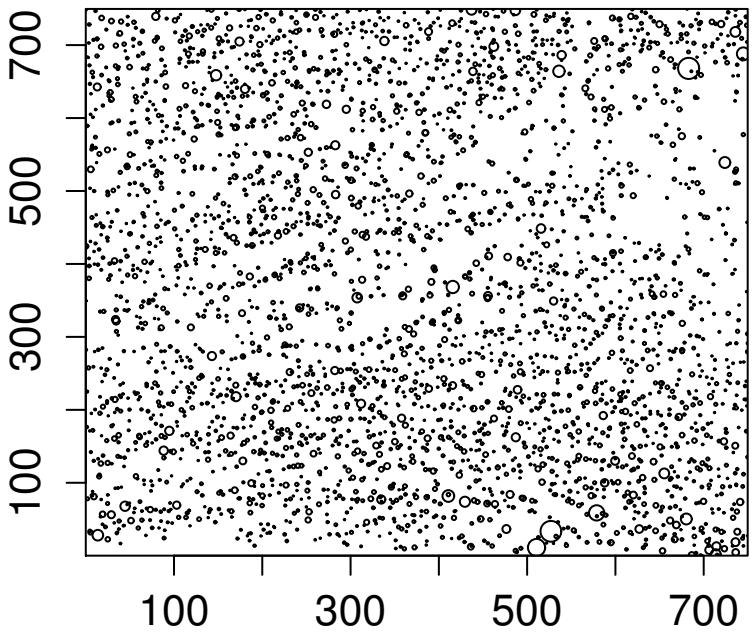

Figure 1: A 750-by-750 foot stand of 3396 trees. The area of each circle is proportional to the total basal area of the tree, which is the cross-sectional area of the trunk at about 2 meters from the ground.

This computing project centers on different methods used to estimate the total of some attribute of a population that is distributed across a spatial region. To make this concrete, shown in Figure 1 is a small forest that contains a total of 3396 trees. The area of each circle is proportional to the total basal area of the tree, which is the cross-sectional area of the trunk 1.6 meters from the ground.

There are two objectives for this project. The first is to compare several different methods for estimating the total basel area of the 3396 trees in this forest, in order to compare their relative accuracy. For each method, you will write code to randomly select a sample and then calculate an estimate for the total basal area of all trees in the forest based on the sample. Each sampling method will be carried out 10^5 times, and then we compute the standard deviation of the 10^5 estimates for the totals. A lower standard deviation indicates a more accurate estimation method so is considered desirable.

The second objective is to write code that runs *fast*. For each part, include code to time how long it takes for your code to run.

You will be working from the data set `trees.csv` that has 3396 records, each with the following entries:

```
spp = species code
dbh = diameter at breast height, the trunk diameter 1.6 meters from the ground
ba = basal area of tree
volume = volume of wood
x = x-coordinate of tree location
y = y-coordinate of tree location
```

Part 1: Masuyama's Method[1]

For this method, we will select a random circular plot of radius $r > 0$. The center for the plot is a randomly selected point that is either in the stand or within a distance of r from the stand. For the purpose of this assignment, a random center (x, y) is selected uniformly from the region $[-r, 750 + r] \times [-r, 750 + r]$. (See Figure 2 for examples of plots.)

Once a random plot is selected, then all trees within the plot are selected for the sample, which we denote by S. Now let

$$a = \pi r^2, \qquad A^* = (750 + 2r)^2$$

The probability that a tree z_i is selected into a random sample is given by

$$\pi_i = \frac{a}{A^*}$$

for every tree in the stand. We use the following formula to estimate the total basal area (TBA) for the entire stand:

$$\hat{t} = \sum_S \frac{y_i}{\pi_i} = \frac{A^*}{a} \sum_S y_i$$

where the sum runs over all trees in the sample S, and y_i is the basal area for tree z_i. For each tree, y_i is given in the data set by `ba`.

Simulation We will use simulation to determine the accuracy of each estimation method. Your code should allow the user to define variables that set the plot radius r and the minimum and maximum x and y coordinates for the stand.

Set $r = 37$ (this gives a plot size of about 400 m^2). Generate 10^5 random plots, and compute an estimate $\hat{t}$ from each plot. Compute two things:

Percentage Bias This is given by

$$100 \frac{\text{average}(\hat{t}) - t}{t}$$

For this project, $t = 311.906$ for the entire stand. The percentage bias gives a measure of how different the estimated values are (on average) from the true total, as a percentage of the total. In theory this is zero, but due to randomization in the simulations this value is typically between -0.5% and 0.5%. (If it is not, then your code may have errors.)

Percentage Root Mean Square Error This is given by

$$100 \frac{\sqrt{\text{Var}(\hat{t})}}{t}$$

The root mean square error (RMSE) gives a measure of the variation in $\hat{t}$ as a percentage of the total. It is desirable to have $\hat{t}$ near t, so a smaller RMSE is better. We will use the RMSE to compare methods.

Your output should include three labeled things: (1) the percentage bias; (2) the percentage RMSE; (3) the elapsed time to run your code, using the `proc.time` function to compute the elapsed time. See the .R file for an example.

[1] Masuyama is the name of the statistician who proposed this method.

