def histogram\_horizontal(ph, num\_bins=100, min\_bin=None, max\_bin=None):

ph1 = [p[:2] for p in ph] # simplify to ensure ph corresponds to 2d barcode

if min\_bin is None:

min\_bin = np.min(ph1)

if max\_bin is None:

max\_bin = np.max(ph1)

bins = np.linspace(min\_bin, max\_bin, num\_bins, dtype=float)

binsize = (max\_bin - min\_bin) / (num\_bins - 1.)

results = np.zeros(num\_bins - 1)

for p in ph1:

ph\_decompose = np.linspace(np.min(p), np.max(p),

math.ceil((np.max(p) - np.min(p)) / binsize),

dtype=float)

bin\_ph = np.histogram(ph\_decompose, bins=bins)[0]

results = np.add(results, bin\_ph)

return bins, results

* calculates how many barcodes are there in each bin of the cell

def histogram\_stepped(ph1):

bins = np.unique(list(chain(\*ph1)))

results = np.zeros(len(bins) - 1)

for ph in ph1:

for it, \_ in enumerate(bins[:-1]):

if min(ph) <= bins[it + 1] and max(ph) > bins[it]:

results[it] = results[it] + 1

return bins, results

* calculates step distance of each cell i.e. number of unique buckets and the number of barcodes in them

Therefore, difference comes in the number of bins.