RSDexport

Yichen Han

2024-08-09

Experiment 37

Spec.region excluded for “abnormal”. Functional change not guaranteed.

Call: `GenePermutation(triplets, keyed, num.perm=15000, min.subs=10, max.subs=30, spec.region=30:60)`

```r
checkpoint_path <- file.path("checkpoints")
dir_path <- file.path("outputs")
run_name <- "rsd-permutation_37"
modelloaded <- load_cp(paste0(checkpoint_path, "/", run_name),
                 cp_filter = "acc")
```

```
## Using checkpoint checkpoints/rsd-permutation_37/Ep.008-val_loss0.24-val_acc0.968.hdf5
```

```r
model <- load_cp(paste0(checkpoint_path, "/", run_name),
                 cp_filter = "last_ep")
```

```
## Using checkpoint checkpoints/rsd-permutation_37/Ep.010-val_loss1.02-val_acc0.910.hdf5
```

Model with best ACC

```r
path_special_test <- file.path("special/test")
path_normal_test <- file.path("normal/test")
path_abnormal_test <- file.path("abnormal/test")
eval_model <- evaluate_model(path_input = c(path_normal_test,
  path_abnormal_test, path_special_test),
  model = modelloaded,
  batch_size = 64,
  step = 5,
  vocabulary_label = list(c("normal", "abnormal", "special")),
  number_batches = 10,
  mode = "label_folder",
  verbose = FALSE
)

eval_model
```

```
## [[1]]
## [[1]]$confusion_matrix
##           Truth
## Prediction normal abnormal special
##   normal       95        0       6
##   abnormal     96      256      11
##   special      65        0     239
##
## [[1]]$accuracy
## [1] 0.7682292
##
## [[1]]$categorical_crossentropy_loss
## [1] 3.482314
##
## [[1]]$AUC
## NULL
##
## [[1]]$AUPRC
## NULL
```

```
special_seq <- permute_sequence(triplets, keyed, type="ok", min.subs=5, max.subs=30,
                                 dict=codon.dict, spec.cond=FALSE, spec.region=NULL)
special_seq <- permute_sequence(special_seq, keyed, type="func", min.subs=5, max.subs=30,
                                 dict=codon.dict, spec.cond=TRUE, spec.region=30:60)
special_seq <- paste(special_seq, collapse = "")
onehot_instance <-  seq_encoding_label(char_sequence = special_seq,
                                          maxlen = 400,
                                          start_ind = 1,
                                          vocabulary = c("A", "C", "G", "T"))
head(onehot_instance[1,,])
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    1    0    0    0
## [2,]    0    0    0    1
## [3,]    0    0    1    0
## [4,]    0    1    0    0
## [5,]    0    0    0    1
## [6,]    0    0    0    1
```

