

**ITCS 6112 Software System Design and Implementation**

**A PROJECT REPORT**

**ON**

**Analyzing NCAA Baseball dataset**

**TEAM MEMBERS**

Dhenuka Bhargavi Rangam 800963261

Hari Krishna Raja Karuppiah 800966462

Sunisha Chalasani 800962445

Vichithra Reddy Yellolu 800960406

**ABSTRACT**

In recent years, the amount of data generated has been increasing exponentially. The data is coming from different sources such as machine logs, gene sequencing, sensor networks, network flows and social media. Researchers in education and research sector from areas e.g. Bioinformatics, computer science, astronomy, environmental science has huge data sets and would like to analyze this data without worrying about the scale of data sets. Thus, there is an increasing demand of getting this data to work by storing and processing it in a horizontally scalable way. Analyzers who play a key role in the game decisions, also has huge data sets containing previous history of the matches and they want to analyze this data.

In the proposed system, data analysis on NCAA Baseball dataset is performed. The major idea of this project is to develop a system that analyses the baseball dataset and produce the result of the search criteria based on the year. System will result output such as winning trends of a team and other data related to search criteria.

The proposed system will use Python for programming, SAS Enterprise guide for analyzing the dataset and Tableau Desktop for drawing trends. The programming is used to give all the information related to a year. The Tableau is used to find the relation between different elements in the data set and show the graphs between them.

**INTRODUCTION**

The NACC Baseball data set has the following attributes such as year, Regional Site, Game, Winning Team, Runs, Seed, Losing Team, Runs, Seed.

The attributes can be well described as follows:

Year- describes the year in which the match is held

Regional Site- the place or the region where the match is held

Game- the number of the game that is held in that region

Winning Team- name of the winning team in the game

Runs- runs scored by the winning team

Seed- describes the seed of the winning team

Losing Team- name of the losing team in the game

Runs- runs scored by the losing team

Seed- describes the seed of the losing team

The system takes the data set as input and characterizes the data using SAS enterprise. Mining is performed on this data and an interface which displays the results of the problem solution is developed.

The interface outputs the winning trends of a match in a region in each year. The user interface is developed using python programming which takes the year as input.

**TECHNICAL DOCUMENTATION**

**Python:**

Python is a widely used high-level, general-purpose, interpreted, dynamic programming language. Its design philosophy emphasizes code readability, and its syntax allows programmers to express concepts in fewer lines of code than possible in languages such as C++ or Java. The language provides constructs intended to enable writing clear programs on both a small and large scale.

Python supports multi-programming paradigm, including object-oriented, imperative, and functional programming or procedural styles. It features a dynamic type system and automatic memory management and has a large and comprehensive standard library.

Python interpreters are available for many operating systems, allowing Python code to run on a wide variety of systems using third-party tools, such as Py2exe or Pyinstaller. Python code can be packaged into stand-alone executable programs for some of the most popular operating systems, so Python-based software can be distributed to, and used on, those environments with no need to install a Python interpreter.

Python uses dynamic typing and a mix of reference counting and a cycle-detecting garbage collector for memory management. An important feature of Python is dynamic name resolution (late binding), which binds method and variable names during program execution.

The design of Python offers some support for functional programming in the Lisp tradition. The language has map(), reduce() and filter() functions; list comprehensions, dictionaries, and sets; and generator expressions. The standard library has two modules (itertools and functools) that implement functional tools borrowed from Haskell and Standard ML.

**Code:**

import csv

import cgi, cgitb

import os

import webbrowser

f=open('baseball.csv')

csv\_f=csv.reader(f)

d=0

yr=[]

rs=[]

game=[]

wt=[]

runsw=[]

seedw=[]

lt=[]

runsl=[]

seedl=[]

for row in csv\_f:

yr.append(row[0])

rs.append(row[1])

game.append(row[2])

wt.append(row[3])

runsw.append(row[4])

seedw.append(row[5])

lt.append(row[6])

runsl.append(row[7])

seedl.append(row[8])

d+=1

w=[]

dw=[]

l=[]

dl=[]

r=[]

dr=[]

a=0

aa=1

ab=1

b=0

c=0

m=0

n=0

var = input("Year for which you want analysis: ")

for i in range(d):

if(yr[i]==var):

w.append(wt[i])

l.append(lt[i])

r.append(rs[i]

dw=set (w)

a=len(dw)

dw=list(dw)

dl=set(l)

b=len(dl)

dl=list(dl)

dr=set(r)

c=len(dr)

dr=list(dr)

file=open('anal.txt','w')

if a>0:

file.write ("\n Number of teams which won in year " +var+ " are ")

file.write (str(a))

file.write ("\n Teams which won in year " +var+ " are ")

file.write ("\n"+str(dw))

file.write ("\n")

file.write ("\n")

file.write ("\n")

