My Final Project

Deap Singh Bhandal

February 29 2020

Abstract

This project was inspired by the research from Cavender-Bares et al. and that team's reasearch on the live oak clade *Virentes*. I plan on expanding this section by introducing the results more.

Contents

List of Figures

1 Introduction

The study by Cavender-Bares et al. looked at several species of the American Live Oak clade which span North America and Mesoamerica. The researchers measured several variables from several oak trees in specific sites ranging from the genetic makeup of a specific tree to the climate the tree lives in[1]. While the researchers focused more on the genetic makeup differences between the tree species, I was more interested in the climate data. For one, there was an immense amount of data for each tree and I wanted to compare all the climate data variables to location, site, as well as characteristics of the tree such as foliar area. I was motivated to dig deeper into this since the researchers did not touch too much on the relationship between all the climate data and characteristics of the tree. I also recently took a class on plants and learned that depending on the climate, a tree's leaf area index can range from under 1 to over 12[2]. The leaf area index is also known as foliar area which the researchers measured.

2 Materials & Methods

I mainly used five functions in python to open, extract, manipulate, and close my data file. Each function is under it's own subsection. The outputs of the functions can be found in the Results section.

2.1 Finding all the Locations

```
# setting up a function that will print only the unique names from the Location list
def unique(file, num=1):
    # opening data file
    data_file = open(file)
    # importing pandas
    import pandas as pd
    # assigning tmp_data variable to pandas file
    tmp_data = pd.read_csv(data_file)
    # assigning data variable to numpy file
    data = tmp_data.to_numpy()
    # Creating a list called cols that contains all the data from the col
    cols = (data[:, num]).tolist()
    # new empty list
    unique_list = []
    # for any city in col
    for ii in cols:
        # it will be appended to list if it's not in unique list
        if ii not in unique_list:
            unique_list.append(ii)
    # closing the file
    data_file.close()
    # returns the list
    return unique_list

print(unique('Dataset.csv', 1)) # seeing the unique locations
```

2.2 Counting all the Instances of a Location and Species Pair

```
# setting a function that will count occurances of two values in two columns
```

```
def count_pair(file, numb1=1, numb2=2):
    # opening data file
    data_file = open(file)
    # importing pandas
    import pandas as pd
    # assigning tmp_data variable to pandas file
    tmp_data = pd.read_csv(data_file)
    # assigning data variable to numpy file
    data = tmp_data.to_numpy()
    # creating a new list of lists called bicols that has data from 2 cols
    bicols = (data[:, [numb1,numb2]]).tolist()
    # converting list of lists to list of tuples so it is easier for the program to count.
    bicols = [tuple(i) for i in bicols]
    check = False
    # need new empty list
    new_list = []
    ii = 0
    # closing the file
    data_file.close()
    # for loop checking if entry in new_list exists.
    for x in bicols:
        if x in new_list:
            check = True
            continue
        # if entry is not there it will append it,
# if there, it will not re-append it but will increase the count of ii
        else:
            ii = 0
            #
            for y in bicols:
                if y[0] == x[0] and y[1] == x[1]:
                    ii = ii + 1
            # printing the number of occurences only for entries with multiple occurences
            if(ii > 1):
                print(x, "-", ii)
            new_list.append(x)
    if check == False:
        # let's me know if an entry does not repeat
        print("No repeats")

# ran program successfully on Location_Species
count_pair('Dataset.csv',1,2)
```

2.3 Plotting Foliar Area Verses all Climate Data

```
def mass_plotting(file, xdata=1, ystart=2, yend=3):
    # opening data file
    data_file = open(file)
    # importing pandas
    import pandas as pd
    # assigning tmp_data variable to pandas file
```

```
    tmp_data = pd.read_csv(data_file)
    # assigning data variable to numpy file
    data = tmp_data.to_numpy()
    # importing matplotlib.pyplot as plt since I need it later
    import matplotlib.pyplot as plt
    # closing the file
    data_file.close()
    # creating a list of all the column names located in header
    header_list = list(tmp_data)
    # since there are several columns describing the climate an individual tree is in,
    # and I want to compare foliar area to all of them, I made a function to
    # plot Foliar Area vs. all of the climate columns
    # for loop searches through columns 19-57 which are the climate ones
    for column in range(ystart, yend):
        column_list = (data[:, [xdata,column]]).tolist()
        # made a list of tuples in [(x,y)] format
        column_list = [tuple(i) for i in column_list]
        xval = [x[0] for x in column_list]
        yval = [y[1] for y in column_list]
        # plotted with appropriate x and y labels
        plt.scatter(xval,yval)
        plt.xlabel(header_list[xdata])
        plt.ylabel(header_list[column])
        plt.show()

mass_plotting('Dataset.csv', 9, 19, 57)
```

2.4 Counting the Occurances of all Climate Values

```
# A lot of the climate data had the same values since many came from the same area
# I want to count all the occurances for a value for all the climate data columns
# This function counts the occurances for columns 9-56,
# and pushes the counts to a dictionary for each column
def count_to_dict(file, ystart, yend):
    # opening data file
    data_file = open(file)
    # importing pandas
    import pandas as pd
    # assigning tmp_data variable to pandas file
    tmp_data = pd.read_csv(data_file)
    # assigning data variable to numpy file
    data = tmp_data.to_numpy()
    # importing matplotlib.pyplot as plt since I need it later
    for column in range(ystart,yend):
        header_list = list(tmp_data)
        # for loop searches through columns 19-57 which are the climate ones
        column_list = tmp_data["{}".format(header_list[column])].tolist()
# {}.format used since column names change
        count_dict = dict() # new dict
        # for loop finds all the occurances of an element and
# assigns the value as frequency of the element (key)
```

```
        for ii in column_list:
            count_dict[ii] = count_dict.get(ii, 0) + 1
        # closing the file
        data_file.close()
        print(count_dict) # printing dict for one column
        print("\n") # helps to space them out

count_to_dict('Dataset.csv', 19, 57)
```

2.5 Using Regex to Count all the Occurances of Individuals' Codes

```
# setting up a function that will count all the cases of a regex search command in a column
def regex_count(file, num=1, col_regex=r'\b[\w]+\b'):
    # importing re module
    import re
    # opening data file
    data_file = open(file)
    # importing pandas
    import pandas as pd
    # assigning tmp_data variable to pandas file
    tmp_data = pd.read_csv(data_file)
    # assigning data variable to numpy file
    data = tmp_data.to_numpy()
    # Creating a list called cols that contains all the data from the col
    cols = (data[:, num]).tolist()
    # converting column list to string
    cols_string = ' '.join(cols)
    # extracting all cases of the regex search
    find_regex = re.findall(col_regex, cols_string)
    # new empty list
    regex_list = []
    # appending all finds to empty list
    for ii in find_regex:
        regex_list.append(ii)
    # new empty dictionary
    regex_counts = dict()
    # keys will be regex and values will be occurances
    for iii in regex_list:
        regex_counts[iii] = regex_counts.get(iii, 0) + 1
    # sorting dictionary by value (occurances)
    sorted_regex_count = {k: v for k, v in sorted(regex_counts.items(), key=lambda item: item[1])}
    print(sorted_regex_count)
    # closing the file
    data_file.close()

# counting cases of all words in 'individual' column
regex_count('Dataset.csv', 7)
```

2.6 Using ggplot2 to see Precise Locations of Species

```
# importing the tidyverse module
```

```
library(tidyverse)

# checking the working directory
getwd
# setting the working directory to Documents directory
setwd("C:/Users/Bhandal/Documents")

# loading the dataset with read_csv from the Downloads directory
dataset <- read_csv("C:/Users/Bhandal/Downloads/Dataset.csv")

# assigning the variable to the ggplot
#that will have Species for its x-axis
#and Latitude for its y-axis
Species_v_Latitude <- ggplot(data = dataset, mapping = aes(x = Species, y = Latitude))+
  # underlayed graph will be a jitter plot and
  #dots will be semi translucent green (for trees)
  geom_jitter(alpha = 0.2, color = "green")+
  # overlayed graph will be box plot (transparent)
  geom_boxplot(alpha = 0)+
  # setting formatting for title
  theme(plot.title = element_text(size = rel(1), face = "bold", hjust = 0.5))+
  # naming the title
  ggtitle("Species vs. Latitude")

# assigning the variable to the ggplot
#that will have Species for its x-axis
#and Longitude for its y-axis
Species_v_Longitude <- ggplot(data = dataset, mapping = aes(x = Species, y = Longitude))+
  # underlayed graph will be a jitter plot and
  #dots will be semi translucent green (for trees)
  geom_jitter(alpha = 0.2, color = "green")+
  # overlayed graph will be box plot (transparent)
  geom_boxplot(alpha = 0)+
  # setting formatting for title
  theme(plot.title = element_text(size = rel(1), face = "bold", hjust = 0.5))+
  # naming the title
  ggtitle("Species vs. Longitude")

# saving both plots in the working directory (Documents)
ggsave(Species_v_Latitude, file="Species_v_Latitude.png", width=6, height=4)
ggsave(Species_v_Longitude, file="Species_v_Longitude.png", width=6, height=4)
```

3 Results

3.1 All Locations

```
['Baja', 'Mexico', 'Texas', 'Florida', 'Costa Rica (Mothers)', 'Honduras',
 'Belize', nan, 'Cuba', 'South Carolina', 'Louisiana', 'North Carolina']
```

3.2 Location and Species Pairs

```
('Baja', 'BR') - 350
('Mexico', 'FU') - 656
('Texas', 'FU') - 92
('Florida', 'GE') - 289
('Florida', 'MN') - 126
('Costa Rica (Mothers)', 'OL') - 1028
('Honduras', 'OL') - 916
('Belize', 'OL') - 84
('Mexico', 'OL') - 1241
('Cuba', 'SA') - 120
('Florida', 'VI') - 475
('Texas', 'VI') - 34
('South Carolina', 'VI') - 100
('Louisiana', 'VI') - 60
('Mexico', 'VI') - 20
('North Carolina', 'VI') - 60
('Texas', 'HY') - 46
```

3.3 Foliar Area vs.Climate Data

Please see Figures 1 (on page 11) to 38 (on page 48).

3.4 Mode of each Climate Data Column

I plan to refine my code further to only find the mode of each climate data as that might be more useful than seeing dictionaries for each column

3.5 Occurances of Individuals' Codes

I plan to refine my code further to be a list of tuples so it is easier to see in the document.

3.6 Latitude and Longitude

Please see Figures 39 (on page 49) and 40 (on page 50).

4 Discussion

I plan to expand on the usefulness of any extracted data and the functions that allowed me to acheive that.

References

[1] Jeannine Cavender-Bares, Kaoru Kitajima, and FA Bazzaz. Multiple trait associations in relation to habitat differentiation among 17 floridian oak species. *Ecological Monographs*, 74(4):635–662, 2004.

[2] William M Jolly, Ramakrishna Nemani, and Steven W Running. A generalized, bioclimatic index to predict foliar phenology in response to climate. *Global Change Biology*, 11(4):619–632, 2005.