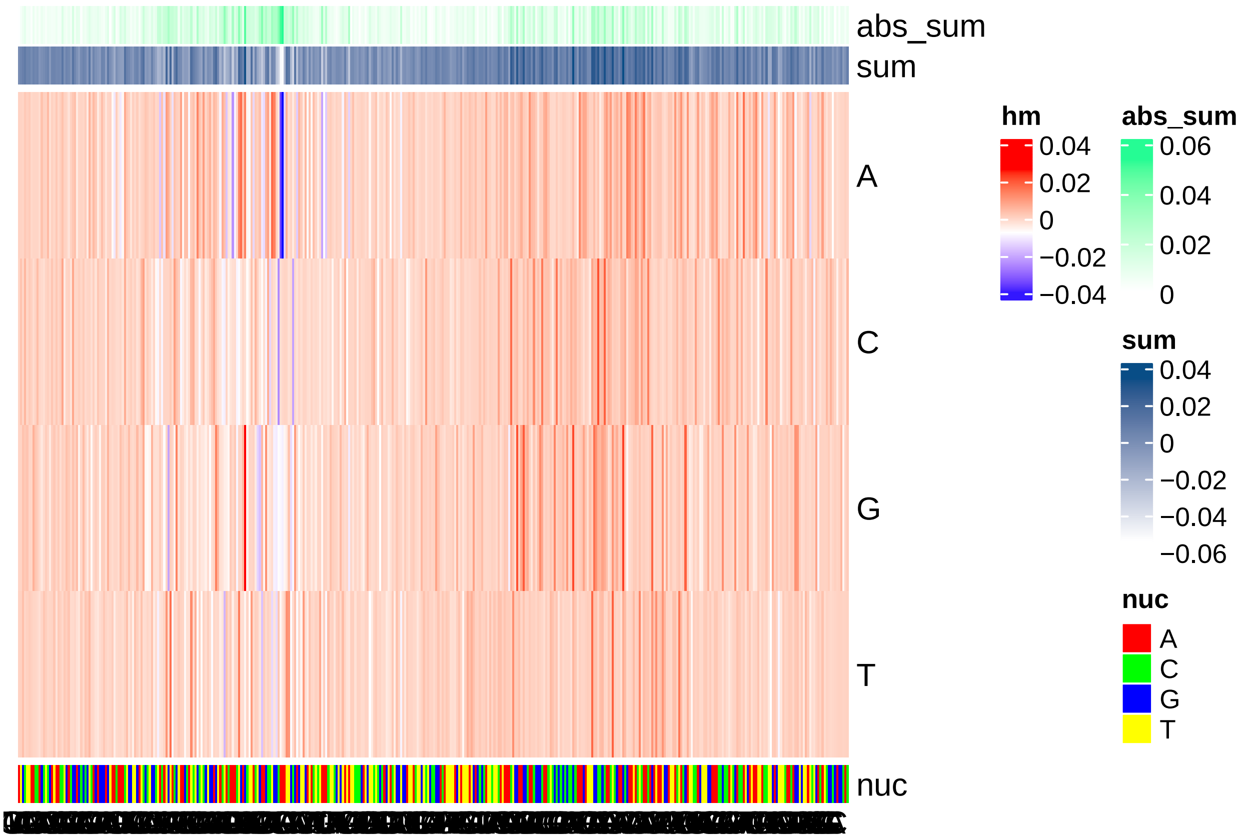
```
pred <- predict(model, onehot_instance, verbose = 0)
pred
```

```
##              [,1]         [,2] [,3]
## [1,] 2.622429e-13 4.472673e-14    1
```

```
ig <- integrated_gradients(
  input_seq = onehot_instance,
  baseline_type = "shuffle",
  target_class_idx = 3,
  model = modelloaded,
  num_baseline_repeats = 10)
```

