Example alignments

sadie

2/5/2020

```
library(readr)
library(dplyr)
library(ggplot2)
library(stringr)
library(plotly)
library(tidyr)
```

```
ex <- read_csv("~/Downloads/example_alignments_2_5_2020.csv")
```

```
## Warning: Duplicated column names deduplicated: 'Substitution rate from
## nucleotide A to nucleotide C' => 'Substitution rate from nucleotide A to
## nucleotide C_1' [34], 'Substitution rate from nucleotide A to nucleotide
## G' => 'Substitution rate from nucleotide A to nucleotide G_1' [35],
## 'Substitution rate from nucleotide A to nucleotide T' => 'Substitution
## rate from nucleotide A to nucleotide T_1' [36], 'Substitution rate from
## nucleotide C to nucleotide G' => 'Substitution rate from nucleotide C to
## nucleotide G_1' [37], 'Substitution rate from nucleotide C to nucleotide
## T' => 'Substitution rate from nucleotide C to nucleotide T_1' [38],
## 'Substitution rate from nucleotide G to nucleotide T' => 'Substitution rate
## from nucleotide G to nucleotide T_1' [39], 'non-synonymous/synonymous rate
## ratio' => 'non-synonymous/synonymous rate ratio_1' [40], 'rate at which 3
## nucleotides are changed instantly within a single codon between synonymous
## codon islands' => 'rate at which 3 nucleotides are changed instantly within
## a single codon between synonymous codon islands_1' [43], 'GDD rate category
## 1.triple' => 'GDD rate category 1.triple_1' [44], 'GDD rate category
## 2.triple' => 'GDD rate category 2.triple_1' [45], 'GDD rate category
## 3.triple' => 'GDD rate category 3.triple_1' [46], 'Mixture auxiliary
## weight for GDD category 1.triple' => 'Mixture auxiliary weight for GDD
## category 1.triple_1' [47], 'Mixture auxiliary weight for GDD category
## 2.triple' => 'Mixture auxiliary weight for GDD category 2.triple_1' [48],
## 'distribution.triple' => 'distribution.triple_1' [49], 'Substitution rate
## from nucleotide A to nucleotide C' => 'Substitution rate from nucleotide A
## to nucleotide C_2' [52], 'Substitution rate from nucleotide A to nucleotide
## G' => 'Substitution rate from nucleotide A to nucleotide G_2' [53],
## 'Substitution rate from nucleotide A to nucleotide T' => 'Substitution
## rate from nucleotide A to nucleotide T_2' [54], 'Substitution rate from
## nucleotide C to nucleotide G' => 'Substitution rate from nucleotide C to
## nucleotide G_2' [55], 'Substitution rate from nucleotide C to nucleotide
## T' => 'Substitution rate from nucleotide C to nucleotide T_2' [56],
## 'Substitution rate from nucleotide G to nucleotide T' => 'Substitution rate
## from nucleotide G to nucleotide T_2' [57], 'non-synonymous/synonymous rate
## ratio' => 'non-synonymous/synonymous rate ratio_2' [58], 'Substitution rate
## from nucleotide A to nucleotide C' => 'Substitution rate from nucleotide A
## to nucleotide C_3' [68], 'Substitution rate from nucleotide A to nucleotide
## G' => 'Substitution rate from nucleotide A to nucleotide G_3' [69],
## 'Substitution rate from nucleotide A to nucleotide T' => 'Substitution
## rate from nucleotide A to nucleotide T_3' [70], 'Substitution rate from
```

```
## nucleotide C to nucleotide G' => 'Substitution rate from nucleotide C to
## nucleotide G_3' [71], 'Substitution rate from nucleotide C to nucleotide
## T' => 'Substitution rate from nucleotide C to nucleotide T_3' [72],
## 'Substitution rate from nucleotide G to nucleotide T' => 'Substitution rate
## from nucleotide G to nucleotide T_3' [73], 'non-synonymous/synonymous rate
## ratio' => 'non-synonymous/synonymous rate ratio_3' [74]
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   `File name` = col_character(),
##   distribution.triple = col_character(),
##   distribution.triple_1 = col_character(),
##   distribution.double = col_character(),
##   distribution.single = col_character(),
##   `Branch Attributes - MG94 with double and triple instantaneous substitutions` = col_character(),
##   `Branch Attributes - MG94 with double instantaneous substitutions` = col_character(),
##   `Branch Attributes - Standard MG94` = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```
dh <- which(ex$`Double-hit vs single-hit - p-value`<=0.05) %>% ex$`File name`[.]
dh
```

```
## [1] "HepatitisD.nex"                 "camelid.nex"
## [3] "InfluenzaA.nex"                 "lysin.nex"
## [5] "yokoyama.rh1.cds.mod.1-990.nex" "flavNS5.nex"
## [7] "bglobin.nex"                    "HIV_RT.nex"
## [9] "adh.nex"
```

```
th <- which(ex$`Triple-hit vs double-hit - p-value`<=0.05) %>% ex$`File name`[.]
th
```

```
## [1] "camelid.nex"                    "HIVvif.nex"
## [3] "lysin.nex"                      "yokoyama.rh1.cds.mod.1-990.nex"
## [5] "bglobin.nex"
```

```
thi <- which(ex$`Triple-hit vs Triple-hit-island - p-value`<=0.05) %>% ex$`File name`[.]
thi
```

```
## [1] "camelid.nex"                    "HIVvif.nex"
## [3] "lysin.nex"                      "yokoyama.rh1.cds.mod.1-990.nex"
```

```
th.dh <- intersect(th,dh)
```