Figures

Figure 1: Foliar Area vs Climate Data 19

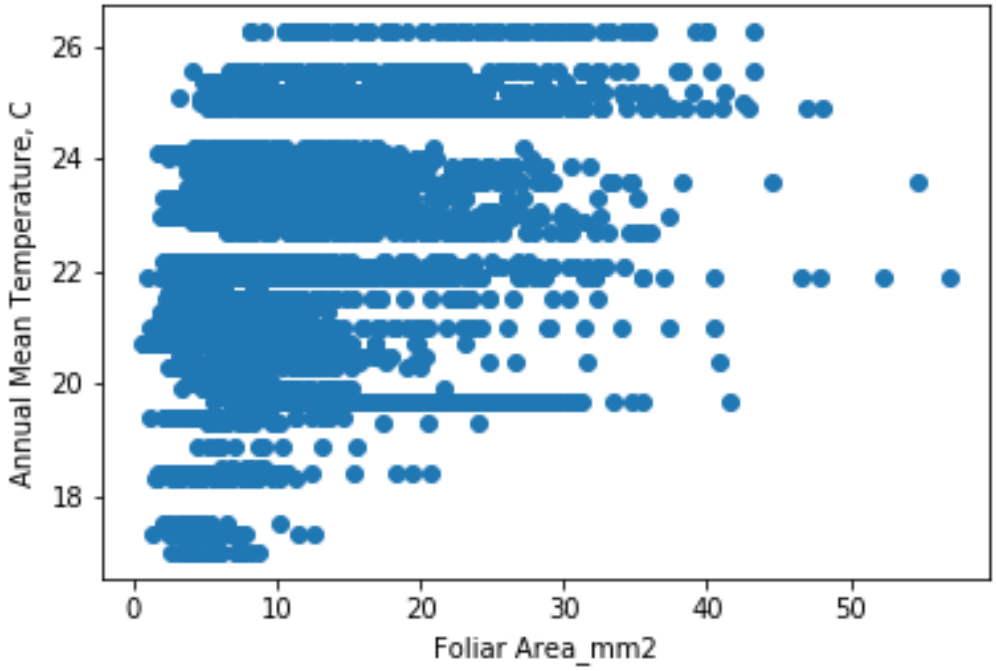

Figure 2: Foliar Area vs Climate Data 20

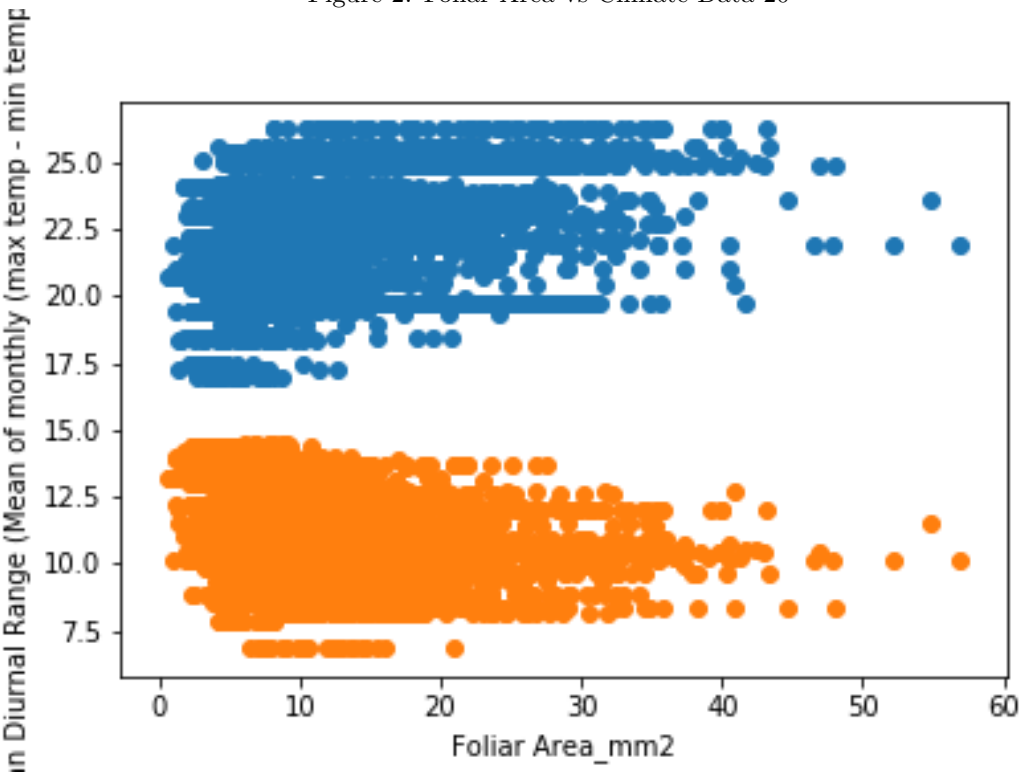

Figure 3: Foliar Area vs Climate Data 21

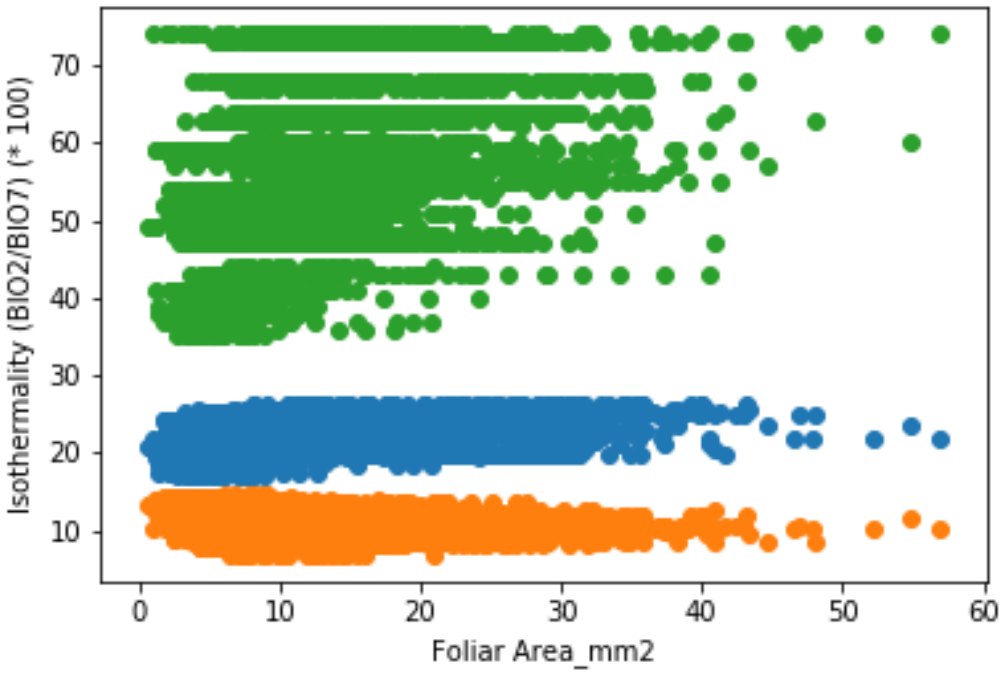

Figure 4: Foliar Area vs Climate Data 22

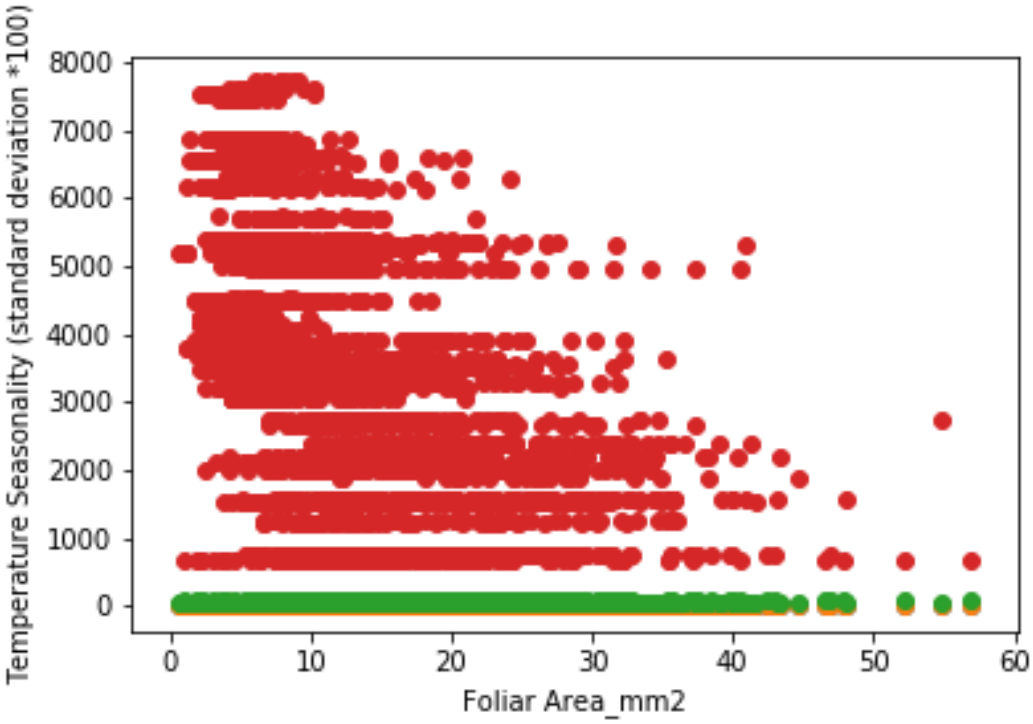

Figure 5: Foliar Area vs Climate Data 23

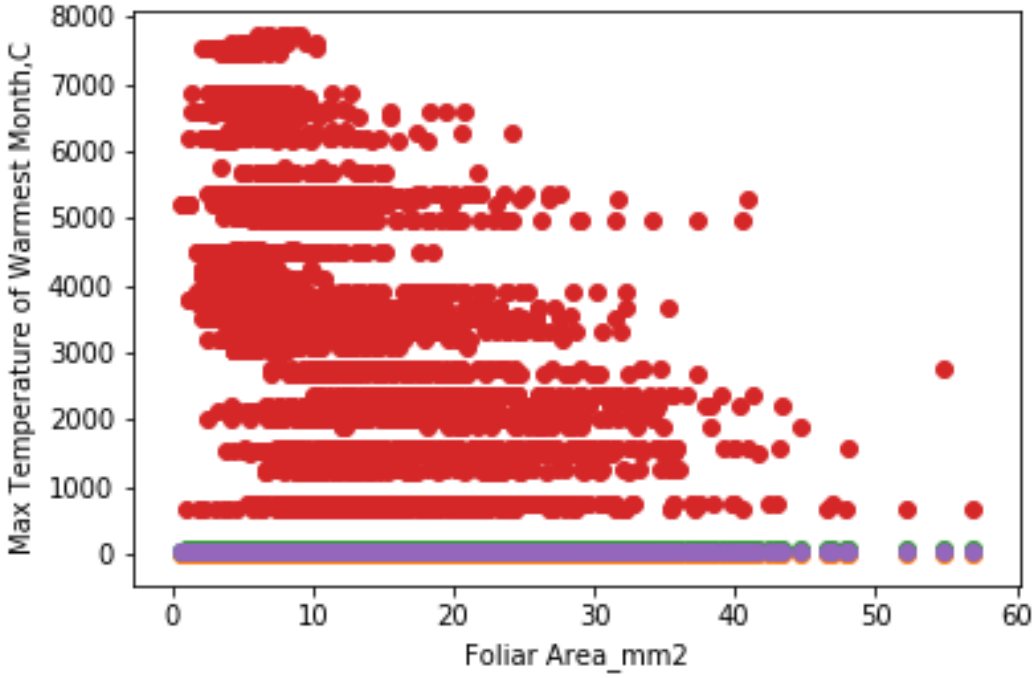

Figure 6: Foliar Area vs Climate Data 24