file.write ("\n------------------------------------------------------------------------------- ")

file.write ("\n")

file.write ("\n")

file.write ("\n Number of teams which lost in year " +var+ " are ")

file.write (str(b))

file.write ("\n Teams which lost in year " +var+ " are ")

file.write ("\n"+str(dl))

file.write ("\n")

file.write ("\n")

file.write ("\n")

file.write ("\n------------------------------------------------------------------------------- ")

file.write ("\n")

file.write ("\n")

file.write ("\n Number of regions in which match was played in year " +var+ " are ")

file.write (str(c))

file.write ("\n All the Regions in which match was played in year " +var+ " are ")

file.write ("\n"+str(dr))

file.write ("\n")

file.write ("\n")

file.write ("\n")

file.write ("\n------------------------------------------------------------------------------- ")

file.write ("\n")

file.write ("\n")

for i in range(d):

if yr[i]==var:

if int(runsw[aa])<int(runsw[i]):

aa=i

#file.write(aa)

file.write("\n Team which did max runs in year " +var+ " are ")

file.write(wt[aa])

file.write("\n runs scored is ")

file.write(runsw[aa])

file.write ("\n")

file.write ("\n")

file.write ("\n")

file.write ("\n------------------------------------------------------------------------------- ")

file.write ("\n")

file.write ("\n")

for i in range(d):

if yr[i]==var:

if int(runsl[aa])>int(runsl[i]):

ab=i

#file.write(aa)

file.write("\n Team which did least runs in year " +var+ " are ")

file.write(lt[ab])

file.write("\n runs scored is ")

file.write(runsl[ab])

else:

file.write("\n Dataset contains only years from 2003 to 2008")

file.close()

f = open('ssd.html','w')

message = """<html>

<head></head>

<body><p>"NCAA BASEBALL DATA ANALYSIS "</p></body>

<p></p>

THE ANALYSIS IS STORED IN THE TEXT FILE

<p></p>

<a href="anal.txt" target="\_explorer.exe">anal</a>

</html>"""

f.write(message)

f.close()

webbrowser.open("ssd.html")

**SAS Software:**

SAS is a software suite that can mine, alter, manage, and retrieve data from a variety of sources and perform statistical analysis on it.SAS provides a graphical point-and-click user interface for non-technical users and more advanced options through the SAS language. To use Statistical Analysis System, Data should be in a spreadsheet table format or SAS format.SAS programs have a DATA step, which retrieves and manipulates data, usually creating a SAS data set, and a PROC step, which analyzes the data.

Each step consists of a series of statements. The DATA step has executable statements that result in the software taking an action, and declarative statements that provide instructions to read a data set or alter the data's appearance.

The DATA step has two phases, compilation, and execution.

* In the compilation phase, declarative statements are processed and syntax errors are identified.
* Afterwards, the execution phase processes each executable statement sequentially. Data sets are organized into tables with rows called "observations" and columns called "variables". Additionally, each piece of data has a descriptor and a value.

The PROC step consists of PROC statements that call upon named procedures. Procedures perform analysis and reporting on data sets to produce statistics, analyses, and graphics. There are more than 300 procedures and each one contains a substantial body of programming and statistical work. PROC statements can also display results, sort data or perform other operations. SAS Macros are pieces of code or variables that are coded once and referenced to perform repetitive tasks.

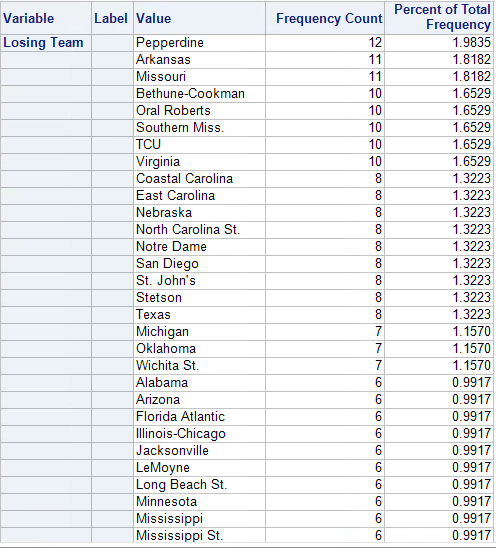
SAS data can be published in HTML, PDF, Excel, and other formats using the Output Delivery System. The SAS Enterprise Guide is SAS' point-and-click interface. It generates code to manipulate data or perform analysis automatically and does not require SAS programming experience to use.