these files camelid.nex, lysin.nex, yokoyama.rh1.cds.mod.1-990.nex, bglobin.nex are

```
only.dh <- setdiff(dh,th)
only.dh
```

```
## [1] "HepatitisD.nex" "InfluenzaA.nex" "flavNS5.nex"    "HIV_RT.nex"
## [5] "adh.nex"
```

```
temp <- ex %>% filter(`File name` %in% only.dh) %>% select(`File name`,starts_with("rate at which"))
```

read in json for Influenza A

```
library(jsonlite)
InfA <- fromJSON("~/Downloads/example_alignments/data/InfluenzaA.nex.FITTER.json")
```

```
HIV_RT <- fromJSON("~/Downloads/example_alignments/data/HIV_RT.nex.FITTER.json")
```

```
InfA$`Site Log Likelihood`$`MG94 with double instantaneous substitutions` %>% which.min()
```

```
## [1] 226
```

```
HIV_RT$`Site Log Likelihood`$`MG94 with double instantaneous substitutions` %>% which.min()
```

```
## [1] 103
```

```
InfA$`Evidence Ratios`$`Two-hit` %>% max()
```

```
## [1] 7.320218
```

```
InfA$`Evidence Ratios`$`Two-hit` %>% min()
```

```
## [1] 0.5524878
```

```
InfA$`Evidence Ratios`$`Two-hit` %>% mean()
```

```
## [1] 1.132726
```

```
InfA$`Evidence Ratios`$`Three-hit` %>% max()
```

```
## [1] 1.947041
```

```
InfA$`Evidence Ratios`$`Three-hit` %>% min()
```

```
## [1] 0.8366965
```

```
InfA$`Evidence Ratios`$`Three-hit` %>% mean()
```

```
## [1] 1.001552
```

```
InfA$`Evidence Ratios`$`Three-hit islands vs 2-hit` %>% max()
```

```
## [1] 1.024654
```

```
InfA$`Evidence Ratios`$`Three-hit` %>% min()
```

```
## [1] 0.8366965
```

```
InfA$`Evidence Ratios`$`Three-hit` %>% mean()
```

```
## [1] 1.001552
```

```
HIV_RT$`Evidence Ratios`$`Two-hit` %>% max()
```

```
## [1] 2461.584
```

```
HIV_RT$`Evidence Ratios`$`Two-hit` %>% which.max()
```

```
## [1] 163
```

```
HIV_RT$`Evidence Ratios`$`Two-hit` %>% min()
```

```
## [1] 0.002059683
```

```
HIV_RT$`Evidence Ratios`$`Two-hit` %>% mean()
```

```
## [1] 12.79567
```

```
HIV_RT$`Evidence Ratios`$`Two-hit` %>% median()
```

```
## [1] 1.021469
```

```
HIV_RT$`Evidence Ratios`$`Three-hit` %>% max()
```

```
## [1] 5.606007
```

```
HIV_RT$`Evidence Ratios`$`Three-hit` %>% min()
```

```
## [1] 0.7499284
```

```
HIV_RT$`Evidence Ratios`$`Three-hit` %>% mean()
```

```
## [1] 1.020086
```

```
HIV_RT$`Evidence Ratios`$`Three-hit`[163]
```

```
## [1] 0.9343057
```

```
HIV_RT$`Evidence Ratios`$`Three-hit vs three-hit islands`[163]
```

```
## [1] 0.9296088
```

```
HIV_RT$`Evidence Ratios`$`Three-hit islands vs 2-hit` %>% max()
```

```
## [1] 1.016396
```

```
HIV_RT$`Evidence Ratios`$`Three-hit` %>% min()
```

```
## [1] 0.7499284
```

```
HIV_RT$`Evidence Ratios`$`Three-hit` %>% mean()
```

```
## [1] 1.020086
```