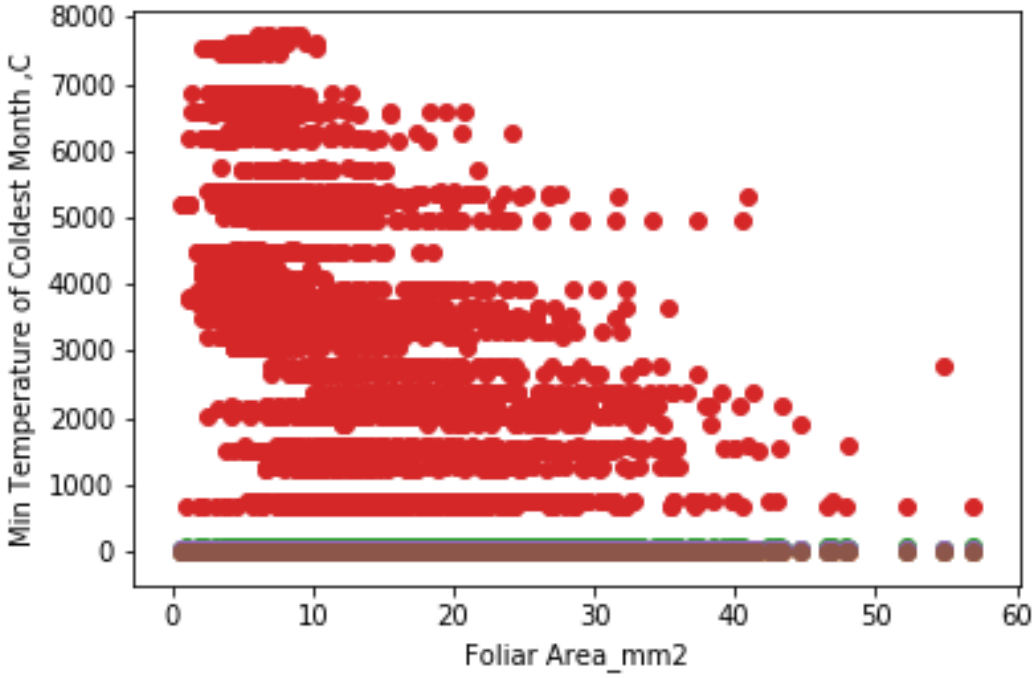

Figure 7: Foliar Area vs Climate Data 25

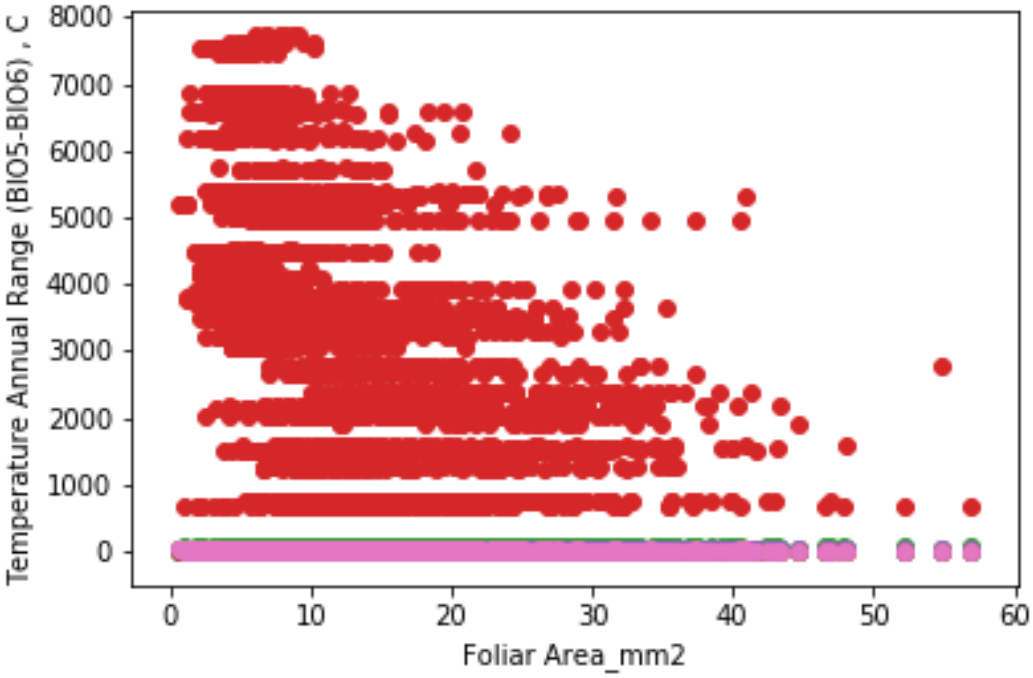

Figure 8: Foliar Area vs Climate Data 26

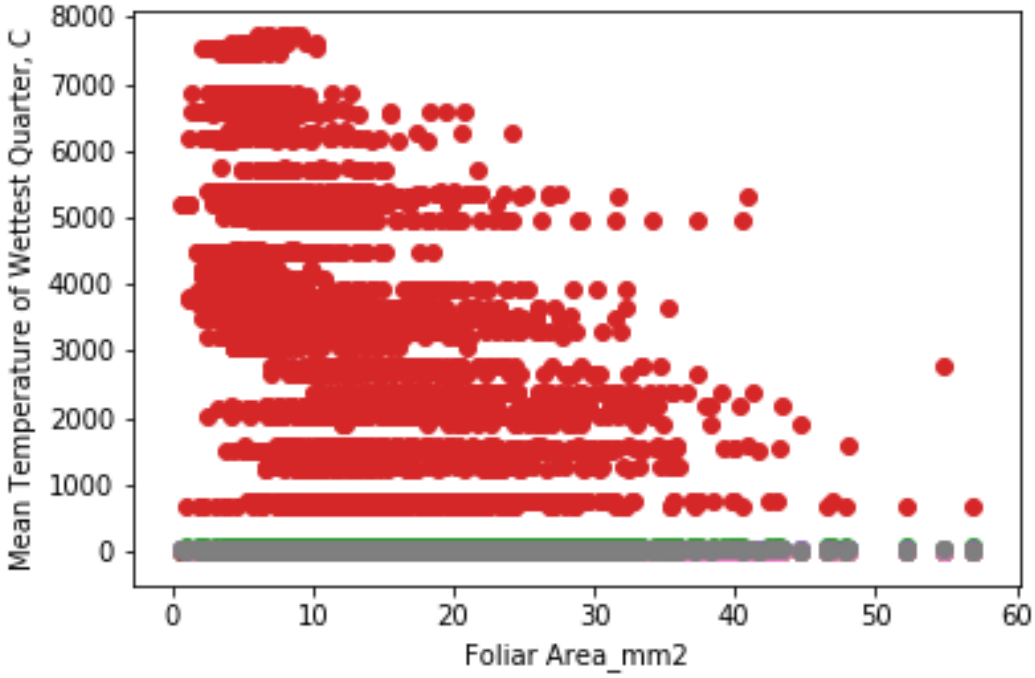

Figure 9: Foliar Area vs Climate Data 27

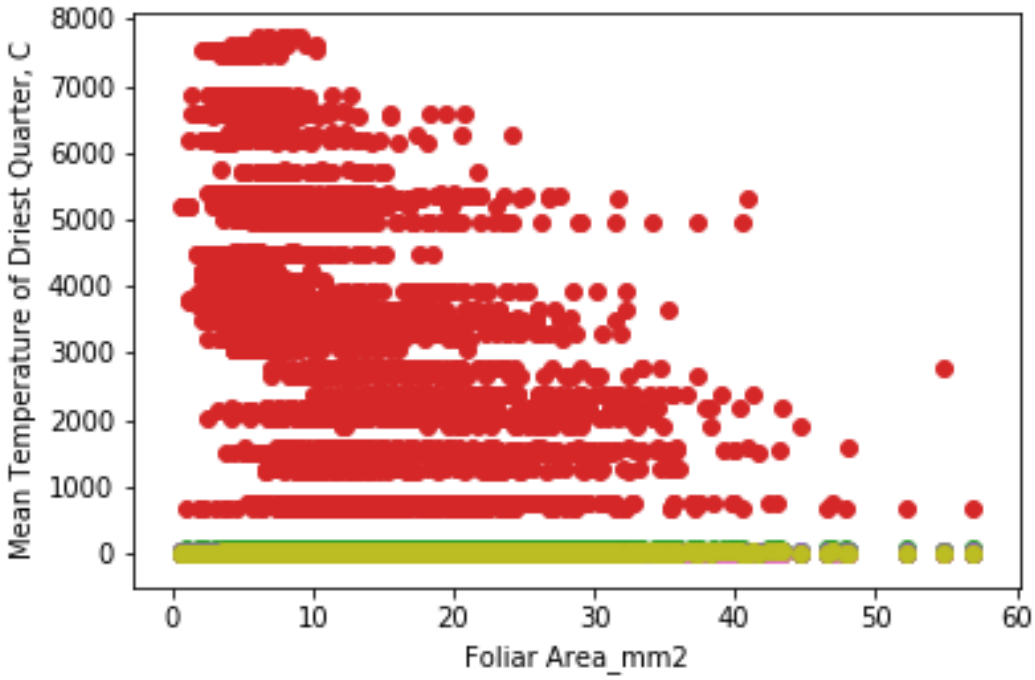

Figure 10: Foliar Area vs Climate Data 28

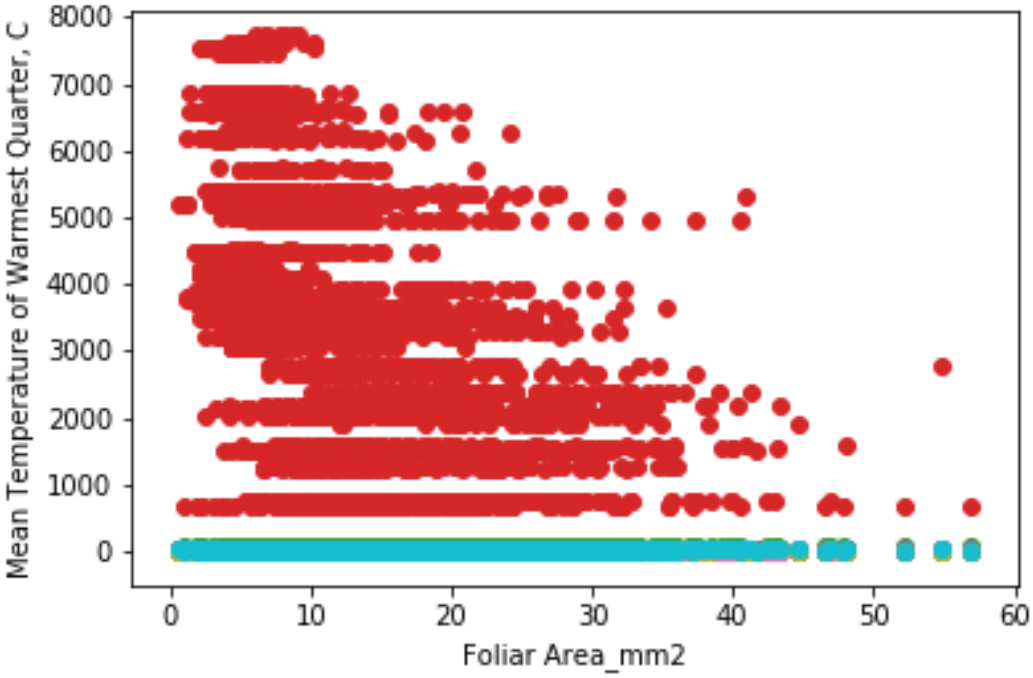

Figure 11: Foliar Area vs Climate Data 29

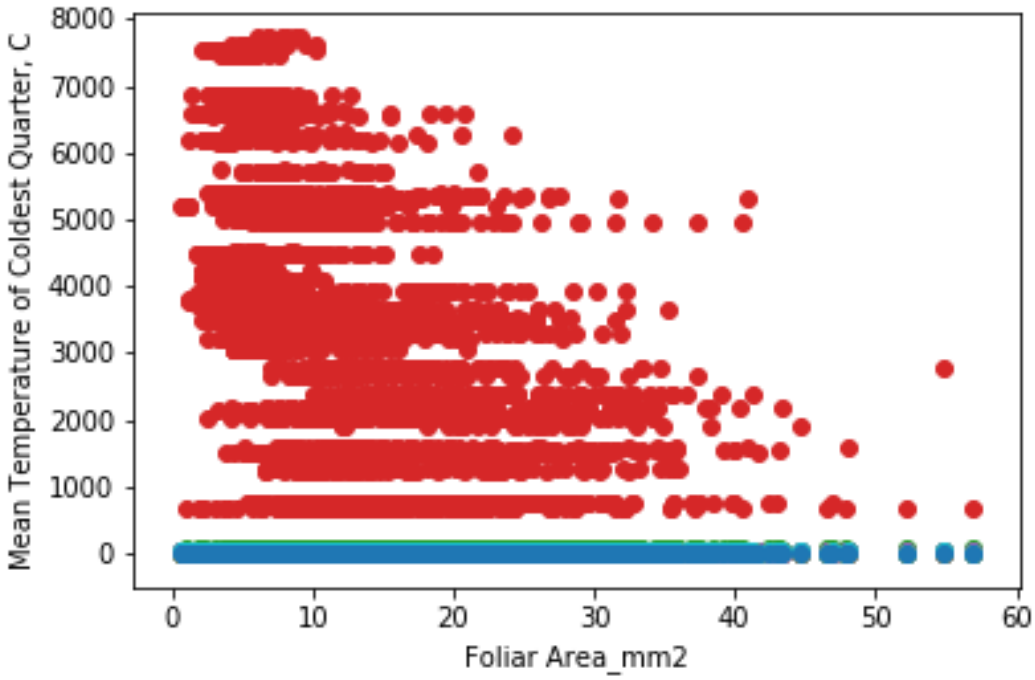

