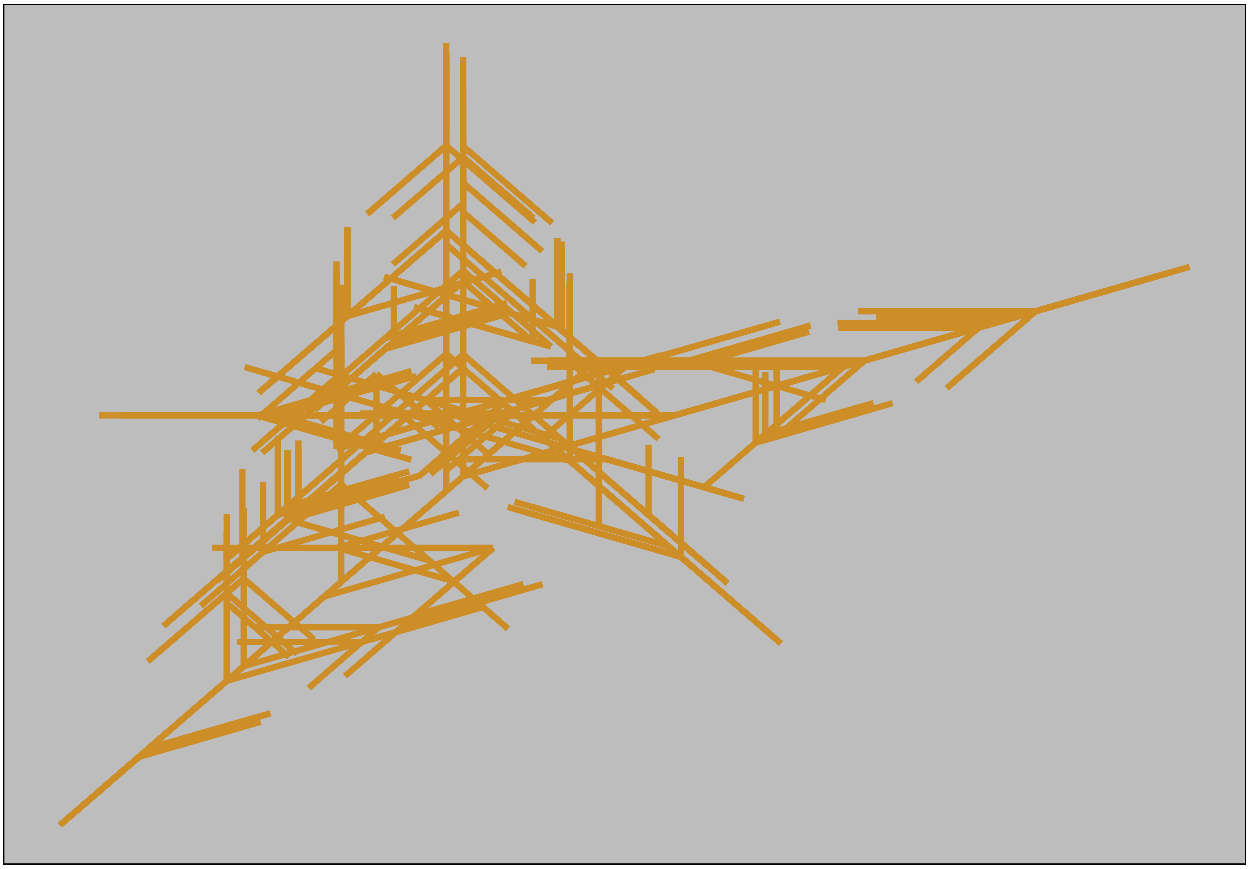
Project Report (vignette)

```r
#devtools::install()
library(tibble)
library(leafart)
# To actually show the knitted vignette document, I had to use devtools::build_vignettes() and then rem
# also using pdf_document instead of html_vignette
```

Complete example

```r
get_params(nleaves = 1,
           split = 3,
           angle = c(-150,0,0,150),
           scale = c(.8,.9),
           n_layer = 5) %>%
  create_leaf_pile() %>%
  plot_leaves()
```



Growing a leaf:

A leaf starts with a stalk. We keep track of the stalk start and endpoint, and then draw a line between them.