Site	ER (2 vs 1)	ER (3 vs 2)	ER (3-island vs 2)	ER (3-island vs 3)	Substitutions
48	106.3661	0.7761	0.9988	0.7770	ACA->AAA(1), TCA->AAA(4)ACA(8)CAA(1)GAA(2)TCC(1)TCG(2), T...
64	13.1124	0.9154	0.9885	0.9261	AAA->AAA(2), AAG->AAA(8)AGG(5)CAC(1)TCG(1), AGG->AAA(3)
69	7.8381	2.0887	1.0014	2.0858	AAC->AAA(1), ACA->AAA(1), ACC->AAA(2)AAC(1)ACT(1), ACT->A...
72	19.9667	0.9422	1.0042	0.9383	AGA->AAA(14)AGG(5)GAA(1), AGG->AAA(2)AGA(5)
75	27.3292	0.8584	1.0006	0.8579	GTA->AAA(9)ATA(4)ATG(1)GTG(3)GTT(2)TCA(1), GTG->AAA(2)GTA...
151	74.1524	0.8338	0.9990	0.8347	CAA->AAA(1), CAG->AAA(10)ATG(4)CAA(18)
162	28.9096	1.1550	0.9816	1.1767	ACT->AAA(1), AGC->AAA(1), AGT->AAA(21)ACT(1)AGC(7)CAT(1)G...
163	2461.5837	0.9343	1.0051	0.9296	AGC->AAA(13)AGT(9)TCC(1), AGT->AAA(1)AGC(1)

Site	ER (2 vs 1)	ER (3 vs 2)	ER (3-island vs 2)	ER (3-island vs 3)	Substitutions
174	8.2639	0.8722	1.0063	0.8667	AAA->AGA(1)CAA(2), AGA->AAA(3), CAA->AAA(22)AAC(1)AAG(1)A...
181	57.3924	0.7499	0.9986	0.7510	TAC->AAA(2)TAT(2)TGC(1), TAT->AAA(14)ATT(3)TAC(16)TGT(5),...
188	903.6130	0.7900	0.9995	0.7905	TAT->AAA(7)CAT(2)CTT(1)TAC(1)TTA(2)TTG(1)
215	205.2119	0.8392	1.0005	0.8388	ACC->AAA(14)AAC(1)ACA(1)ACT(6)GAC(2)TAC(1), ACT->AAA(1)AC...
219	8.3712	3.2710	1.0031	3.2609	AAA->AAA(15)AAG(15)CAG(1)GAA(1)TGG(1), AAG->AAA(2)
228	10.3951	5.6060	1.0026	5.5916	AAG->AAA(1), CTC->AAA(1), CTG->AAA(1), CTT->AAA(17)AAG(1)...

```
object <- '|   Site   |       ER (2 vs 1)       |       ER (3 vs 2)       |   ER (3-island vs 2)    |
|    48    |          106.3661        |           0.7761        |           0.9988        |           0.7
|    64    |           13.1124        |           0.9154        |           0.9885        |           0.9
|    69    |            7.8381        |           2.0887        |           1.0014        |           2.0
|    72    |           19.9667        |           0.9422        |           1.0042        |           0.9
|    75    |           27.3292        |           0.8584        |           1.0006        |           0.8
|   151    |           74.1524        |           0.8338        |           0.9990        |           0.8
|   162    |           28.9096        |           1.1550        |           0.9816        |           1.1
|   163    |         2461.5837        |           0.9343        |           1.0051        |           0.9
|   174    |            8.2639        |           0.8722        |           1.0063        |           0.8
|   181    |           57.3924        |           0.7499        |           0.9986        |           0.7
|   188    |          903.6130        |           0.7900        |           0.9995        |           0.7
|   215    |          205.2119        |           0.8392        |           1.0005        |           0.8
|   219    |            8.3712        |           3.2710        |           1.0031        |           3.1
|   228    |           10.3951        |           5.6060        |           1.0026        |           5.5
HIV_RT_ER_DF <- read_delim(object, delim = '|', trim_ws = TRUE)
```