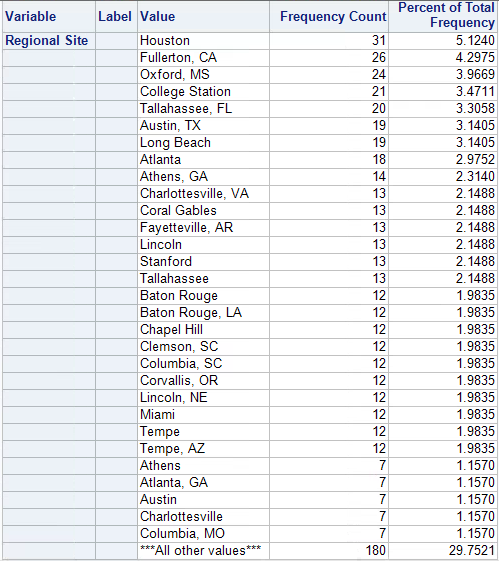
The software classifies the data set into the categorical and numerical based on the types of values present in the set. It outputs the set of categorical variables which contains the value corresponding to that variable, frequency count and also the percent of total frequency.

The categorical variables discovered in our data set are:

1. Losing Team
2. Regional Site
3. Winning Team





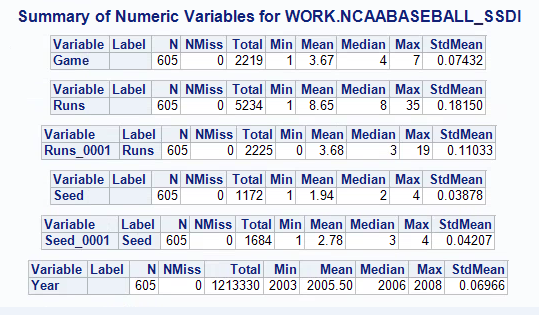




The numeric variables in the baseball data set are:

1. Game
2. Runs of the Winning Team
3. Seed of the Winning Team
4. Runs of the Losing Team(Runs\_0001)
5. Seed of the Losing Team(Seed\_0001)
6. Year

‘Year’ variable comes under Date(mm-dd-yyyy) variable. Since the year(yyyy) in the data set is not in the format of the date variable(mm-dd-yyyy), it is identified as numeric variable by SAS.



N - Count of the variables present in the data set

NMiss- Number of missing values corresponding to that variable

From SAS analysis, it is identified that there are no missing values in the data set.

**Tableau:**

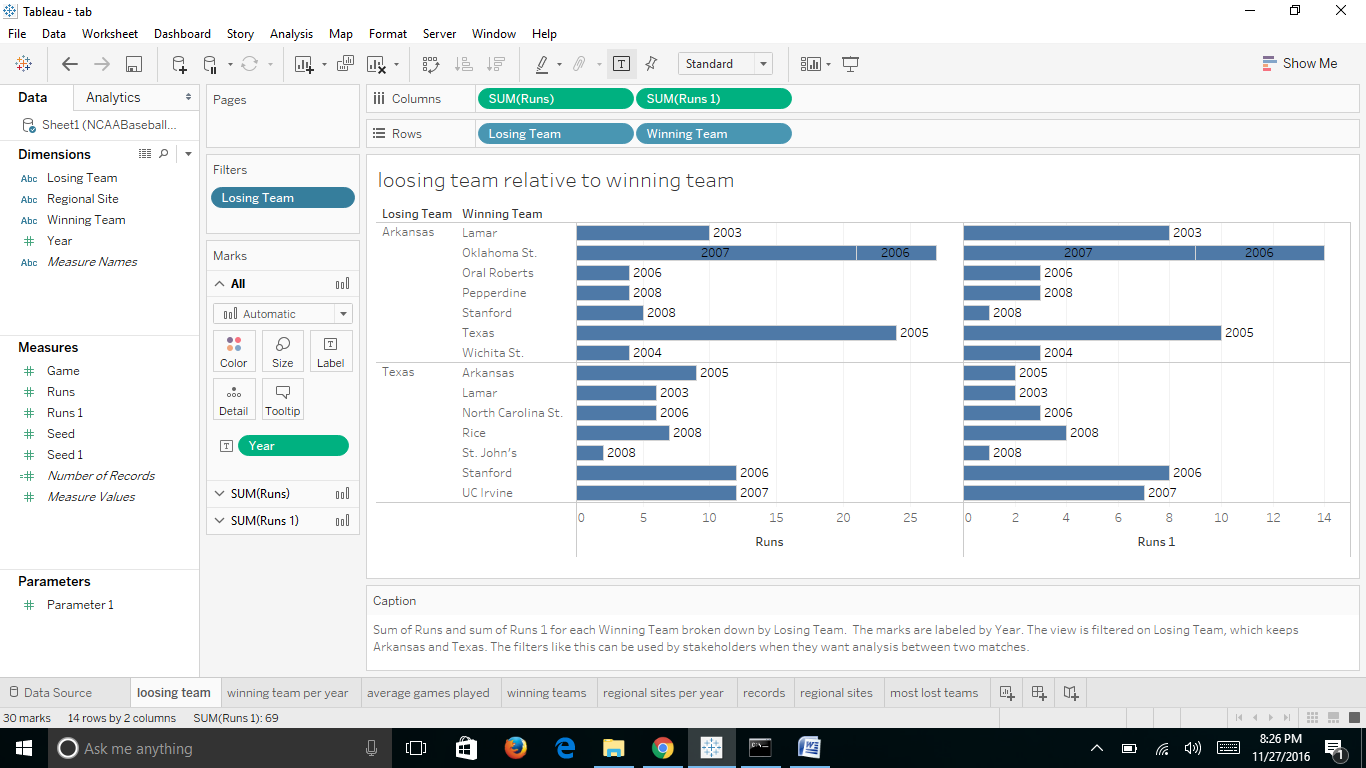
Tableau Desktop is a data visualization application that lets you analyze virtually any type of structured data and produce highly interactive, beautiful graphs, dashboards, stories and reports in just minutes. It generates a number of graph types. Tableau allows for instantaneous insight by transforming data into visually appealing, interactive visualizations called dashboards. This is achieved through the use of an easy to use drag-and-drop interface.