Figure 12: Foliar Area vs Climate Data 30

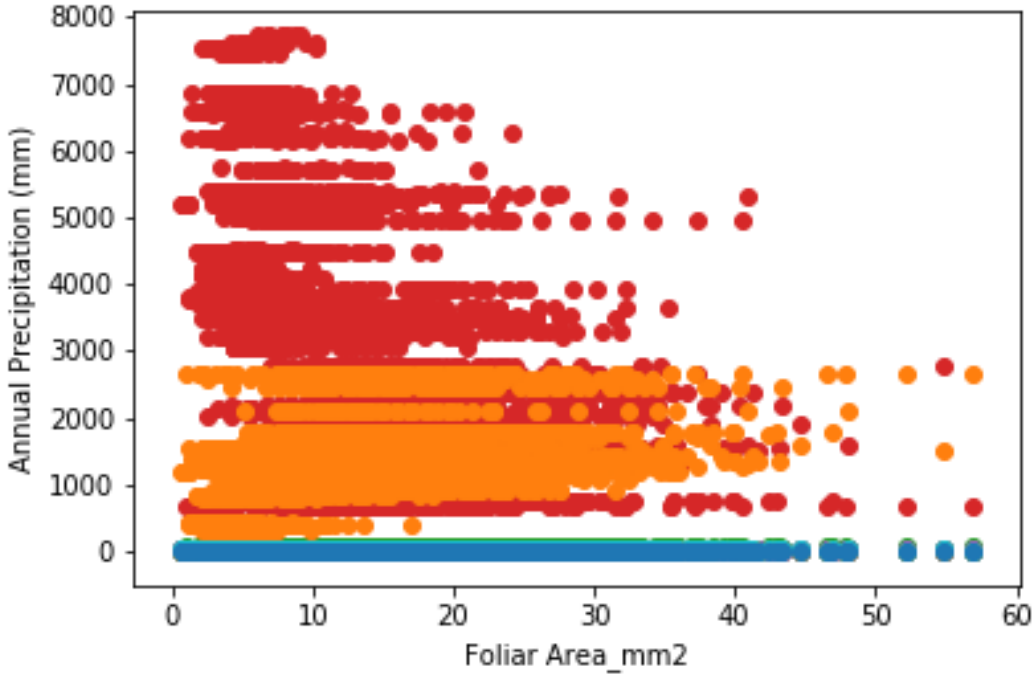

Figure 13: Foliar Area vs Climate Data 31

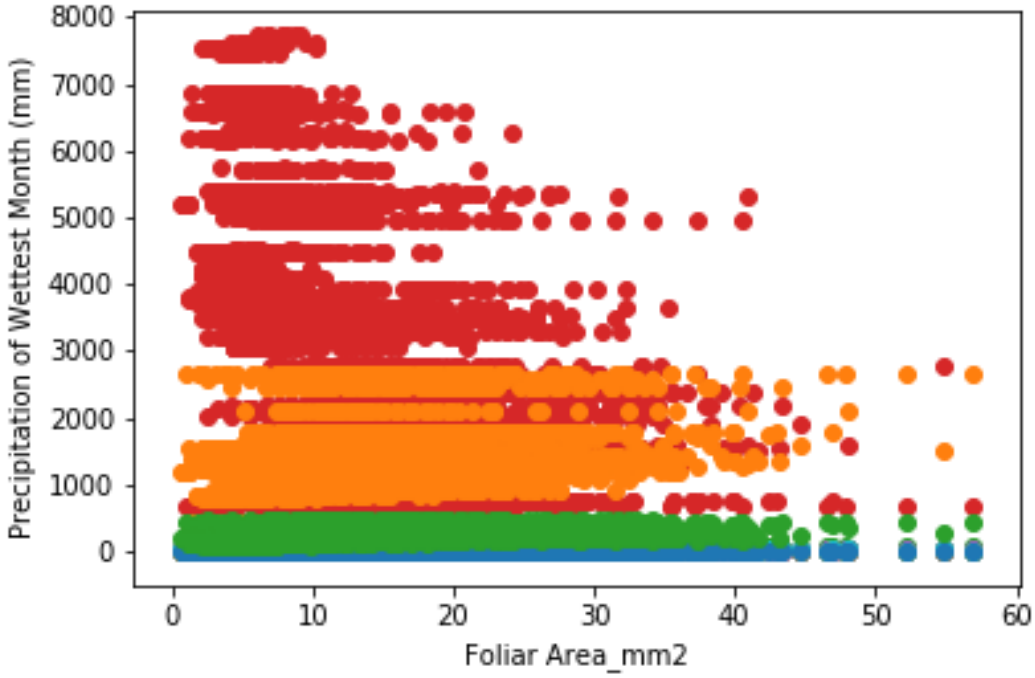

Figure 14: Foliar Area vs Climate Data 32

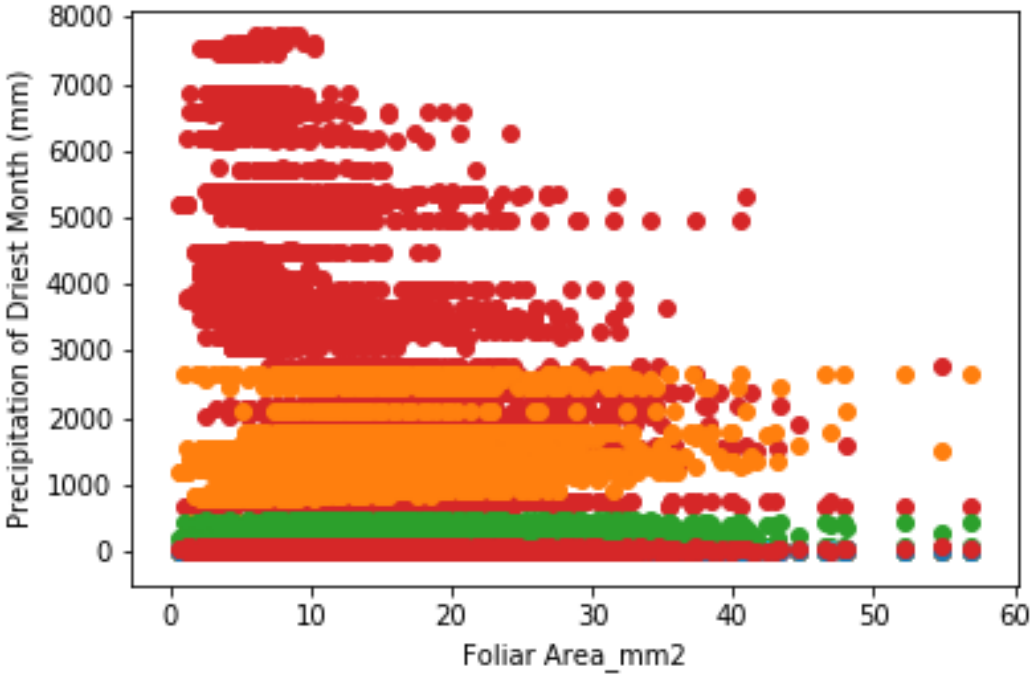

Figure 15: Foliar Area vs Climate Data 33

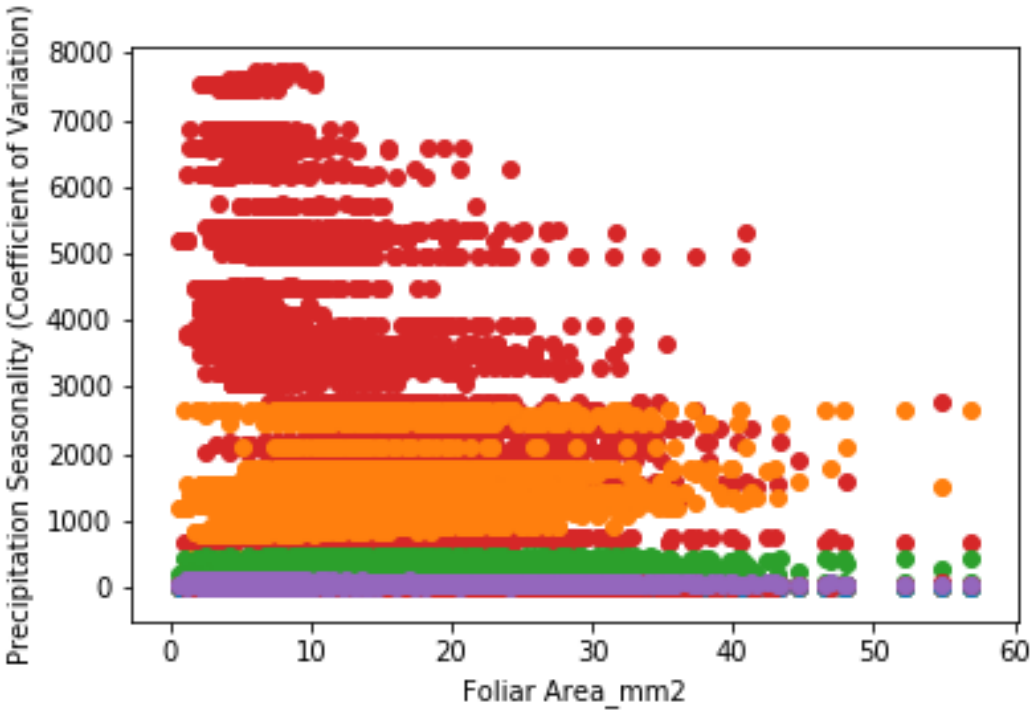

Figure 16: Foliar Area vs Climate Data 34

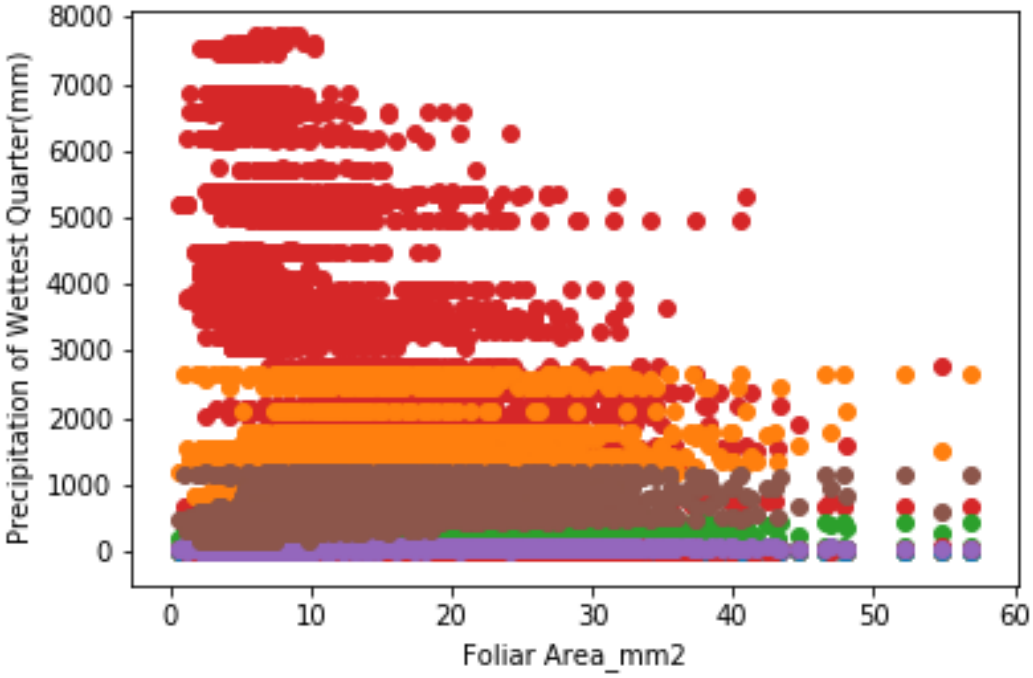

Figure 17: Foliar Area vs Climate Data 35

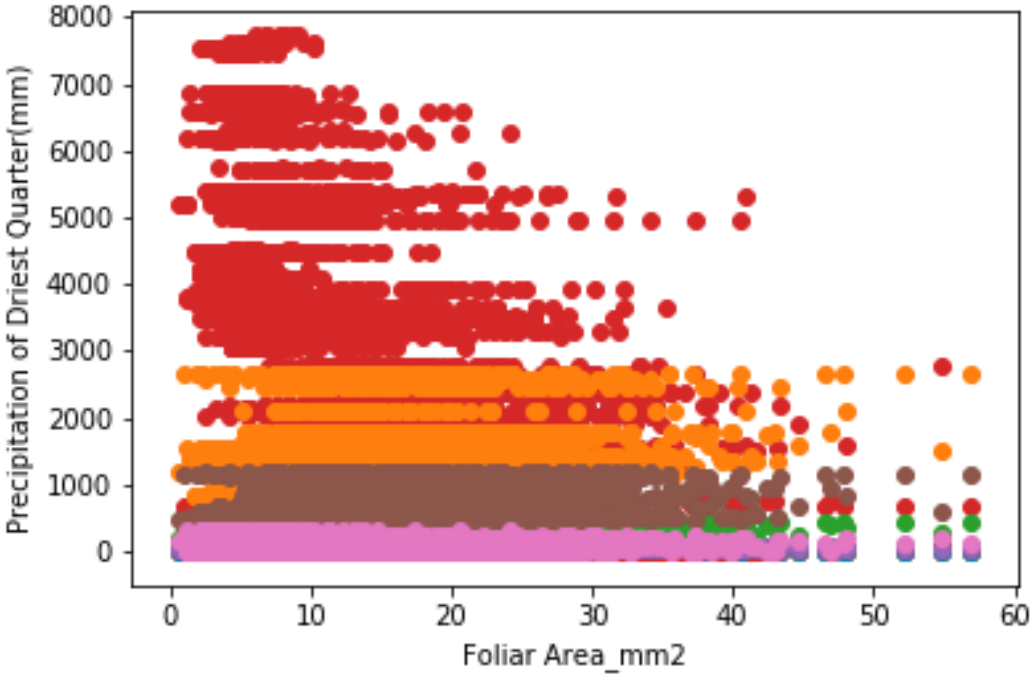