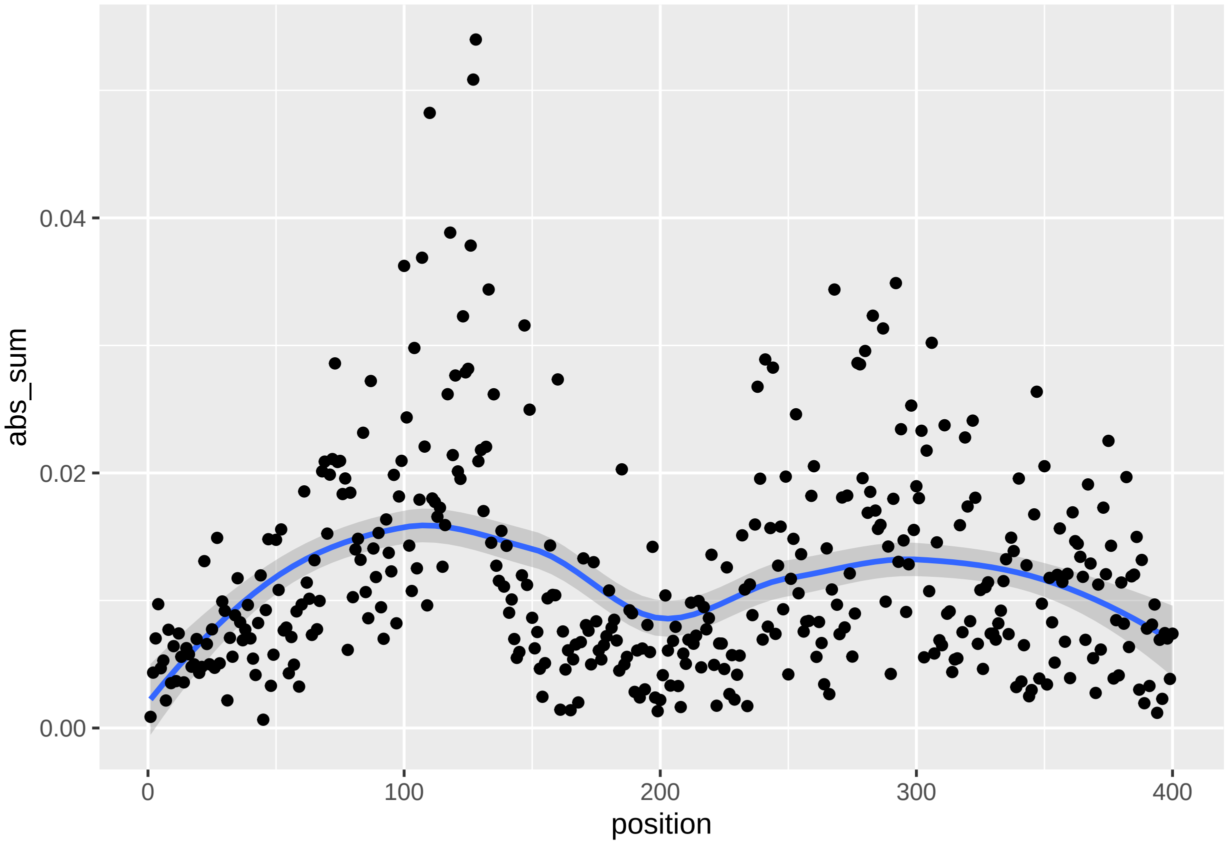
```
heatmaps_integrated_grad(integrated_grads = ig,
                         input_seq = onehot_instance)
```

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



```
abs_sum <- rowSums(abs(as.array(ig)))
df <- data.frame(abs_sum = abs_sum, position = 1 : 400)
ggplot(df, aes(x = position, y = abs_sum)) + geom_smooth() + geom_point()
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



Model at last Epoch

```
path_special_test <- file.path("special/test")
path_normal_test <- file.path("normal/test")
path_abnormal_test <- file.path("abnormal/test")
eval_model <- evaluate_model(path_input = c(path_normal_test,
  path_abnormal_test, path_special_test),
  model = model,
  batch_size = 64,
  step = 5,
  vocabulary_label = list(c("normal", "abnormal", "special")),
  number_batches = 10,
  mode = "label_folder",
  verbose = FALSE
)

eval_model
```

```
## [[1]]
## [[1]]$confusion_matrix
##           Truth
## Prediction normal abnormal special
##   normal       96        8       6
##   abnormal    141      239      12
##   special      19        9     238
##
## [[1]]$accuracy
## [1] 0.7460938
##
## [[1]]$categorical_crossentropy_loss
## [1] 3.715905
##
## [[1]]$AUC
## NULL
##
## [[1]]$AUPRC
## NULL
```