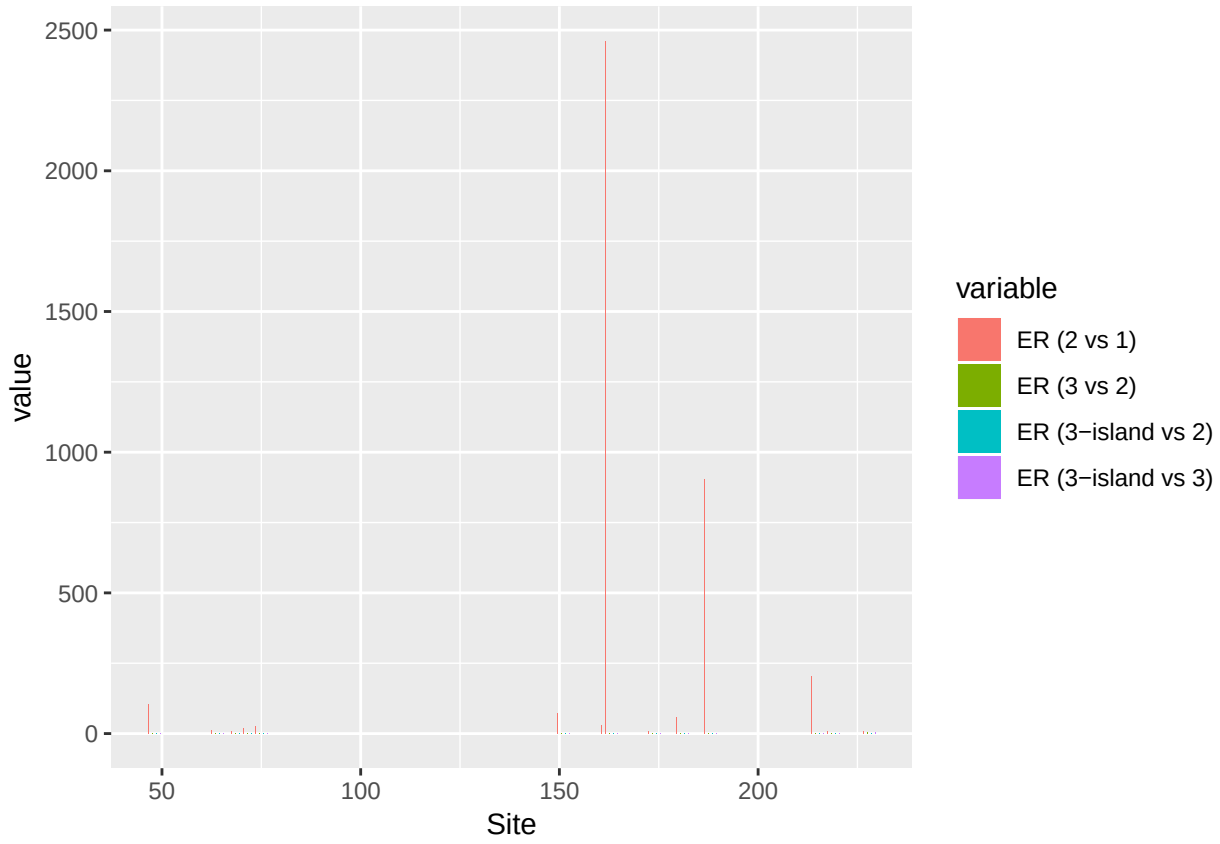
```
## Warning: Missing column names filled in: 'X1' [1], 'X8' [8]
```

```
library(reshape2)
```

```
##
## Attaching package: 'reshape2'
```

```
## The following object is masked from 'package:tidyr':
##
##     smiths
```

```
HIV_RT_ER_DF %>%  melt(id.vars = "Site", measure.vars = c("ER (2 vs 1)","ER (3 vs 2)","ER (3-island vs 2
  geom_col(aes(x = Site, y = value, fill = variable), position = position_dodge(4))
```