Figure 18: Foliar Area vs Climate Data 36

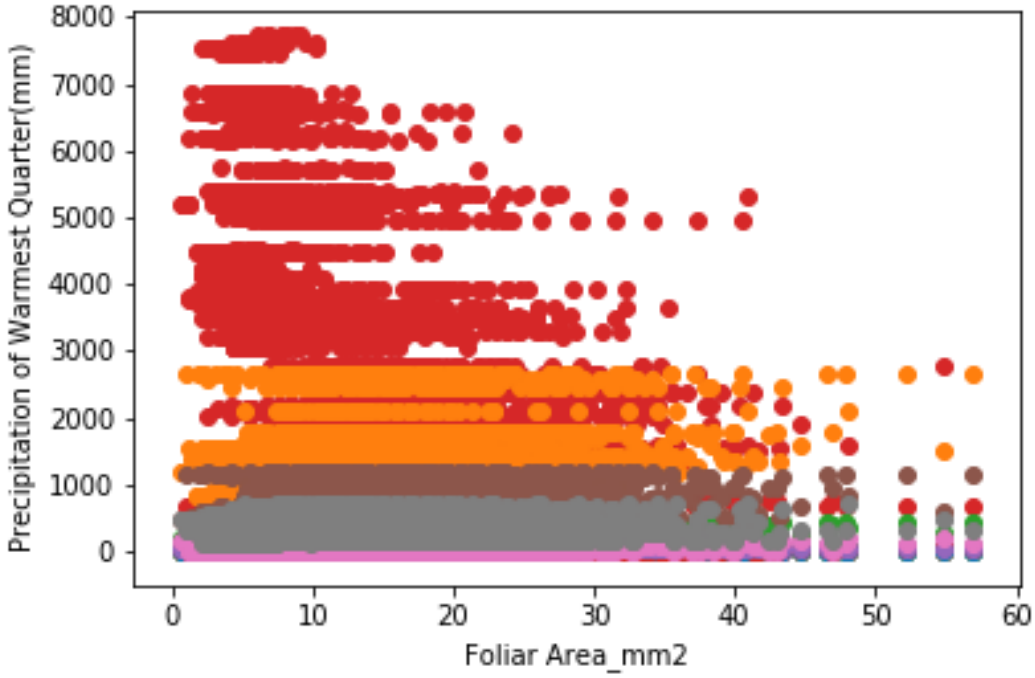

Figure 19: Foliar Area vs Climate Data 37

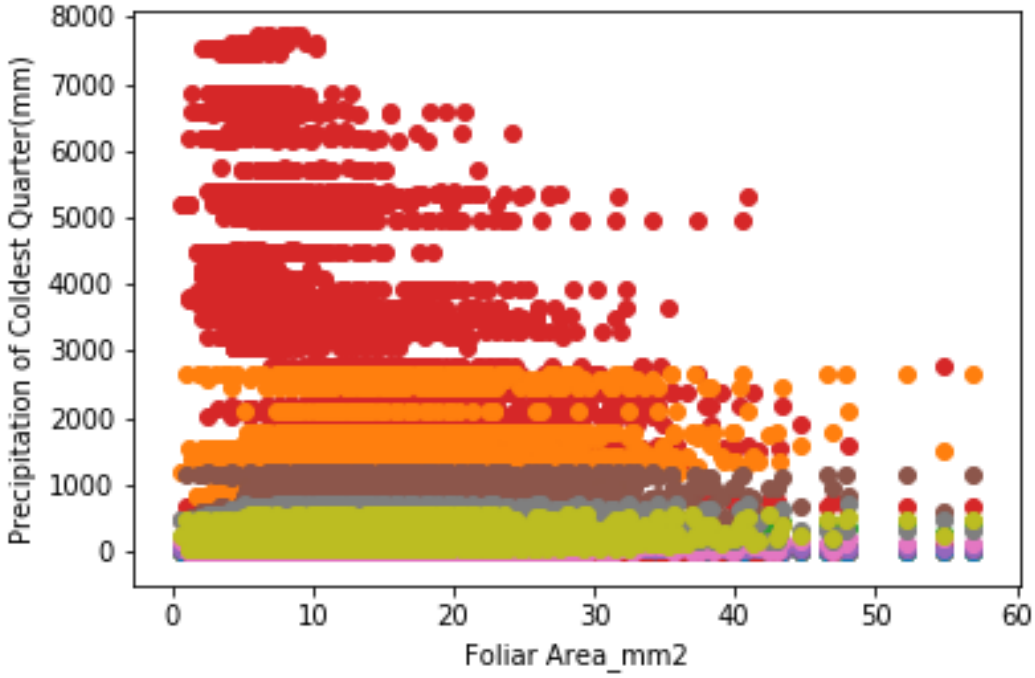

Figure 20: Foliar Area vs Climate Data 38

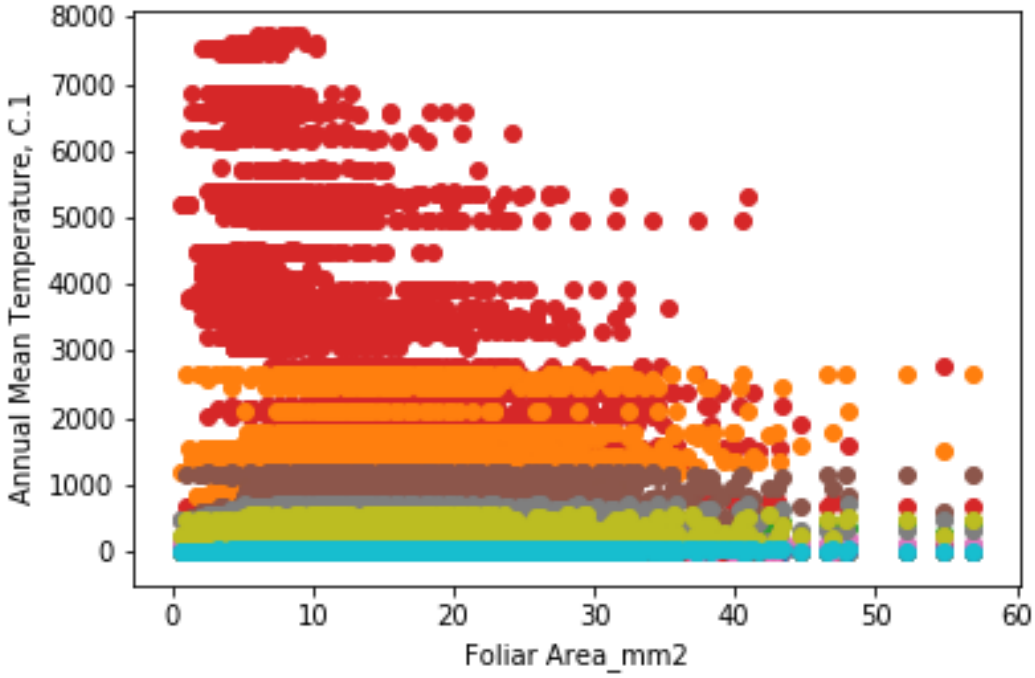

Figure 21: Foliar Area vs Climate Data 39

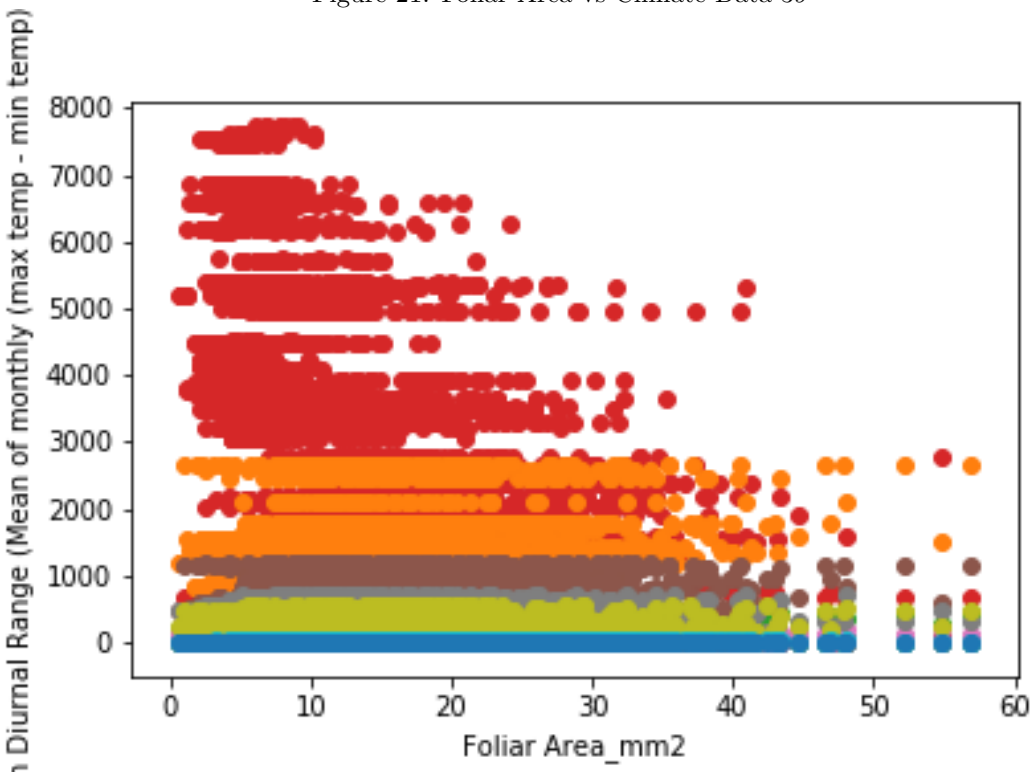

Figure 22: Foliar Area vs Climate Data 40

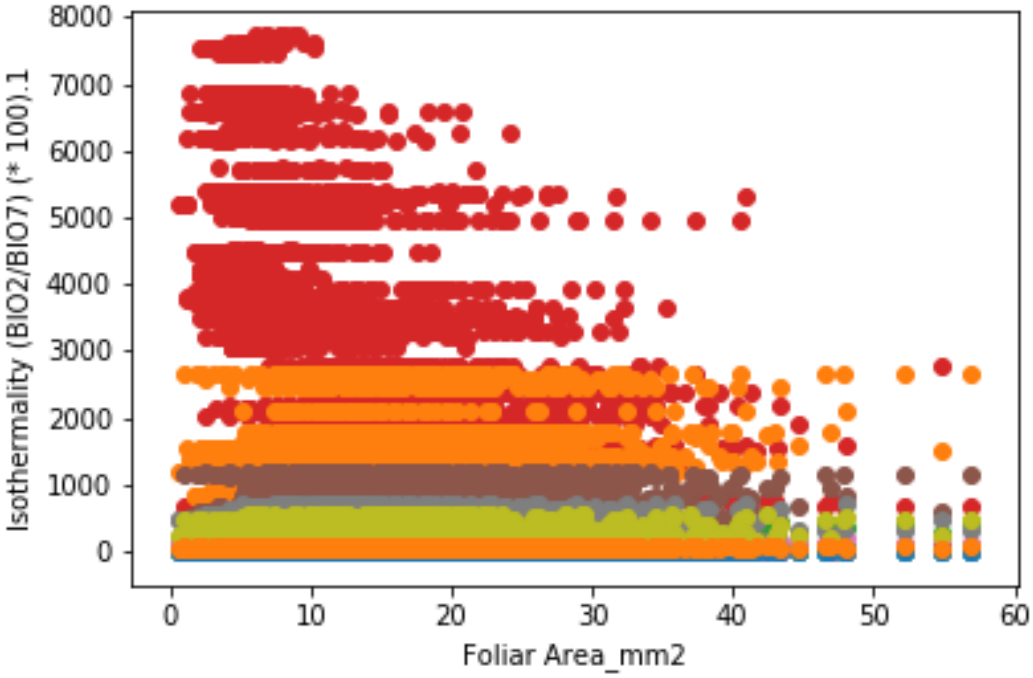

Figure 23: Foliar Area vs Climate Data 41