```
special_seq <- permute_sequence(triplets, keyed, type="ok", min.subs=5, max.subs=30,
                                dict=codon.dict, spec.cond=FALSE, spec.region=NULL)
special_seq <- permute_sequence(special_seq, keyed, type="func", min.subs=5, max.subs=30,
                                dict=codon.dict, spec.cond=TRUE, spec.region=30:60)
special_seq <- paste(special_seq, collapse = "")
onehot_instance <-  seq_encoding_label(char_sequence = special_seq,
                                          maxlen = 400,
                                          start_ind = 1,
                                          vocabulary = c("A", "C", "G", "T"))
head(onehot_instance[1,,])
```

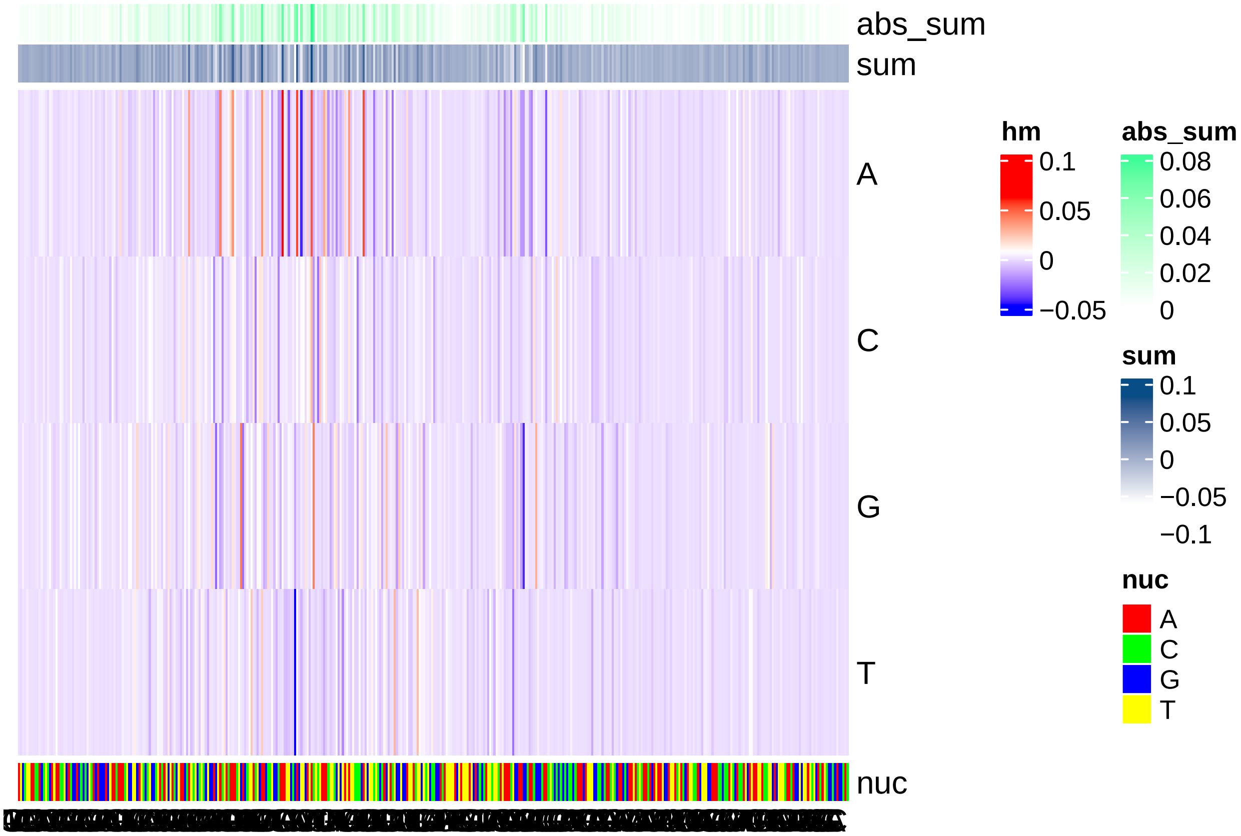
```
##      [,1] [,2] [,3] [,4]
## [1,]    1    0    0    0
## [2,]    0    0    0    1
## [3,]    0    0    1    0
## [4,]    0    1    0    0
## [5,]    0    0    0    1
## [6,]    0    0    0    1
```