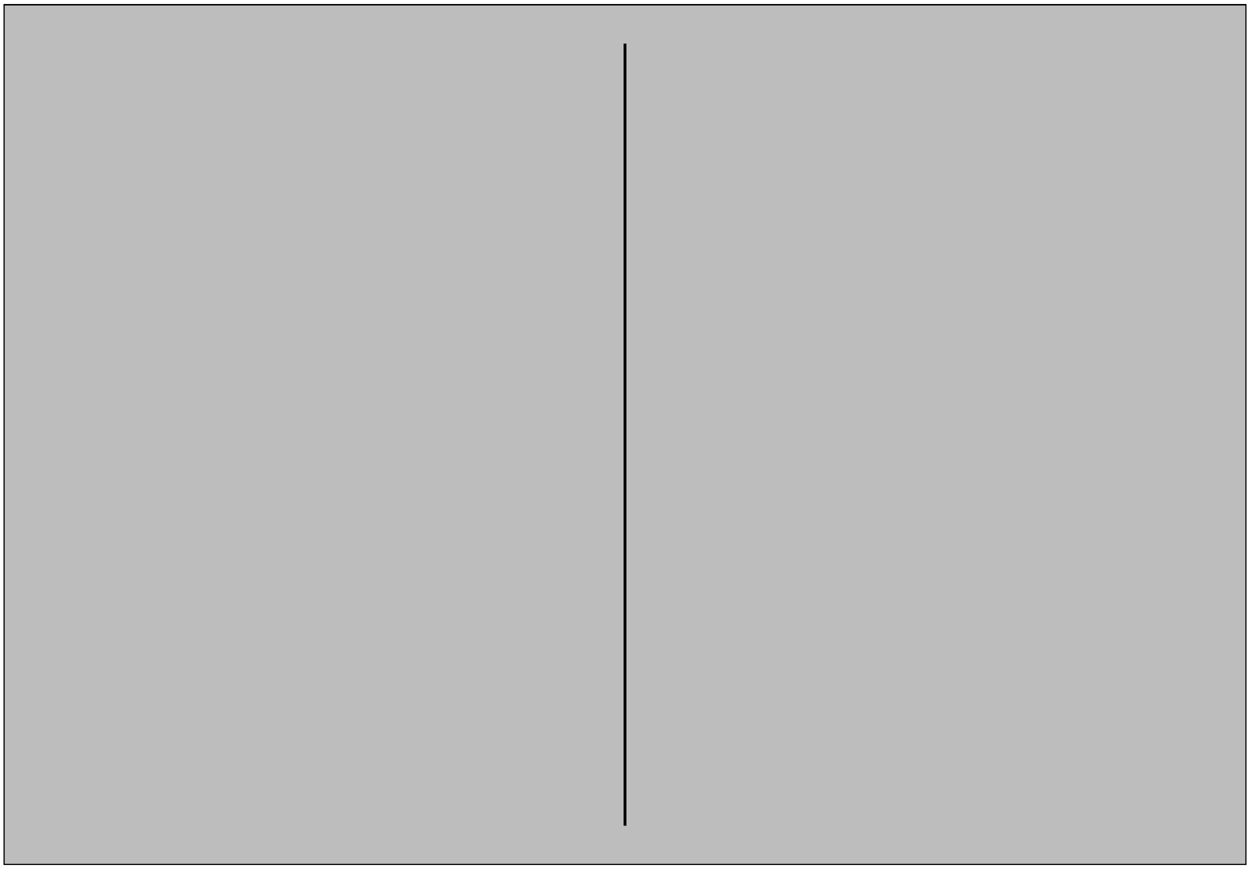
Here's a stalk:

```
(initial_leaf <- tibble::tibble(
    x_0 = 0,
    y_0 = 0,
    x_1 = 0,
    y_1 = 1
  ))
#> # A tibble: 1 x 4
#>     x_0   y_0   x_1   y_1
#>   <dbl> <dbl> <dbl> <dbl>
#> 1     0     0     0     1
```

This format makes it harder to plot. The function `rake_leaves()` will transform the leaf into something better suited to plotting. The `plot_leaves()` function will plot a leaf after it is "raked"

```
(rake_leaves(initial_leaf))
#> # A tibble: 2 x 3
#>       x     y  step
#>   <dbl> <dbl> <int>
#> 1     0     0     1
#> 2     0     1     1

rake_leaves(initial_leaf) %>%
  plot_leaves()
```



We consider this initial stalk to be the first "layer"

Now lets create some branches. The way the leaf/tree branches depends on the parameter values we pass to the growing function. These parameters could be single values, or a vector of values to be sampled from.

We need to know the number of branches to branch off from, the angle to branch, and how long the new branches should be.

The function `get_params()` will give us a list of parameter values. We can specify what values we want, or we can use its defaults. It will default to making one tree. Since we just want to make a tree with 1 additional layer now, we will also specify that.

```r
params <- get_params(n_layer = 1,
                     scale = .5,
                     angle = 90,
                     split = 2)
```

To grow one branch, we use the function `one_branch()`. This function creates a new list that contains the growth of the new branch. Note the initial stalk also keeps track of some initial angle values.

```r
initial_leaf <- tibble::tibble(
    x_0 = 0,
    y_0 = 0,
    x_1 = x_0,
    y_1 = y_0 + 1,
    angle = 90,
    length = 1L,
    iter_n = 1L
  )
```

```
one_branch(initial_leaf, params)
#> # A tibble: 1 x 7
#>     x_0   y_0    x_1   y_1 angle length iter_n
#>   <dbl> <dbl>  <dbl> <dbl> <dbl>  <dbl>  <int>
#> 1     0     1 -0.492  1.09   170    0.5      2
```