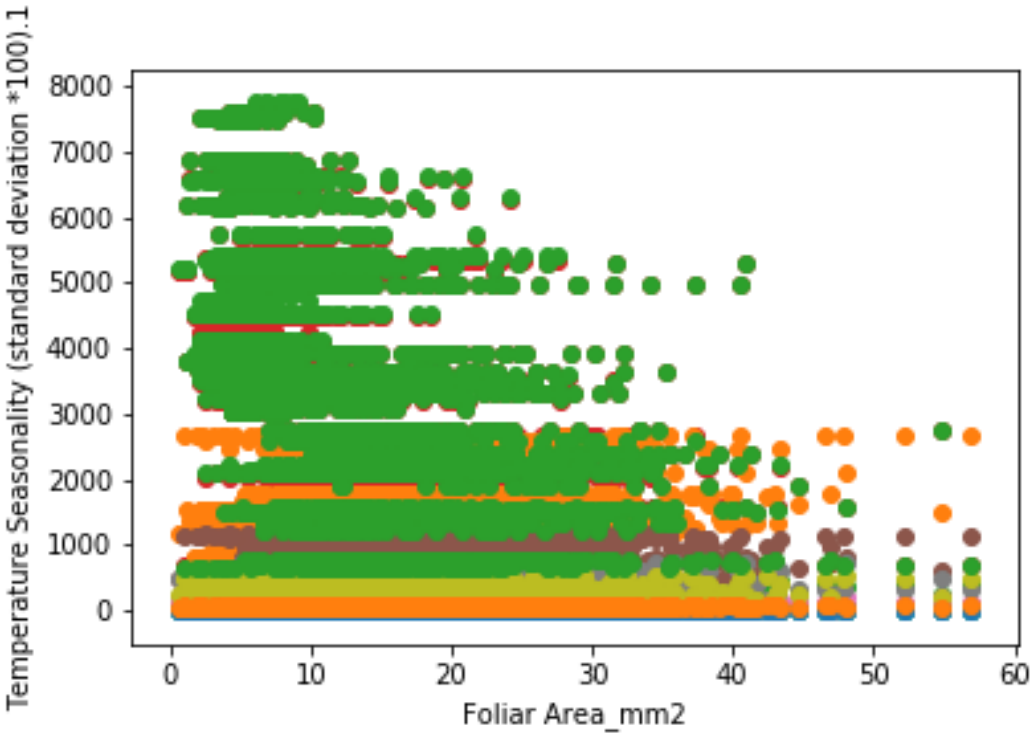

Figure 24: Foliar Area vs Climate Data 42

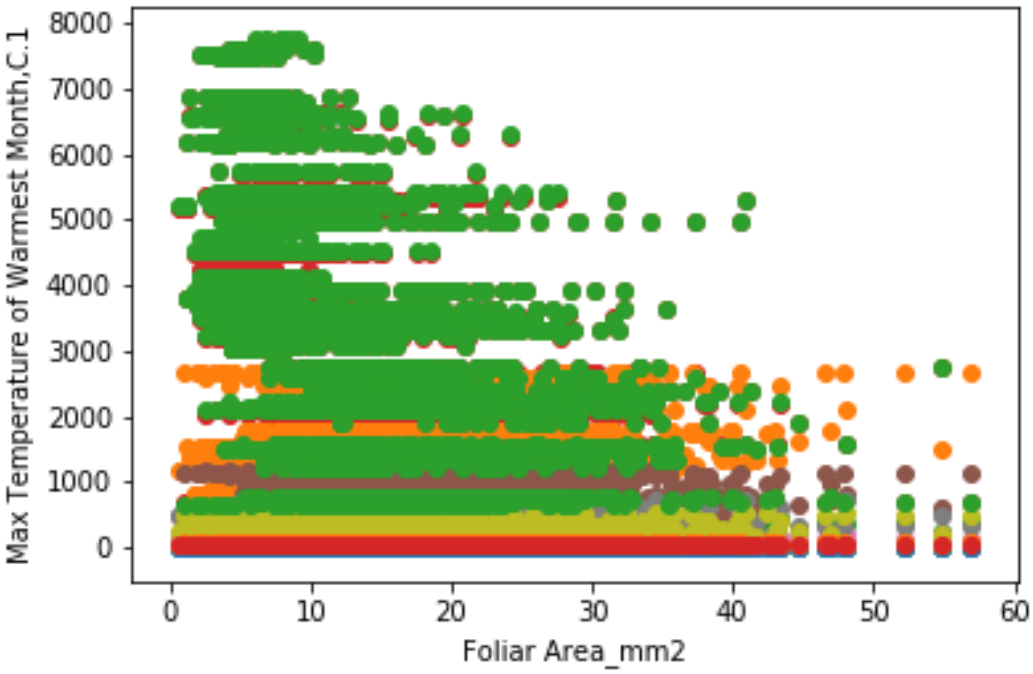

Figure 25: Foliar Area vs Climate Data 43

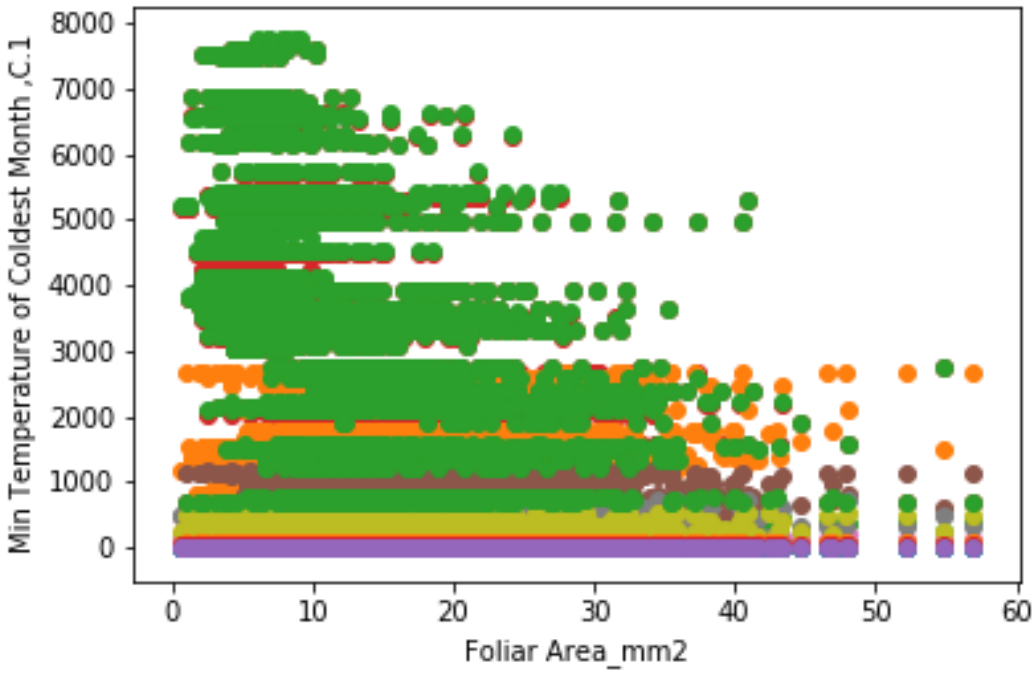

Figure 26: Foliar Area vs Climate Data 44

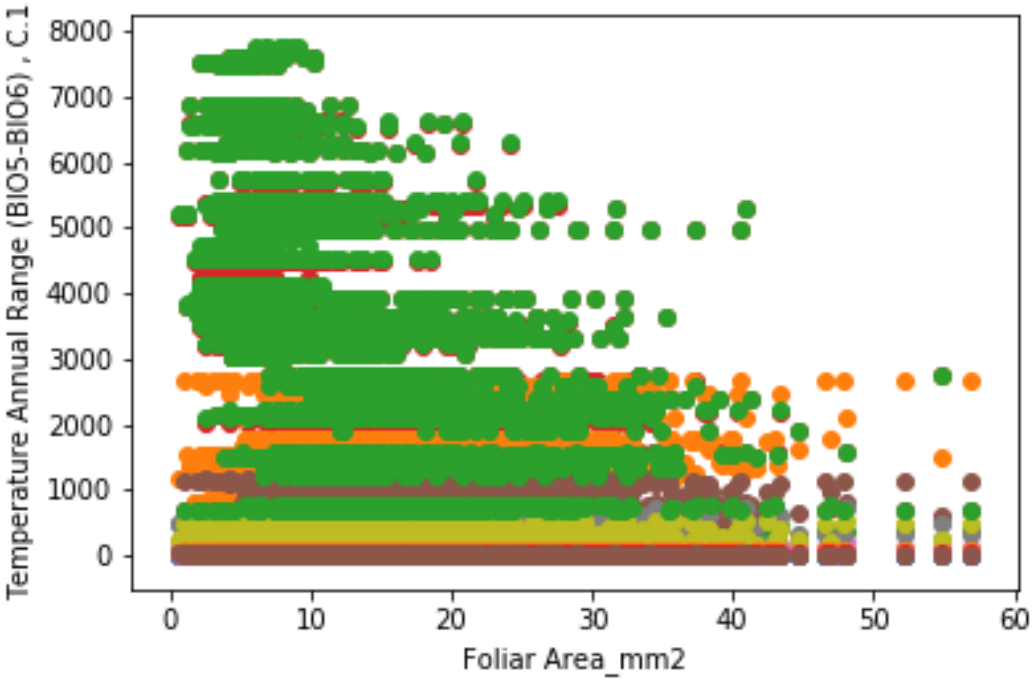

Figure 27: Foliar Area vs Climate Data 45

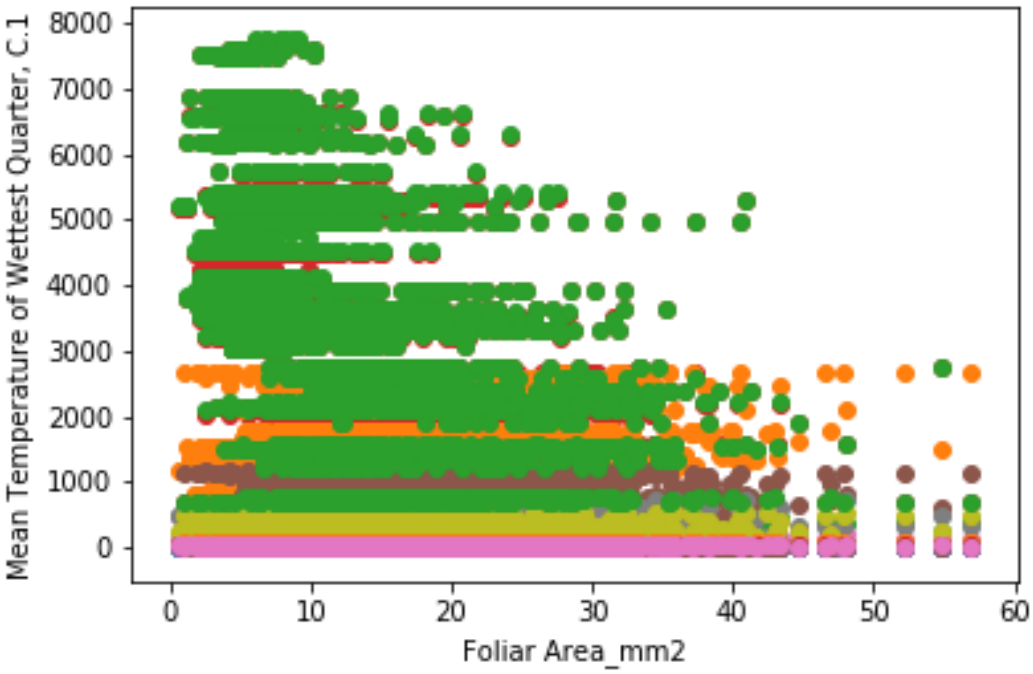

Figure 28: Foliar Area vs Climate Data 46

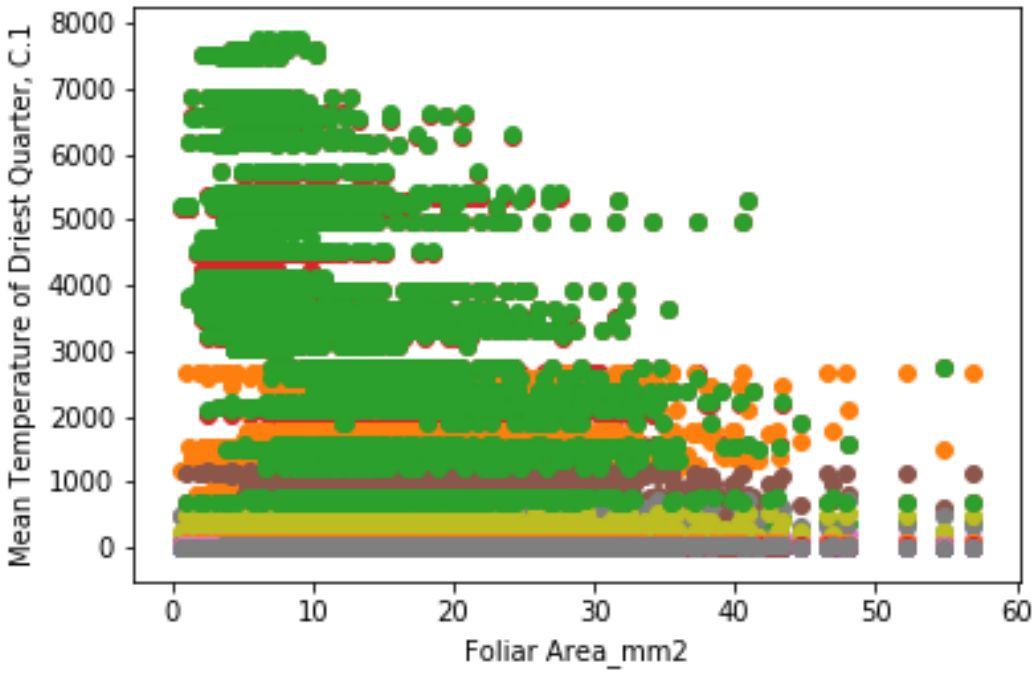

