在计算祖先时期 TAI 值时，我们的计算方法与 Domazet-Lošo and Tautz (Domazet-Lošo and Tautz, 2010) 不同。我们通过逐层去除相对更年轻的基因，来推

演祖先时期的转录组状态。例如，要计算扇贝转录组在双壳纲祖先 (PS10) 时的 TAI，则在输入文件中完全去除 PS11 基因。以此类推，在计算处于进化层级 N 时的 TAI 值时，起源晚于 N 的基因总是从输入文件中移除（即，从计算公式的“分子”和“分母”中同时移除）。

4. 发育时期 TAI 变化曲线及显著性分析（图 8，图 9）

```
>PlotSignature (ExpressionSet=dataset,
                measure="TAI",
                TestStatistic="FlatLineTest",
                xlab="Ontogeny",
                ylab="TAI")
```

用于显著性分析的 FlatLineTest 功能其具体参数如下：

```
FlatLineTest ( ExpressionSet = dataset,
                permutations = 1000,
                plotHistogram = TRUE )
```

图 8. 虾夷扇贝发育时期 global TAI 变化曲线及显著性分析

图 9. 虾夷扇贝处于不同进化层级时的 TAI 曲线

5. 计算每个基因对 global TAI 的贡献值，输出 csv 格式的结果文件（图 10）

```
>partialTAI<-pMatrix(dataset)
```

```
>head(partialTAI)
```

```
>write.csv(partialTAI,'pMatrix.csv')
```

注：根据 TAI 计算公式，所有基因在某一发育时期的 TAI 贡献值之和等于该时期的 global TAI 值。

```
> partialTAI<-pMatrix(dataset)
> head(partialTAI)
```

	Zyg	X2.8cell	Bla	Gas	Tro	Dst	E.umbo	M.umbo	P.umbo
hete_35829218.1	0.000000e+00	0.000000e+00	0.00e+00	0.000000e+00	0.000000e+00	0.000000e+00	0	0	0
hete_35832346.1	2.172001e-05	3.185999e-05	0.00e+00	1.590001e-05	5.520004e-06	1.044001e-05	0	0	0
hete_35838452.1	0.000000e+00	0.000000e+00	0.00e+00	0.000000e+00	0.000000e+00	0.000000e+00	0	0	0
hete_35851862.1	0.000000e+00	0.000000e+00	0.00e+00	0.000000e+00	0.000000e+00	0.000000e+00	0	0	0
hete_35859914.1	0.000000e+00	0.000000e+00	0.00e+00	0.000000e+00	0.000000e+00	0.000000e+00	0	0	0
hete_35861874.1	0.000000e+00	3.199999e-06	2.55e-06	0.000000e+00	0.000000e+00	1.700002e-06	0	0	0
	Ped Spa	Juv	A.1y	A.2y	A.3y	A.4y			
hete_35829218.1	0.000000e+00	0.000000e+00	0.000e+00	0.000e+00	0.000e+00	0.000000e+00			
hete_35832346.1	1.559999e-06	2.700001e-06	1.152e-05	2.070e-06	1.860e-06	3.660001e-06			
hete_35838452.1	0.000000e+00	0.000e+00	0.000e+00	0.000e+00	0.000e+00	0.000000e+00			
hete_35851862.1	0.000000e+00	0.000e+00	0.000e+00	0.000e+00	0.000e+00	0.000000e+00			
hete_35859914.1	7.589995e-06	4.966501e-05	0.000e+00	3.762e-05	4.114e-05	0.000000e+00			
hete_35861874.1	0.000000e+00	0.000e+00	0.000e+00	0.000e+00	0.000e+00	0.000000e+00			

图 10. 虾夷扇贝各基因对 global TAI 的贡献值

6. 计算不同进化层级对 TAI 的相对贡献（图 11）

进化层级 N 中的基因对转录组 TAI 的相对贡献用 $PS_{S(N)}$ 表示，计算公式为 $PS_{S(N)} =$

$$TAI_{S(N)} - TAI_{S(N-1)} = \left(1 * \frac{E_{1s}}{E_{1s} + E_{2s} + \dots + E_{Ns}} + 2 * \frac{E_{2s}}{E_{1s} + E_{2s} + \dots + E_{Ns}} + \dots + N * \frac{E_{Ns}}{E_{1s} + E_{2s} + \dots + E_{Ns}} \right) - \left(1 * \frac{E_{1s}}{E_{1s} + E_{2s} + \dots + E_{(N-1)s}} + 2 * \frac{E_{2s}}{E_{1s} + E_{2s} + \dots + E_{(N-1)s}} + \dots + (N-1) * \frac{E_{(N-1)s}}{E_{1s} + E_{2s} + \dots + E_{(N-1)s}} \right)$$

这里 $E_{Ns} = \sum_{i=1}^n e_{is} = e_{1s} + e_{2s} + \dots +$

e_{ns} ，表示属于进化层级 N 的所有基因在发育时期 s 中的表达量之和， e_{is} 表示基因 i 在发育时期 s 时的表达量， n 表示进化层级 N 的基因数。

图 11. 不同进化层级基因对扇贝担轮幼虫 TAI 的相对贡献

致谢

本研究受国家自然科学基金项目(31871499 & 31900369)，山东省泰山学者计划项目和山东省博士后创新项目资助。

参考文献

1. Domazet-Lošo, T., Brajković, J. and Tautz, D. (2007). [A phylostratigraphy approach to uncover the genomic history of major adaptations in metazoan lineages](#). *Trends in Genetics*, 23(11):533-539.
2. Domazet-Lošo, T. and Tautz, D. (2010). [A phylogenetically based transcriptome age index mirrors ontogenetic divergence patterns](#). *Nature* 468(7325): 815-818.

3. Drost, H.G., Gabel, A., Liu, J., Quint, M. and Grosse, I. (2018). [myTAI: evolutionary transcriptomics with R](#). *Bioinformatics*, 34(9):1589-1590.
4. Futo, M., Opašić, L., Koska, S., Čorak, N., Široki, T., Ravikumar, V., Thorsell, A., Lenuzzi, M., Kifer, D., Domazet-Lošo, M., Vlahoviček, K., Mijakovic, I. and Domazet-Lošo, T. (2020). [Embryo-like features in developing *Bacillus subtilis* biofilms](#). *Molecular Biology and Evolution*, 38(1):31-47.
5. Hemmrich, G., Khalturin, K., Boehm, A.M., Puchert, M., Anton-Erxleben, F., Wittlieb, J., Klostermeier, U.C., Rosenstiel, P., Oberg, H.H., Domazet-Lošo, T., Sugimoto, T., Niwa, H. and Bosch, T.C.G. (2012). [Molecular signatures of the three stem cell lineages in Hydra and the emergence of stem cell function at the base of multicellularity](#). *Molecular Biology and Evolution*, 29(11):3267-3280.
6. Wang, J., Zhang, L., Lian, S., Qin, Z., Zhu, X., Dai, X., Huang, Z., Ke, C., Zhou, Z., Wei, J., Liu, P., Hu, N., Zeng, Q. and Dong, B. (2020). [Evolutionary transcriptomics of metazoan biphasic life cycle supports a single intercalation origin of metazoan larvae](#). *Nature Ecology & Evolution*, 4(5): 725-736.