Team 7 Project Phase 1: Data Pre-processing Report

➢ Step 1: Merging 31 years data

- Merge each meteorological variable data for 31 years into one csv file.
- Example: z300_1980, z300_1981, …., z300_2010 => z300_31years.
- We have now 10 files for each of the 9 variables and Iowa.
- We have achieved this by using command prompt.

➢ Step 2: t9(11323 X (5328*9)) creation

- We created a file t9.csv which contains 5328*9 columns and 11323 rows.
- This is done by concatenating (side by side) each of the 9 files created in step 1.
- z300_31years, z500_31years,….., pw_31years => t9.csv

Following is the code for this step: t9_creation.py

```
#This script is used to create the t9 table. This table will contain 5328*9
columns and 11323 rows
import pandas as pd
fileNames = ["z1000", "u300", "v300", "u850", "v850", "t850" ,"pw"]

df = pd.read_csv("raw_data/z300_31years.csv", header = None)
df1 = pd.read_csv("raw_data/z500_31years.csv", header = None)
#we initially create result with concat of z300_31years and z500_31years
result = pd.concat([df, df1], axis = 1)

#For rest of 7 meteorological variable we do concatenation in the following
loop
for fileName in fileNames:
    df = pd.read_csv("raw_data/" + fileName + "_31years.csv", header = None)
    result = pd.concat([result, df], axis = 1)

#Write result to get the t9.csv
result.to_csv('t9.csv', index = False)
```

➢ Step 3: Class Label calculation

We used iowa_31years.csv for calculating Class Labels:

- Calculate sum of first 15 rows from iowa_31years.csv and store the value.
- Repeat this for remaining rows of iowa_31years.csv until 11309th row. From 11310th row onwards we don't have sufficient data to calculate the sum.
- Calculate 95th percentile of the store values as p.
- Iterate through the stored values and for each value greater than p, assign Class Label as 1, else assign 0.
- Remove first 14 class labels. Since 15th class label should be assigned to the first row in features table.
- We now have 11295 class labels.

Following is the code for this step: classLabel_creation.py

```
#This script is for creating the class labels
import csv
import itertools
#We just need the iowa data for 31 years for creating the class labels
fileName = "raw_data/iowa_31years.csv"
sumList = []
for time in range(0,11309):
    with open(fileName, 'r') as csvFileIn:
        csvReader = csv.reader(csvFileIn)
        sum = 0
        #take 15 days sum and put into sum
        for row in itertools.islice(csvReader, time, time+15):
            sum = sum + float(row[0])
    #append each sum to sumList
    sumList.append(sum)

import numpy as np
#convert sumList to numpy array, so as to do percentile
sumArray = np.array(sumList)
#p is the 95th percentile
p = np.percentile(sumArray, 95)

classLabel = []
for row in sumList:
    #if sum is > than p, classLabel is 1. Else its 0
    if row > p:
        classLabel.append(1)
    else:
        classLabel.append(0)

#we need only class labels from 1-15-1980, so remove first 14 rows
classLabel = classLabel[14:]

#Write into class_label.csv
with open("classLabels.csv", 'wb') as csvFileOut:
    csvWriter = csv.writer(csvFileOut)
    for label in classLabel:
        csvWriter.writerow([label])
```

➢ Step 4: Creating features

The feature.csv is created using the t9 file created in step2:

- Extract first 10 rows into a Dataframe. Concatenate each of the 10 rows to form a single row and append the resultant row into features.csv.
- Repeat this for remaining rows of t9 until 11295th row.
- Now we have the features table which has 5328*9*10 columns, i.e. 479520 columns and 11295 rows.

Following is the code for this step: feature_creation.py

```
#This script creates the feature table using t9 table(11323 rows and 5328*9
columns)
import pandas as pd
import numpy

#idea is that each row in the final table is constructed from only 10 rows, so
extract only 10.
for m in range(0,11295):
    df = pd.read_csv('t9.csv', skiprows = m, nrows=10)
    #this is done to create a dummy dataframe with only a single element(0).
    final = numpy.zeros(shape=(1,1))
    df_final = pd.DataFrame(final)
    for time in range(0,10):
        #We cut out the timeth row
        temp= df.iloc[time:time+1,:]
        #We need to concatenate timeth row at the end of dummy table, so index
is made to 0
        temp.index = [0]
        df_final = pd.concat([df_final,temp], axis = 1)
    #this step is done to remove the 0 we initially created
    del df_final[0]
    #At this moment df_final will have one complete row of our final_table, so
append it to output.csv file.
    df_final.to_csv('features.csv', mode='a', header=False, index=False)
```

- **Step 4: Creating headers**

We created headers for each column in the features.csv. Below is the screenshot of first 7 column headers:

	1	2	3	4	5	6	7
1	T9z3S1	T9z3S2	T9z3S3	T9z3S4	T9z3S5	T9z3S6	T9z3S7

Finally, we merged the headers and features to get the final features.csv. Below is a screenshot of our final features.csv:

	1	2	3	4	5	6	7
1	T9z3S1	T9z3S2	T9z3S3	T9z3S4	T9z3S5	T9z3S6	T9z3S7
2	8458	8371	8267	8193	8185	8238	8314
3	8391	8320	8250	8200	8198	8265	8385
4	8223	8144	8132	8192	8298	8415	8525

Note: We have generated the header row as this might be useful in the next phase of this project.

Following is the code for this step: header_creation.py

```
#This script is used to create the headers for the feature table
#The headers will be in the format T9z3S1 T9z3S2 T9z3s3 .....
import csv
meteoVariable = ["z3", "z5", "z1", "u3", "v3", "u8", "v8", "t8", "pw"]
time = [9, 8, 7, 6, 5, 4, 3, 2, 1, 0]
header = []
#We need the header so as to keep track of the columns in the feature table(for
SAOLA)
for t in time:
    s = "T" + str(t)
    for mv in meteoVariable:
        s = s + mv + "S"
        for location in range(1,5329):
            temp = s
            temp = temp + str(location)
            header.append(temp)
        s = "T" + str(t)

with open("header.csv", 'wb') as csvFileOut:
    csvWriter = csv.writer(csvFileOut)
    csvWriter.writerow(header)
```

Finally, we now have the below tables:

1. features table with headers
2. class label table

Team member names and emails:

Name	**Email**
Sonica Kalmangi	sonica.kalmangi001@umb.edu
Sidhraj Solanki	Sidhraj.solanki001@umb.edu
Jacob Robins	jacob.robins001@umb.edu
Swapnil Patil	swapnil.patil001@umb.edu