Figure 29: Foliar Area vs Climate Data 47

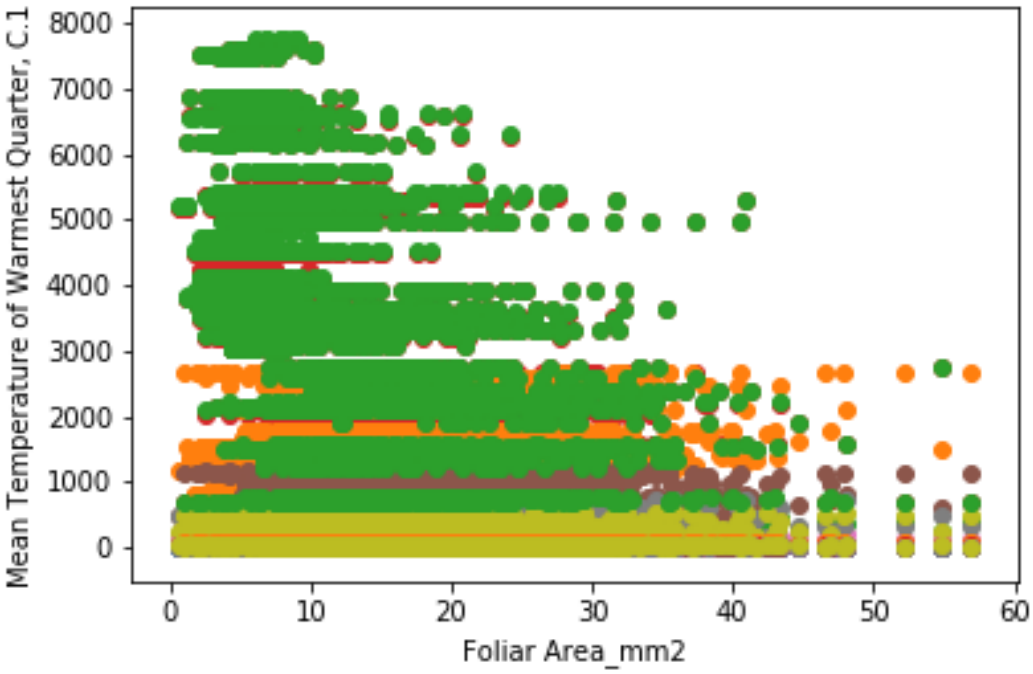

Figure 30: Foliar Area vs Climate Data 48

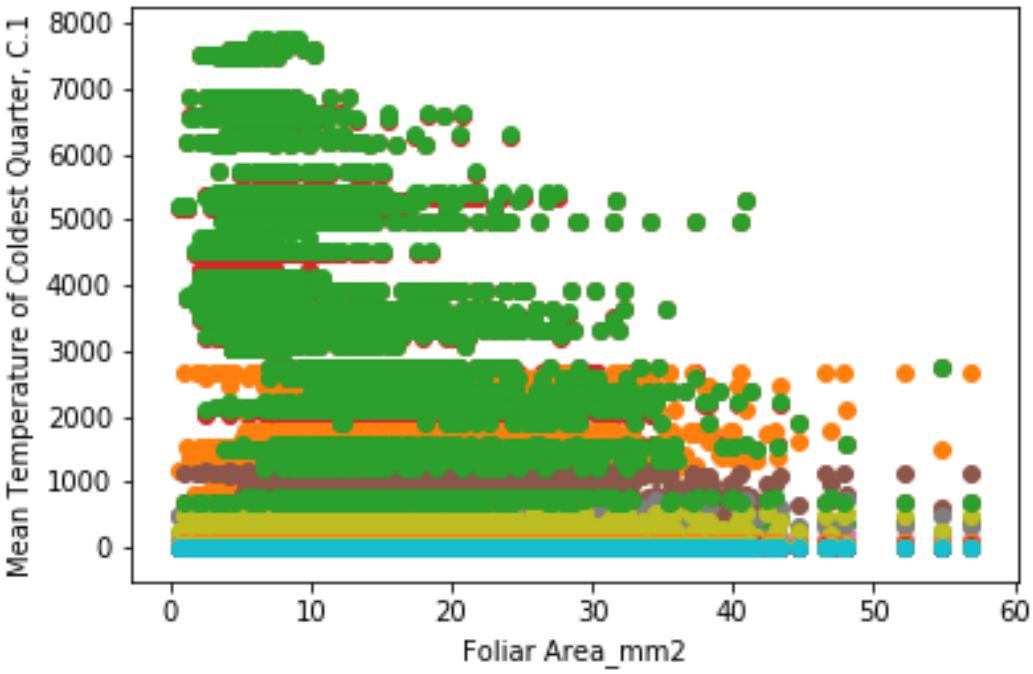

Figure 31: Foliar Area vs Climate Data 49

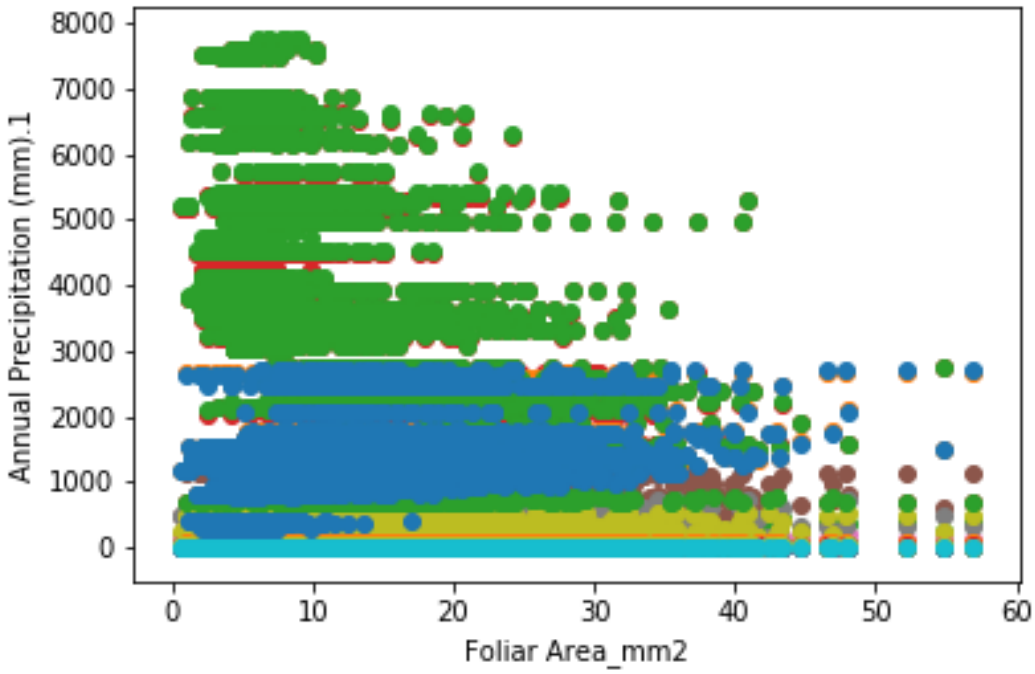

Figure 32: Foliar Area vs Climate Data 50

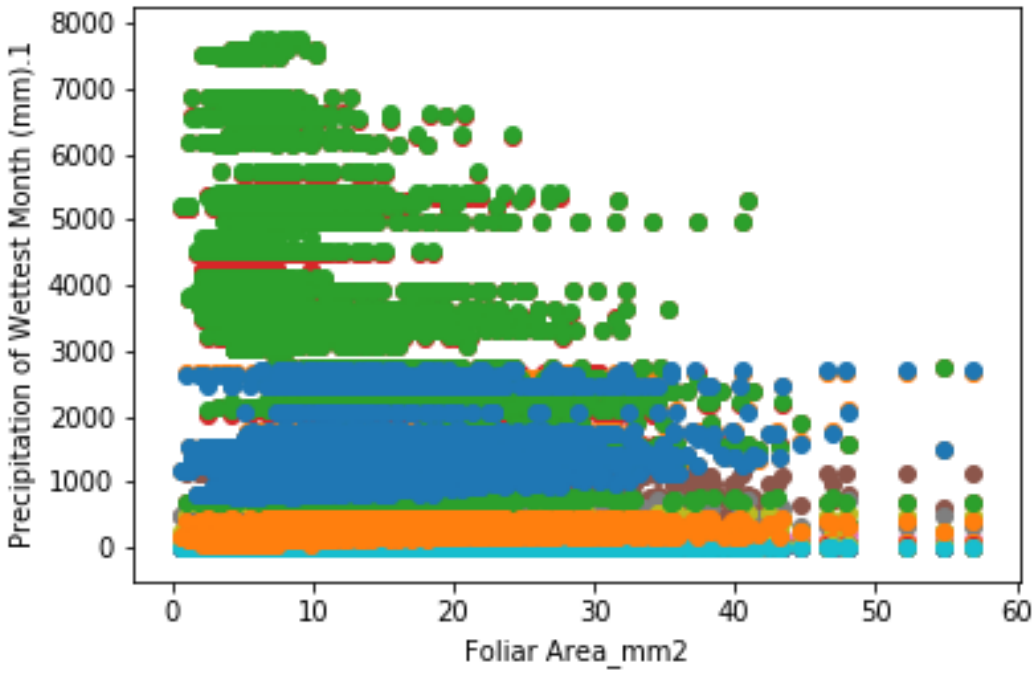

Figure 33: Foliar Area vs Climate Data 51

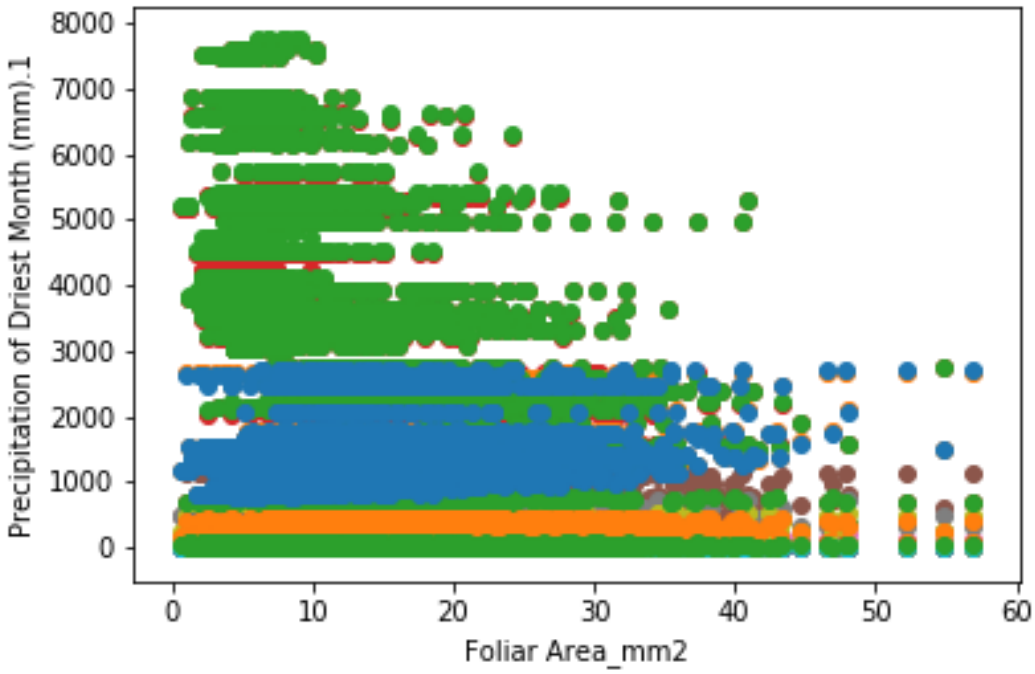

Figure 34: Foliar Area vs Climate Data 52