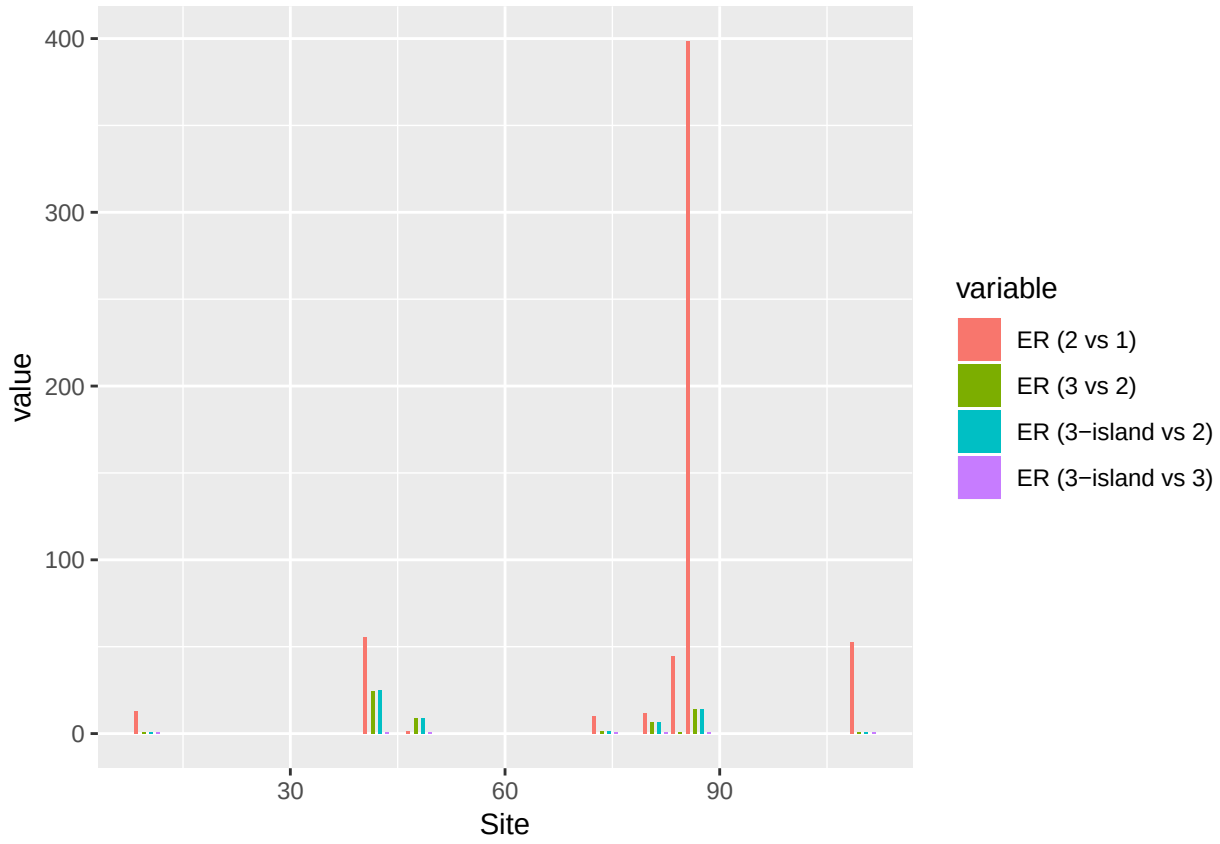

```
  #geom_bar(aes(y = value, fill = variable))
```

```
bglobin.obj <- "|   Site   |       ER (2 vs 1)       |       ER (3 vs 2)       |   ER (3-island vs 2)
|    10    |          12.9735        |           0.7324        |           0.7297        |           1.0
|    42    |          55.7399        |          24.5501        |          24.7731        |           0.9
|    74    |           9.9600        |           1.1959        |           1.1919        |           1.0
|    81    |          11.9079        |           6.7999        |           6.8504        |           0.9
|    85    |          44.4935        |           0.7913        |           0.7928        |           0.9
|    87    |         398.6468        |          14.1893        |          14.1004        |           1.0
|   110    |          52.8830        |           0.8326        |           0.8300        |           1.0
|    48    |           1.6586        |           8.7580        |           8.7911        |           0.9

BG_ER_DF <- read_delim(bglobin.obj, delim = '|', trim_ws = TRUE)
```

```
## Warning: Missing column names filled in: 'X1' [1], 'X8' [8]
```

```
BG_ER_DF %>%  melt(id.vars = "Site", measure.vars = c("ER (2 vs 1)","ER (3 vs 2)","ER (3-island vs 2)",'
  geom_col(aes(x = Site, y = value, fill = variable), position = position_dodge(4))
```




```
#geom_bar(aes(y = value, fill = variable))
```

HIV vif single site

Site	ER (2 vs 1)	ER (3 vs 2)	ER (3-island vs 2)	ER (3-island vs 3)	Substitutions
6	2.2495	11.1569	1.0014	11.1411	CAG->GCA(1)

COX1 ### 2 individual sites which showed sufficiently strong preference for multiple-hit models

Site	ER (2 vs 1)	ER (3 vs 2)	ER (3-island vs 2)	ER (3-island vs 3)	Substitutions
271	9.4083	1.0009	1.0011	0.9998	GTC->ACC(1)GTA(1)GTT(6), GTT->GTA(1)
483	7.5306	1.0012	1.0017	0.9995	GAA->GAG(1)TCA(1), TCA->TCC(1)TCG(1)

yokoyama

Site	ER (2 vs 1)	ER (3 vs 2)	ER (3-island vs 2)	ER (3-island vs 3)	Substitutions
13	52.2073	0.5929	0.8707	0.6810	ATG->TTC(2)