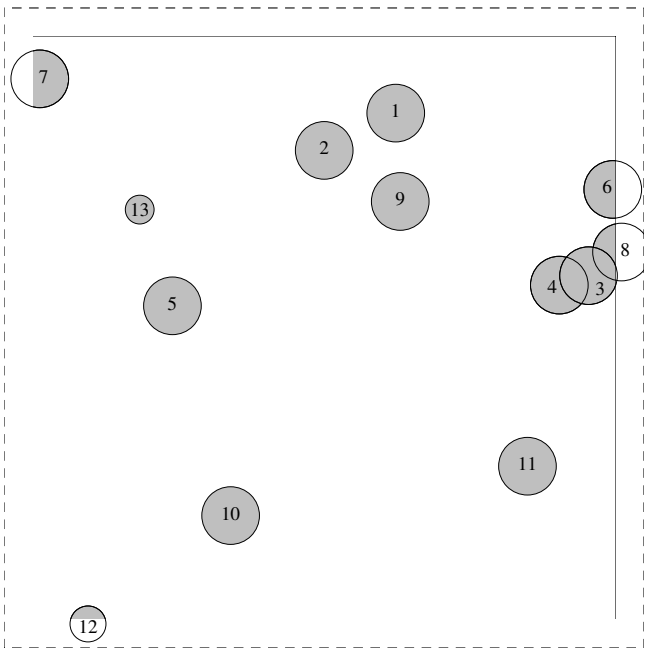

Figure 2: Example plots are numbered 1–11. (Ignore 12 and 13, which are smaller.) Note that some plots overlap the stand edge (solid line), and plot 8 is centered outside the stand. The dashed lines indicated the extended region of radius r where plot centers can be located.

Part 2: Measure π_i Method

This method is similar to Masuyama's, with two important differences:

1. Plot centers must be within the stand, not extending over the edge as with Masuyama.
2. π_i is not the same for all trees. Now we have

$$\pi_i = \frac{a(z_i)}{A}$$

where A equals the stand area (750^2 for this project), and $a(z_i)$ is the area of the portion of the plot of radius r centered at z_i (the location of tree i) that overlaps the stand. In Figure 2, if the circles were centered on tree locations, then $a(z_i)$ would be the area of the shaded regions. Trees at the edge of the stand have a smaller π_i than the rest.

The estimator for the Measure π_i method is the same as for Masuyama,

$$\hat{t} = \sum_S \frac{y_i}{\pi_i}$$

Carry out a simulation similar to Part 1: $r = 37$ with 10^5 simulated samples. Compute the Percentage Bias and the Percentage RMSE from your simulated estimates for the population TBA. As with Part 1, the Percentage Bias should be small, less than 1%.

Your output should include three labeled things: (1) the percentage bias; (2) the percentage RMSE; (3) the elapsed time to run your code, using the `proc.time` function to compute the elapsed time.

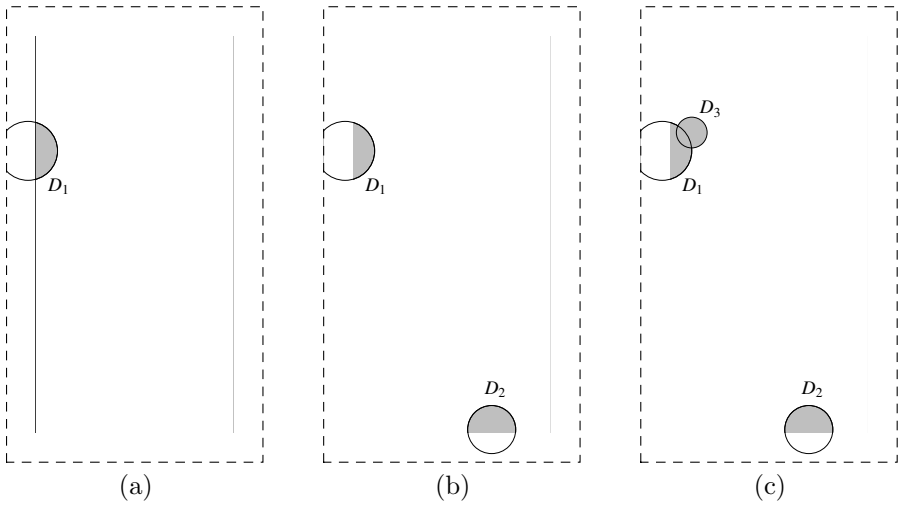

Figure 3: An example of an repeated Masuyama sample. For parts (a)–(c), the stand $\mathcal{A}$ is the region enclosed by the solid rectangle and $\mathcal{A}^*$ is the region enclosed by the dashed rectangle. (a) The first sample plot is D_1, which has total area equal to $a = \pi r^2$. (b) The second sample plot is D_2, which has total area equal to the portion of D_1 outside $\mathcal{A}$. (c) The third sample plot is D_3, which has total area equal to the portion of D_2 outside of $\mathcal{A}$. Since D_3 is entirely within $\mathcal{A}$, it is the last plot in the sample. The sample consists of the population elements contained in the shaded regions. The total area of the portions of the plots inside $\mathcal{A}$ is a. For this sample, trees in the intersection of D_1 and D_3 will be counted twice in the estimate.

Part 3: Repeated Masuyama

This method is similar to Masuyama in Part 1, but if a sample plot overlaps the stand boundary, then we generate another smaller sample plot. The second plot has radius selected so that its area is equal to the amount of stand overlap of the first plot. This approach is repeated until a plot is obtained that does not overlap the boundary. The selection method is such that the combined stand area within the plots is equal to a. All trees in the combined plots are included in the sample, with trees included in several plots counted once for each plot they are in. Figure 3 shows an example of plot selection.

The estimator for repeated Masuyama is

$$\hat{t} = \frac{A}{a} \sum_{S^*} y_i$$

where S^* indicates all trees in the sample plots, with trees in multiple plots counted once per plot.

As with the previous parts, use $r = 37$ and 10^5 simulations. Compute the Percentage Bias and the Percentage RMSE. Your output should include three labeled things: (1) the percentage bias; (2) the percentage RMSE; (3) the elapsed time to run your code, using the `proc.time` function to compute the elapsed time.