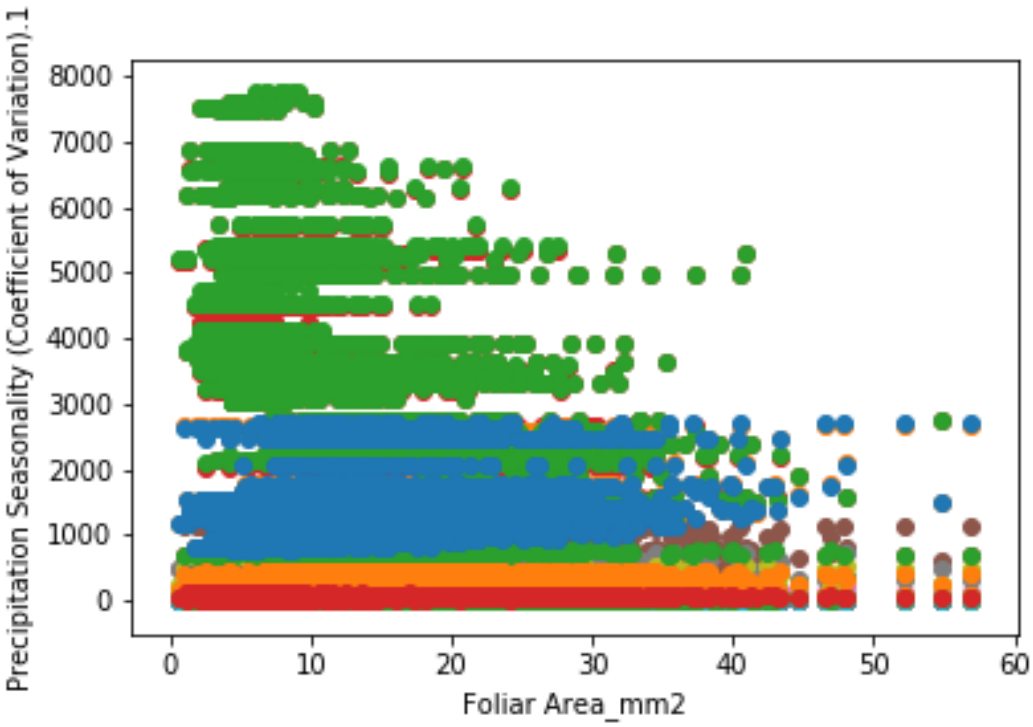

Figure 35: Foliar Area vs Climate Data 53

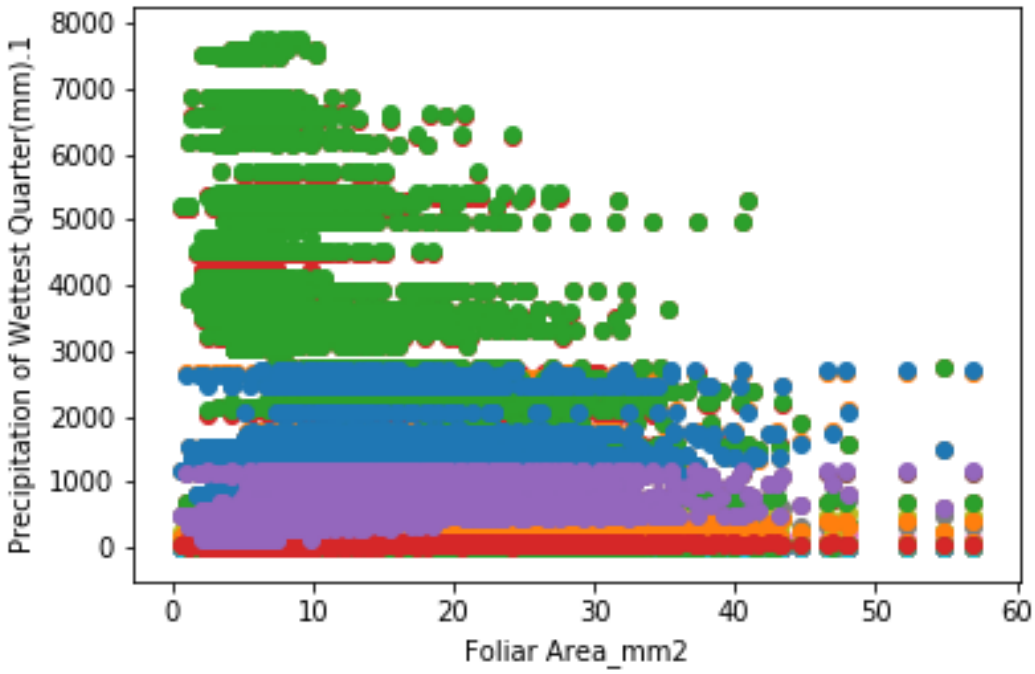

Figure 36: Foliar Area vs Climate Data 54

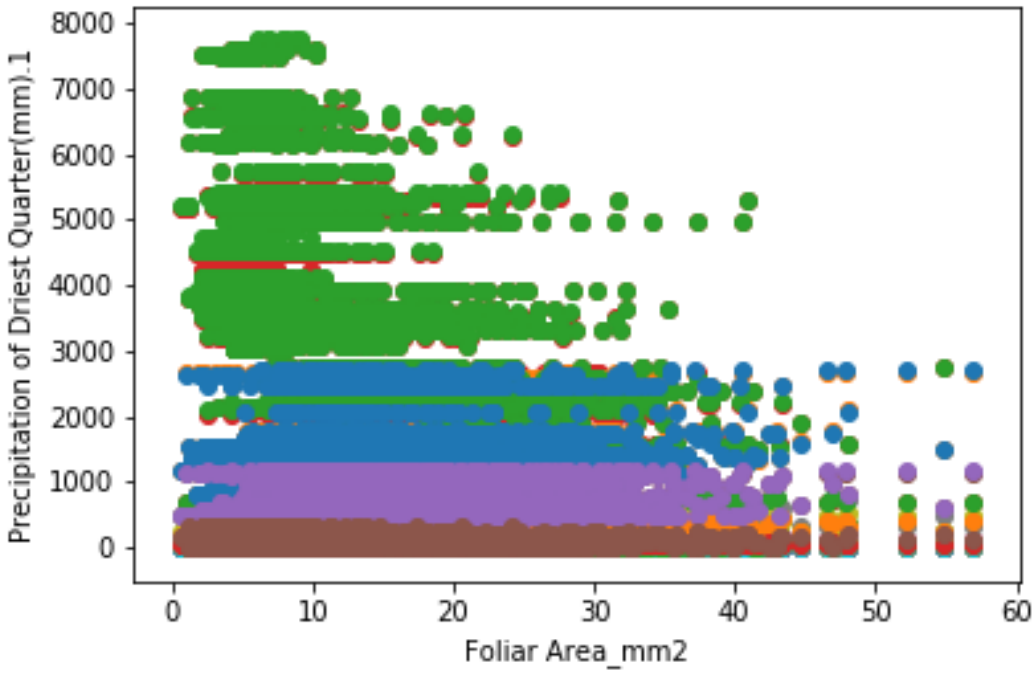

Figure 37: Foliar Area vs Climate Data 55

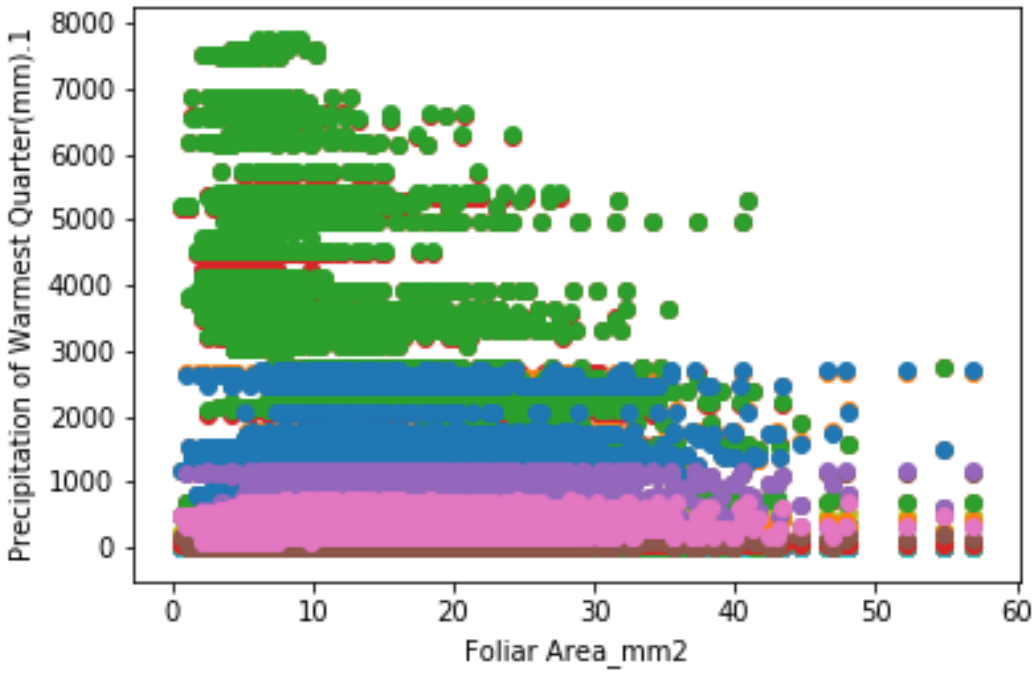

Figure 38: Foliar Area vs Climate Data 56

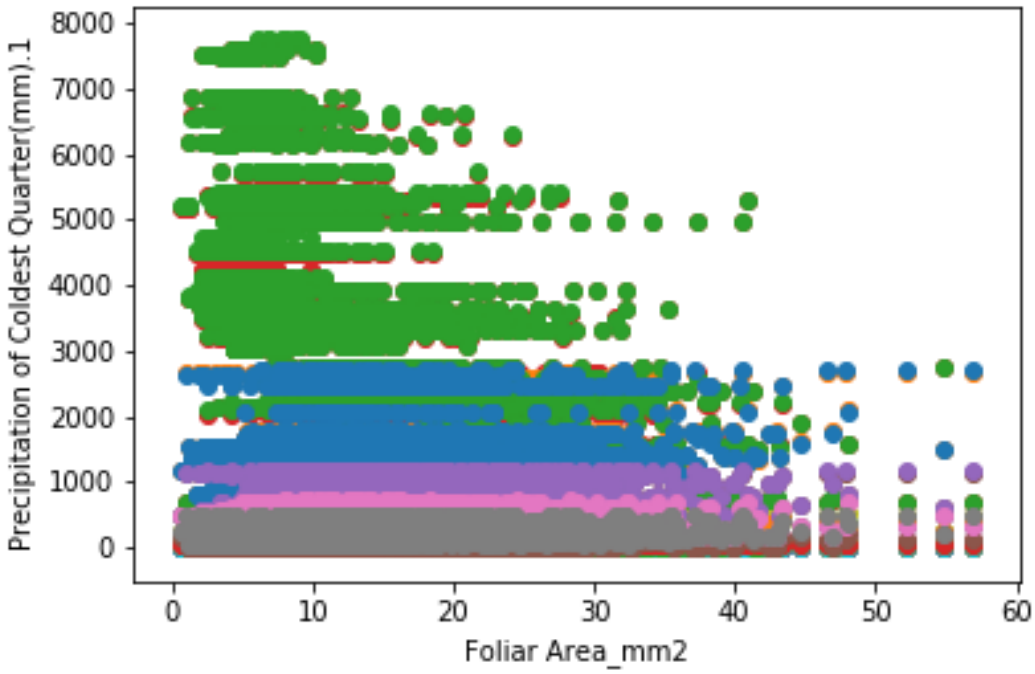

Figure 39: Species vs. Latitude

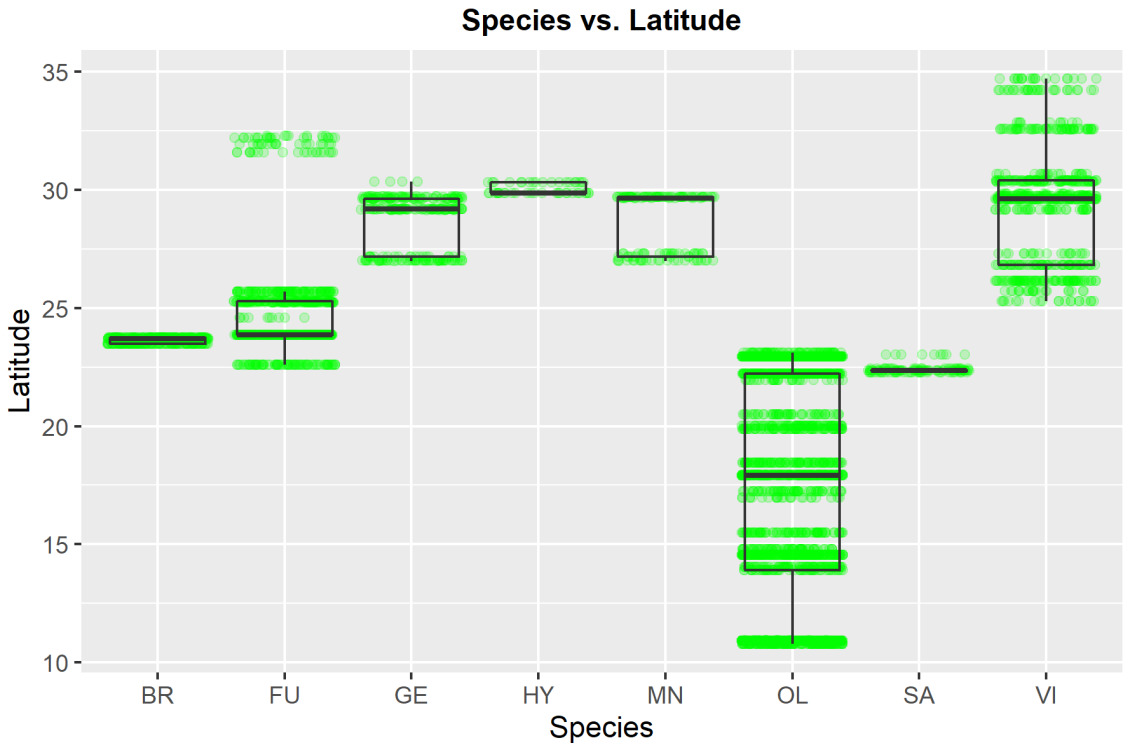

Figure 40: Species vs. Longitude