To grow all the branches in this second layer we use the function `grow_leaf_layers()`. This combines each branch (row) into one data frame that contains one row per new branch.

```
grow_leaf_layers(initial_leaf,params)
#> # A tibble: 2 x 7
#>     x_0   y_0    x_1   y_1 angle length iter_n
#>   <dbl> <dbl>  <dbl> <dbl> <dbl>  <dbl>  <int>
#> 1     0     1 -0.483  1.13   165    0.5      2
#> 2     0     1 -0.366  1.34   137    0.5      2
```

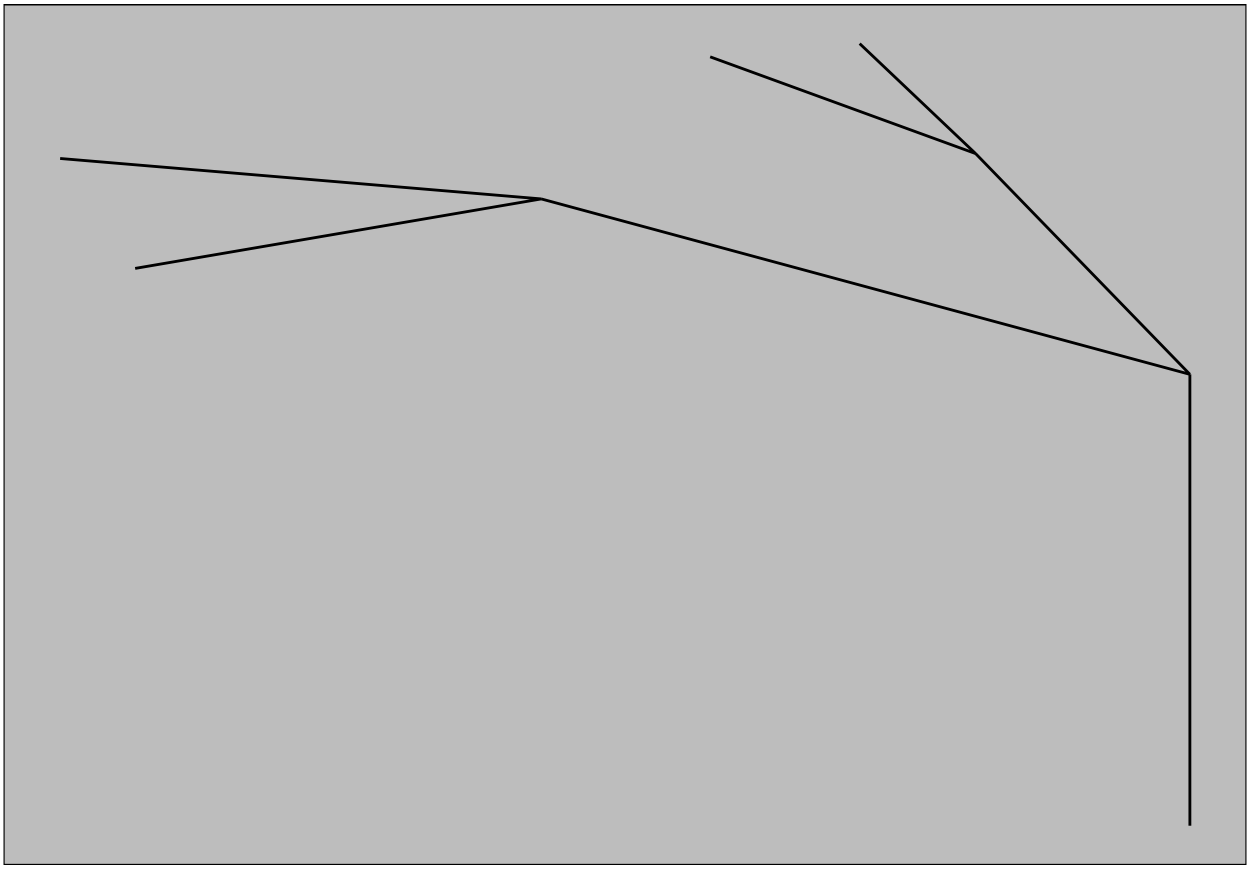
The function `grow_leaf()` will do this entire process, creating the initial stalk and growing the new layer (depending on the parameters)

```
small_leaf <- get_params(n_layer = 1,
                         scale = .5,
                         angle = 90,
                         split = 2) %>%
  grow_leaf()
```

And again, we have to clean up the data frame to be easier to plot. We now also have to keep track of which layer the branch belongs to so the graphing knows which points to connect.

```
get_params(n_layer = 1,
                         scale = .5,
                         angle = 90,
                         split = 2) %>%
  grow_leaf() %>%
  rake_leaves()
#> # A tibble: 6 x 3
#>       x     y  step
#>   <dbl> <dbl> <int>
#> 1  29    33       1
#> 2  29    34       1
#> 3  29    34       2
#> 4  28.6  34.2     2
#> 5  29    34       3
#> 6  28.6  34.2     3


get_params(nleaves = 1,
           n_layer = 2,
           scale = .5,
           angle = 90,
           split = 2) %>%
  grow_leaf() %>%
  rake_leaves() %>%
  plot_leaves()
```



Now we have two layers. We repeat this growing process at the endpoint of each newly grown branch, untill we have fully grown out the tree.

Attempts at speeding up runtime