**A.*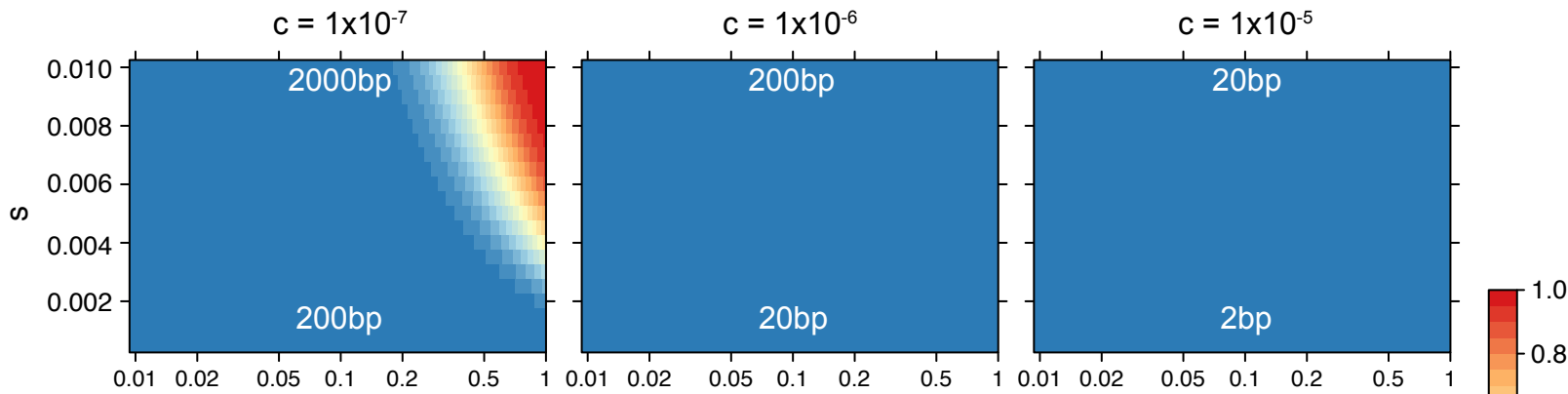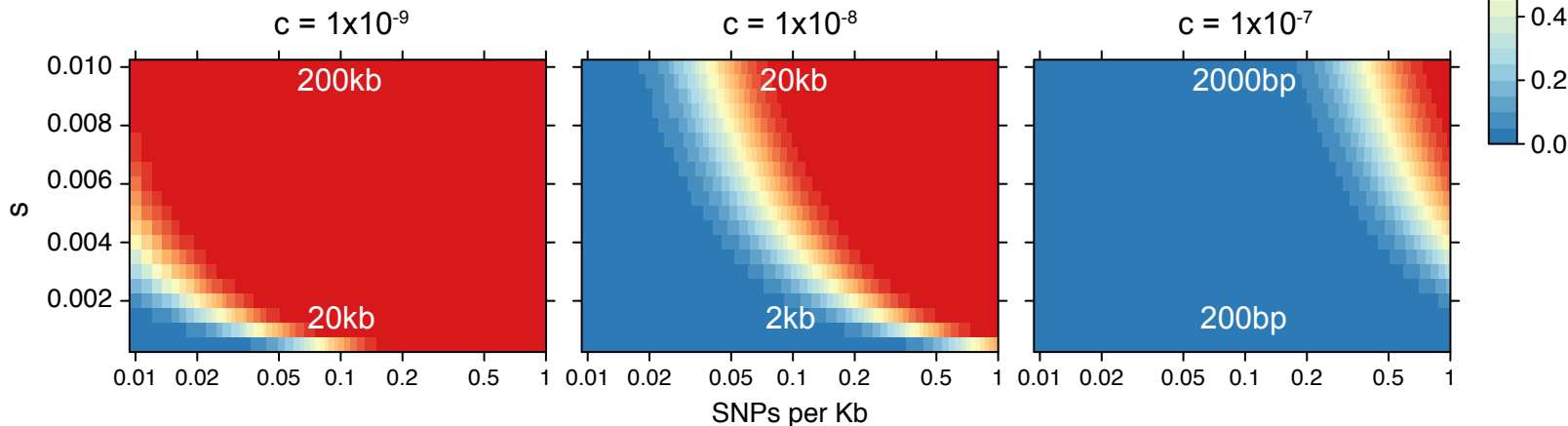* Probability of finding 50% of 10 sweeps under different parameter values. Recreated Figure 1B, Tiffin and Ross-Ibarra, 2014. Based on the values of *s* and *c*, we have overlayed the size of the simulated sweeps.

B. Corrected recombination rate: Probability of finding 50% of 10 sweeps under different parameter values.

SNPs per Kb

C. Original, public R Code to simulate the sweeps.

```
# https://rpubs.com/rossibarra/21584

c=1E-7 # recombination rate per bp
s=0.01 # selection coefficient
G=2.1E9 # genome size
snps=1E5 # number of SNPs
fragment=60 # each SNP represents a 60 bp fragment


# Calculations

# Basic naive sweep model (Hudson and Kaplan) gives length of
# sweep \(L=\frac{0.02s}{c}\)

L=0.02*s/c

# and our data represent \(x=\frac{snps\,\times\,fragment}{G}\)
# proportion of the genome

x=snps*fragment/G

# If we represent the nonindependence of bp by dividing
# sweepsize by fragment length, our probability of missing
# a sweep is \((1-x)^\frac{L}{fragment}\):

p=(1-x)^(L/fragment)
```

D. Modified R Code to simulate the sweeps.

```
calculate_sweeps <- function(G, fragment, c, snps, selv, sweeps) {
  K = G/1000     # genome size in kilobases

  # Calculate the SNPs per kilobase
  snps_per_kb = snps / K

  # Fraction of the genome sequenced
  x = snps*fragment/G

  d1 = expand.grid(x, selv)
  colnames(d1)=c("x", "sel")

  # Compute the lengths of sweeps, for particular values of
  # selection (sel) and recombination (c).
  L = 0.02 * d1$sel / c

  # Probability we miss a sweep.
  p_miss = (1-x)^((L)/fragment)

  # Probability we hit a sweep.
  p_hit  = 1 - p_miss

  # Probability we find 50% of 10 sweeps
  p50 = 1 - pbinom(sweeps * 0.5, sweeps, p_hit)

  d2 = expand.grid(snps, selv)
  colnames(d2)=c("snps", "sel")
  d3 = expand.grid(snps_per_kb, selv)
  colnames(d3)=c("snps_per_kb", "sel")

  sweep_df = data.frame(L, p_hit, p50, d2$sel, d2$snps, d3$snps_per_kb)
  colnames(sweep_df) = c("L", "p", "p50", "sel", "snps", "snps_per_kb")

  return(sweep_df)
}

# vector of number of SNPs
snps = seq(from = 21000, to = 2100000, by = 3000)

# vector of selection coefficients
selv = seq(from = 0.0005, to = 0.010, by = 0.0005)

G = 2.1E9      # genome size
fragment = 60 # each SNP represents a 60 bp fragment
sweeps   = 10

c = 1E-5       # recombination rate per bp, 1E-5 ... 1E-9
sweep_df = calculate_sweeps(G, fragment, c, snps, selv, sweeps)
```