Site	ER (2 vs 1)	ER (3 vs 2)	ER (3-island vs 2)	ER (3-island vs 3)	Substitutions
14	150.9647	1.8552	0.7652	2.4244	ATA->CTC(1), GTA->GTG(1)GTT(1), GTG->TCT(1), TCA->ATA(1)G...
19	27.9752	0.6240	1.0113	0.6171	GTG->ATA(1)ATC(1)CTG(1)GTA(2)GTT(2), GTT->ATC(1)ATT(4)GTA...
26	37.3421	0.7866	0.9703	0.8107	TAC->GCC(1)TAT(1), TAT->TAC(8)
35	8.0963	1.0832	0.9637	1.1240	GCA->GCC(1), GCG->GCA(3)GTG(1), TGG->GCG(1)
45	26.7908	0.7598	0.9401	0.8082	TTC->ATG(1)TTT(2)
46	14.4686	2.1330	0.9158	2.3290	CTG->ATG(1)CTT(1)TTG(1), TTC->ACG(1)CTG(2)TTT(1)
50	18.9419	1.0181	1.0734	0.9485	ACT->ACC(1), ATT->ATC(2)GTC(1)GTG(1)TTT(2), CTT->CTC(2)CT...
84	46.5318	0.5237	0.8465	0.6187	CTC->CAC(1)CAT(1)TTG(3)
93	52.8115	2.9123	0.9325	3.1230	ACC->ACA(2)ACT(1)GTC(1)GTG(1)
96	37.1341	3.2911	0.9412	3.4968	TAC->GTG(1)TAT(2)
107	9.7138	1.2380	0.9826	1.2599	CAA->CAG(1), CCA->ACA(1)CAA(1)CCC(3)CGC(1)GAA(1)GTA(1), C...
108	101.0508	1.8546	0.8973	2.0669	ACA->ACG(1), ACG->ACA(2)ACC(1)ACT(3)CTG(1)GTA(1), CTC->TT...
122	17.8947	1.9526	0.9081	2.1502	GAA->CAA(1)GAG(2), GAG->ATC(1)ATG(1)CAG(1)GAA(3)
136	71.0212	0.6404	0.9599	0.6671	TAC->TAT(1), TAT->TAC(2), TGG->TAT(2)
144	318.8699	0.7403	1.1404	0.6491	AGC->ACC(1)GCC(2)TCC(1), GCC->GAA(1)TCC(1)
149	22.8614	2.3053	0.9619	2.3966	GGG->ACT(1)CAG(1)GGA(5)GGC(1)GGT(2), GGT->GGC(1)
158	14.1189	3.0228	0.9465	3.1938	GCA->ACG(1)ATG(1), GCC->ACC(1)AGT(1)ATC(1)ATG(1)GCA(1)GCG...
160	28.9554	1.2120	1.3124	0.9235	ACC->ACA(2)ACT(1)TCC(2), TCC->AGC(1)
162	42.3805	0.5871	1.1305	0.5193	ATC->ACC(1)ATA(1)ATT(1)GTC(3)TTG(1), GGT->GCC(1), GTA->CT...
163	20.8949	0.8189	0.9566	0.8560	ATG->GCG(1)TTG(1)

