gliph.R

exec gliph pipeline

return clusters & edges

abstract exec flow

get_control() util_v1_v2.R	check control input refactor into input_checks
input_check() input_check.R	check input parameters
get_chains() util_v1_v2.R	get cdr3a/b columns
add id	index tcr sample
trim flanks	cut off aa start/end
create edges	prepare unique (v1, v2) or all edges (v3) for clustering input
get_chain_run_v1/2() util_v1.R (v1) ‖ util_v2.R (v2+v3)	local & global clustering
get_edges() util_v1_v2.R	edges between seqs
return	clusters, edges, indexed data_sample & parameters

input

data_sample	tcr sample
data_ref	reference database
version = 2	gliph version
ks = c(2, 3, 4)	motif lengths
cores = 1	cpu cores
(B = 1000)	simulation depth
(global_max_dist = 1)	max hamming distance
(local_min_fdr = 0.05)	max random prob cutoff
(local_min_ove = 2)	min fold enrichment
(local_min_o = 3)	min motif observations
(trim_flanks = FALSE)	cut off aa (replace? flank_size = 0)
(flank_size = 3)	aa left/right cut off i.e. trim_flank_size = 0
(global_pairs = NULL)	precomputed global pairs
(low_mem = FALSE)	slow looping with lower memory footprint

return

clust	local + global clusters
edges	local + global edges
data_sample	indexed input data
control	input parameters

util_v1.R

exec gliph1 pipeline

return local&global pairs

abstract exec flow

get_motifs_v1()	get local motifs
get_motif_enrichment_v1()	get local enrichment
get_motif_filter_v1()	filter by p, fold & observation cutoff
get_local_pair() util_v1_v2.R	find local motif pairs
get_global_pairs/mem() util_v1_v2.R	get global pairs
return	local & global pairs, motif enrichment

input

cdr3	cdr3 sample
cdr3_ref	cdr3 reference database
ks = c(2, 3, 4)	motif lengths
cores = 1	cpu cores
(B = 1000)	simulation depth
(global_max_dist = 1)	max hamming distance
(local_min_fdr = 0.05)	max random prob cutoff
(local_min_ove = 2)	min fold enrichment
(local_min_o = 3)	min motif observations
(trim_flanks = FALSE)	cut off aa (replace? flank_size = 0)
(flank_size = 3)	aa left/right cut off
(global_pairs = NULL)	precomputed global pairs (?)
(low_mem = FALSE)	slow looping with lower memory footpring

return

local_pairs	local cdr3 pairs
global_pairs	global cdr3 pairs
motif_enrichment	indexed input data

util_v2.R

exec gliph2+3 pipeline

return local&global pairs

abstract exec flow

get_motifs_v2()	get local motifs
get_motif_enrichment_fet_v2()	get local enrichment with fisher's exact test
get_motif_filter_v2()	filter by p, fold & observation cutoff
get_local_pair() util_v1_v2.R	find local motif pairs
get_global_pairs/mem() util_v1_v2.R	get global pairs
return	local & global pairs, motif enrichment

input

cdr3	cdr3 sample
cdr3_ref	cdr3 reference database
ks = c(2, 3, 4)	motif lengths
cores = 1	cpu cores
(B = 1000)	simulation depth
(global_max_dist = 1)	max hamming distance
(local_min_fdr = 0.05)	max random prob cutoff
(local_min_ove = 2)	min fold enrichment
(local_min_o = 3)	min motif observations
(trim_flanks = FALSE)	cut off aa (replace? flank_size = 0)
(flank_size = 3)	aa left/right cut off
(global_pairs = NULL)	precomputed global pairs (?)
(low_mem = FALSE)	slow looping with lower memory footpring

return

local_pairs	local cdr3 pairs
global_pairs	global cdr3 pairs
motif_enrichment	indexed input data

util_v1_v2.R

gliph helper functions

different input & return

get_control()

in	control
filter	replace by default sort
out	cleaned control

get_local_pair()

in	cdr3 seq, motifs
filter	find enriched motifs
out	local pairs

get_global_pairs/mem()

in	cdr3 seq, global_max_dist
filter	look for global connections
out	global pairs

get_chains()

in	data_sample columns
filter	look for cdr3a/b column
out	cdr3a/b columns

get_trimmed_flanks()

in	cdr3 seq, flank size
filter	trim cdr3 seqs
out	trimmed cdr3 seqs

get_edges()

in	local_pairs, global_pairs, cdr3 seq, chains
filter	find local&global edges
out	local & global edges