```
pred <- predict(model, onehot_instance, verbose = 0)
pred
```

```
##               [,1]         [,2] [,3]
## [1,] 2.622429e-13 4.472673e-14    1
```

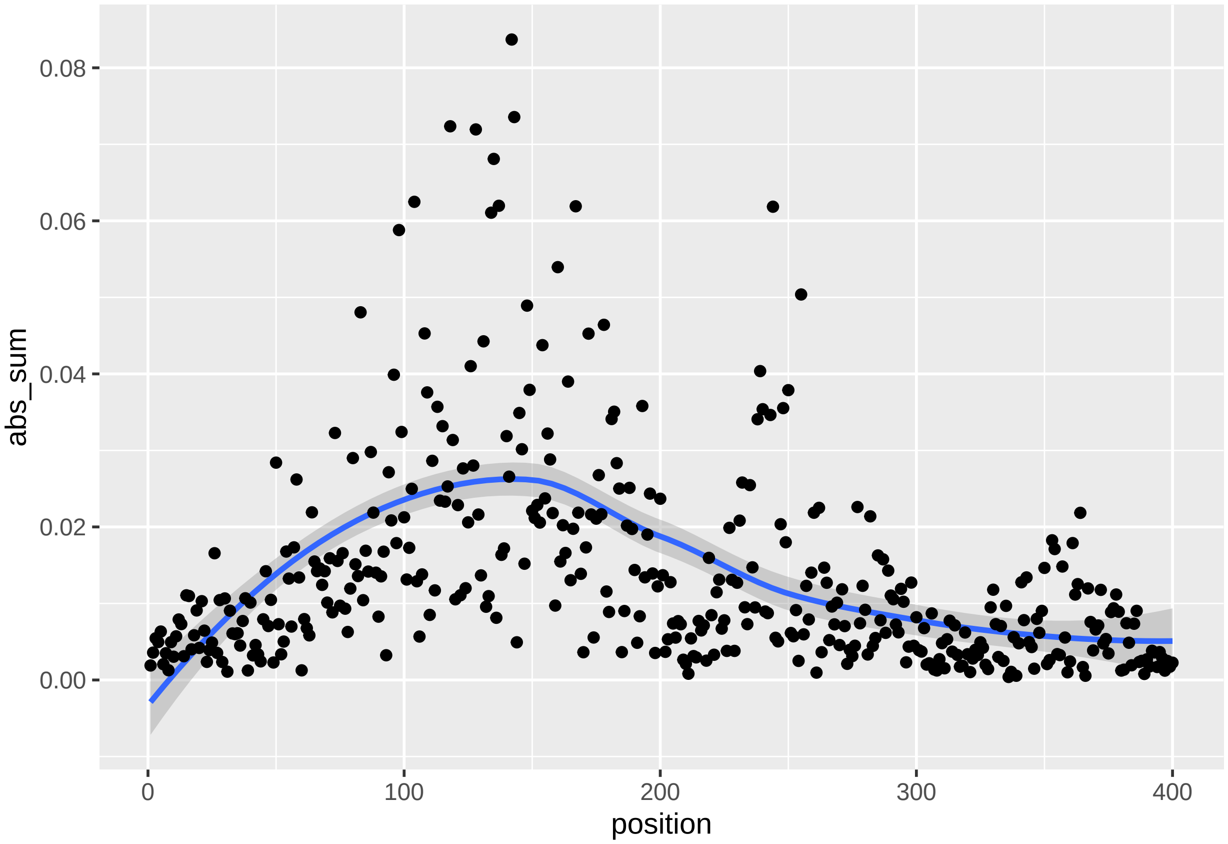
```
ig <- integrated_gradients(
  input_seq = onehot_instance,
  baseline_type = "shuffle",
  target_class_idx = 3,
  model = model,
  num_baseline_repeats = 10)

heatmaps_integrated_grad(integrated_grads = ig,
                         input_seq = onehot_instance)
```

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



```
abs_sum <- rowSums(abs(as.array(ig)))
df <- data.frame(abs_sum = abs_sum, position = 1 : 400)
ggplot(df, aes(x = position, y = abs_sum)) + geom_smooth() + geom_point()
```

```
## ‘geom_smooth()‘ using method = ’loess’ and formula = ’y ~ x’
```


0.08
0.06
0.04
0.02
0.00
abs_sum
0
100
200
300
400
position