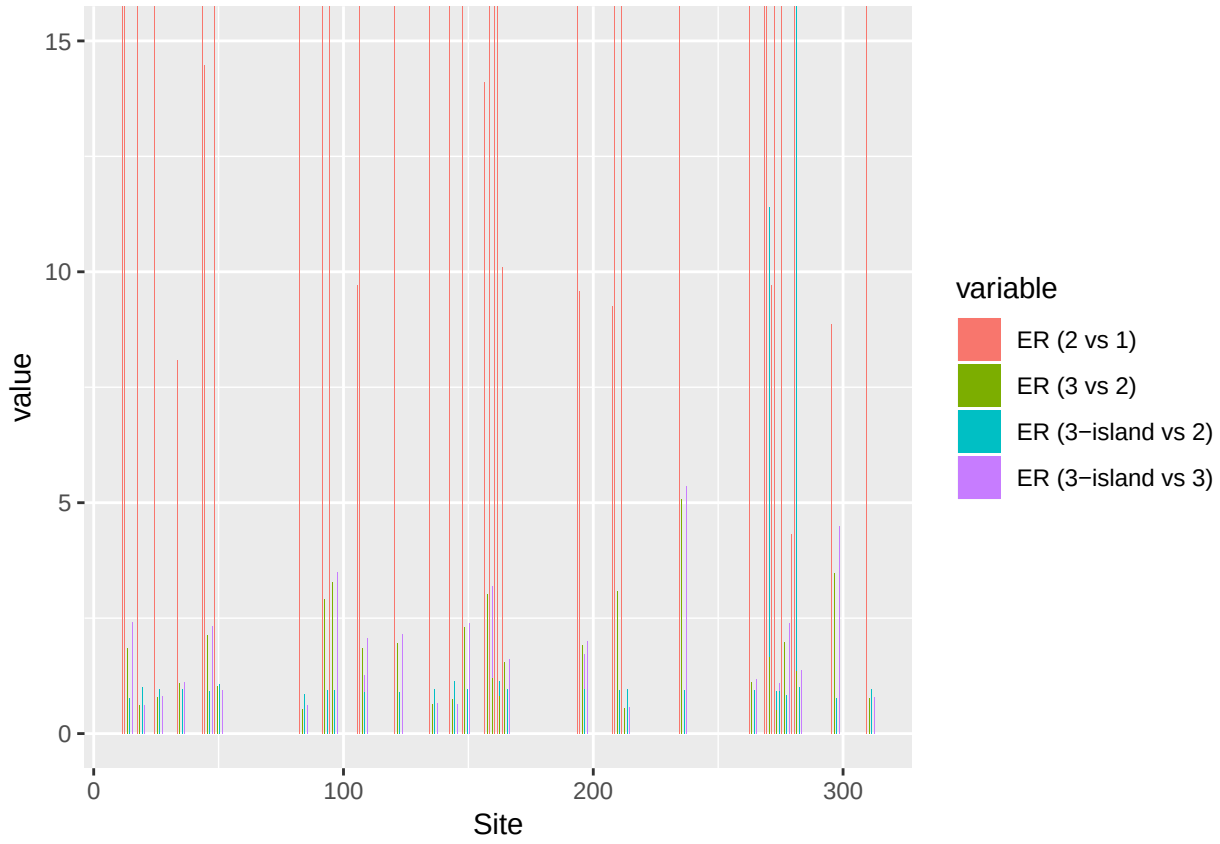
Site	ER (2 vs 1)	ER (3 vs 2)	ER (3-island vs 2)	ER (3-island vs 3)	Substitutions
165	10.1073	1.5528	0.9650	1.6090	CTG->ATG(1), TCC->GGC(1)TGC(1), TCG->TCC(2), TTG->AAT(1)A...
195	41.9450	1.6128	0.9415	1.7131	AAG->AAC(1)ACC(1)CAC(1)GCC(1)GCG(1), CAC->CGC(1), GCC->AC...
196	9.5755	1.9173	0.9614	1.9942	CAG->CCC(1)GAG(1), CCC->CAG(1), CCG->CCA(1), CCT->CCC(1)C...
209	9.2534	1.0200	1.0889	0.9367	ACC->ACA(1), ATC->ACC(1)ATA(1), ATT->CTT(1), GTC->ACC(1)A...
210	78.1724	3.0818	0.9327	3.3040	CTC->GTG(1)TTA(1), GTC->GTG(2)GTT(3)TGC(2), GTT->GTC(1), ...
213	22.7746	0.5560	0.9739	0.5709	ACC->ATC(2)ATG(1)GTC(1)TCC(2), ATC->ATA(1)ATT(1)CTG(2)TCC...
236	45.2105	5.0861	0.9490	5.3591	CAG->CAA(1)GCC(1), GCC->GCT(1)
264	32.4793	1.1110	0.9455	1.1750	ACC->ACA(1), TGC->ACC(1)TGT(4), TGT->TGC(1)
270	216.0457	13.2258	11.3947	1.1607	AGC->AGT(5)GGA(1)GGC(3)GGG(1)TAT(1)TCA(1), GGC->GGA(1)
271	51.2877	1.6525	0.9532	1.7336	GTG->ACC(1)ACG(1)GTA(1)GTC(1)GTT(2), GTT->GTC(1)
273	9.7226	0.9912	0.9105	1.0887	TGG->GCG(1)TTC(1)
274	95.5793	0.5156	0.9282	0.5555	TAC->TAT(2)TTC(2), TAT->TAC(4)TGG(3)
277	121.9976	1.9885	0.8298	2.3963	ACA->ACG(1)TGT(1), ACC->ACA(1)ACT(1)AGC(1)TGC(2)TTC(1)
282	94.8591	1.3634	0.9965	1.3681	ACA->ACT(2), ACT->AGC(1), GAA->AAC(1)ACA(1)GAC(1)GAG(7), ...
297	8.8674	3.4665	0.7715	4.4931	ACT->ACG(1), AGC->ACC(2)ACT(1)AGT(4)GCT(1), AGT->ACT(1)
311	47.4661	0.7582	0.9647	0.7860	AAA->AAG(1), AAG->AAA(1)AGT(1)
281	4.3164	39.0412	60.0429	0.6502	ACC->ACA(1)AGC(1)GCC(1), TCA->TCG(1), TCC->TTC(1), TCT->A...

```
yokoyama.obj <- "|   Site   |       ER (2 vs 1)       |       ER (3 vs 2)       |   ER (3-island vs 2)
|    13    |          52.2073        |           0.5929        |           0.8707        |           0.
|    14    |         150.9647        |           1.8552        |           0.7652        |           2.
|    19    |          27.9752        |           0.6240        |           1.0113        |           0.
|    26    |          37.3421        |           0.7866        |           0.9703        |           0.
|    35    |           8.0963        |           1.0832        |           0.9637        |           1.
|    45    |          26.7908        |           0.7598        |           0.9401        |           0.
|    46    |          14.4686        |           2.1330        |           0.9158        |           2.
|    50    |          18.9419        |           1.0181        |           1.0734        |           0.
|    84    |          46.5318        |           0.5237        |           0.8465        |           0.
|    93    |          52.8115        |           2.9123        |           0.9325        |           3.
|    96    |          37.1341        |           3.2911        |           0.9412        |           3.
|   107    |           9.7138        |           1.2380        |           0.9826        |           1.
|   108    |         101.0508        |           1.8546        |           0.8973        |           2.
|   122    |          17.8947        |           1.9526        |           0.9081        |           2.
|   136    |          71.0212        |           0.6404        |           0.9599        |           0.
|   144    |         318.8699        |           0.7403        |           1.1404        |           0.
|   149    |          22.8614        |           2.3053        |           0.9619        |           2.
|   158    |          14.1189        |           3.0228        |           0.9465        |           3.
|   160    |          28.9554        |           1.2120        |           1.3124        |           0.
|   162    |          42.3805        |           0.5871        |           1.1305        |           0.
|   163    |          20.8949        |           0.8189        |           0.9566        |           0.
|   165    |          10.1073        |           1.5528        |           0.9650        |           1.
|   195    |          41.9450        |           1.6128        |           0.9415        |           1.
|   196    |           9.5755        |           1.9173        |           0.9614        |           1.
|   209    |           9.2534        |           1.0200        |           1.0889        |           0.
|   210    |          78.1724        |           3.0818        |           0.9327        |           3.
|   213    |          22.7746        |           0.5560        |           0.9739        |           0.
|   236    |          45.2105        |           5.0861        |           0.9490        |           5.
|   264    |          32.4793        |           1.1110        |           0.9455        |           1.
|   270    |         216.0457        |          13.2258        |          11.3947        |           1.
|   271    |          51.2877        |           1.6525        |           0.9532        |           1.
|   273    |           9.7226        |           0.9912        |           0.9105        |           1.
|   274    |          95.5793        |           0.5156        |           0.9282        |           0.
|   277    |         121.9976        |           1.9885        |           0.8298        |           2.
|   282    |          94.8591        |           1.3634        |           0.9965        |           1.
|   297    |           8.8674        |           3.4665        |           0.7715        |           4.
|   311    |          47.4661        |           0.7582        |           0.9647        |           0.
|   281    |           4.3164        |          39.0412        |          60.0429        |           0.

yoko_ER_DF <- read_delim(yokoyama.obj, delim = '|', trim_ws = TRUE)
```

```
## Warning: Missing column names filled in: 'X1' [1], 'X8' [8]
```

```
yoko_ER_DF %>%  melt(id.vars = "Site", measure.vars = c("ER (2 vs 1)","ER (3 vs 2)","ER (3-island vs 2)
  geom_col(aes(x = Site, y = value, fill = variable), position = position_dodge(4)) + coord_cartesian(y
```




```
  #geom_bar(aes(y = value, fill = variable))
```

```
yoko_ER_DF$`ER (2 vs 1)` %>% max()
```

```
## [1] 318.8699
```

```
yoko_ER_DF$`ER (2 vs 1)` %>% which.max()
```

```
## [1] 16
```

```
which(log(yoko_ER_DF$`ER (2 vs 1)`) > 2) %>% length()
```

```
## [1] 37
```

```
which(log(yoko_ER_DF$`ER (3 vs 2)`) > 2) %>% yoko_ER_DF$Site[.]
```

```
## [1] 270 281
```

```
which(log(yoko_ER_DF$`ER (3 vs 2)`) > 2) %>% yoko_ER_DF$`ER (3 vs 2)`[.]
```

```
## [1] 13.2258 39.0412
```

```
which(log(yoko_ER_DF$`ER (3-island vs 2)`) > 2) %>% yoko_ER_DF$Site[.]
```

```
## [1] 270 281
```

```
which(log(yoko_ER_DF$`ER (3-island vs 2)`) > 2) %>% yoko_ER_DF$`ER (3-island vs 2)`[.]
```

```
## [1] 11.3947 60.0429
```

```
which(log(yoko_ER_DF$`ER (3-island vs 3)`) > 2) %>% yoko_ER_DF$Site[.]
```

```
## numeric(0)
```

```r
which(log(yoko_ER_DF$`ER (3-island vs 3)`) > 2) %>% yoko_ER_DF$`ER (3-island vs 3)`[.]
```

```
## numeric(0)
```