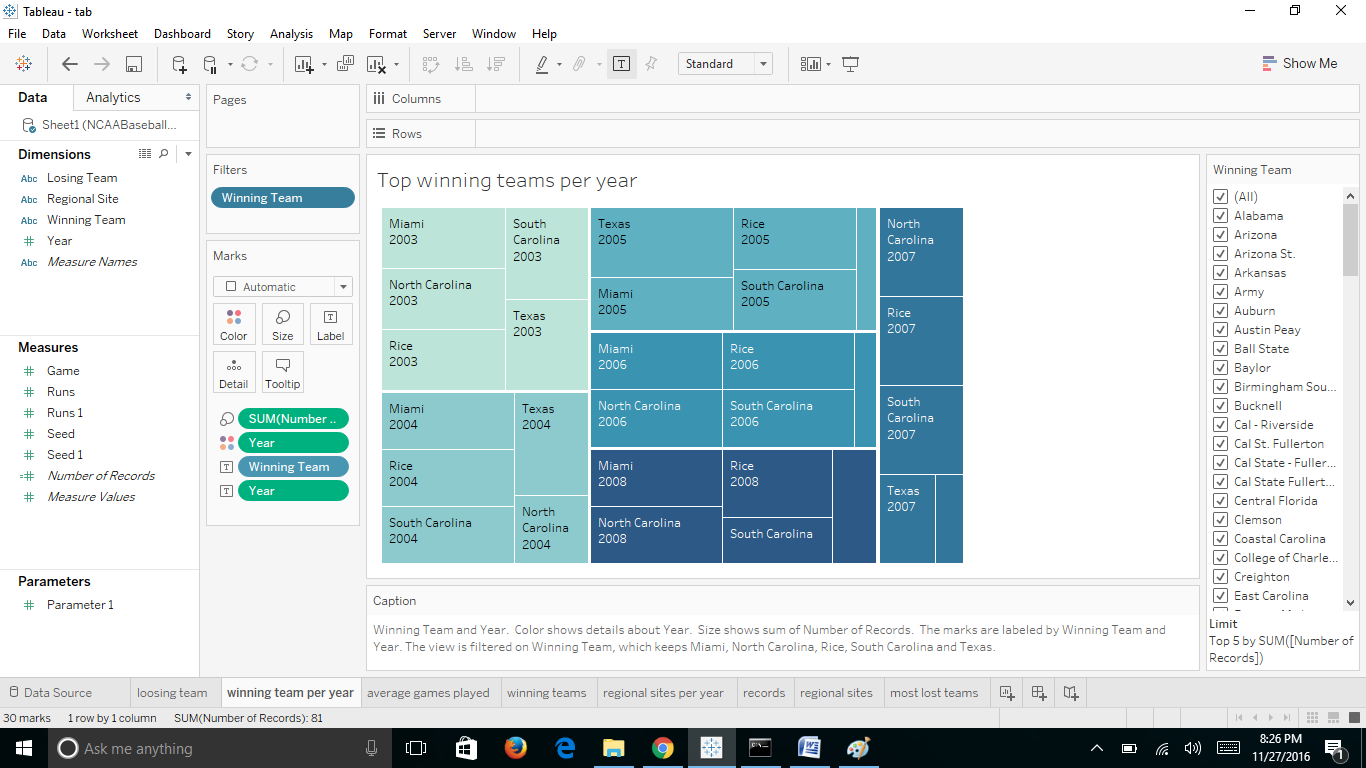
After a quick installation, you can connect to virtually any data source from spreadsheets to data warehouses and display information in multiple graphic perspectives. Designed to be easy to use, you’ll be working faster than ever before. People need effective views of data to understand results, discover relationships, find patterns, locate outliers, uncover structure, and summarize findings.

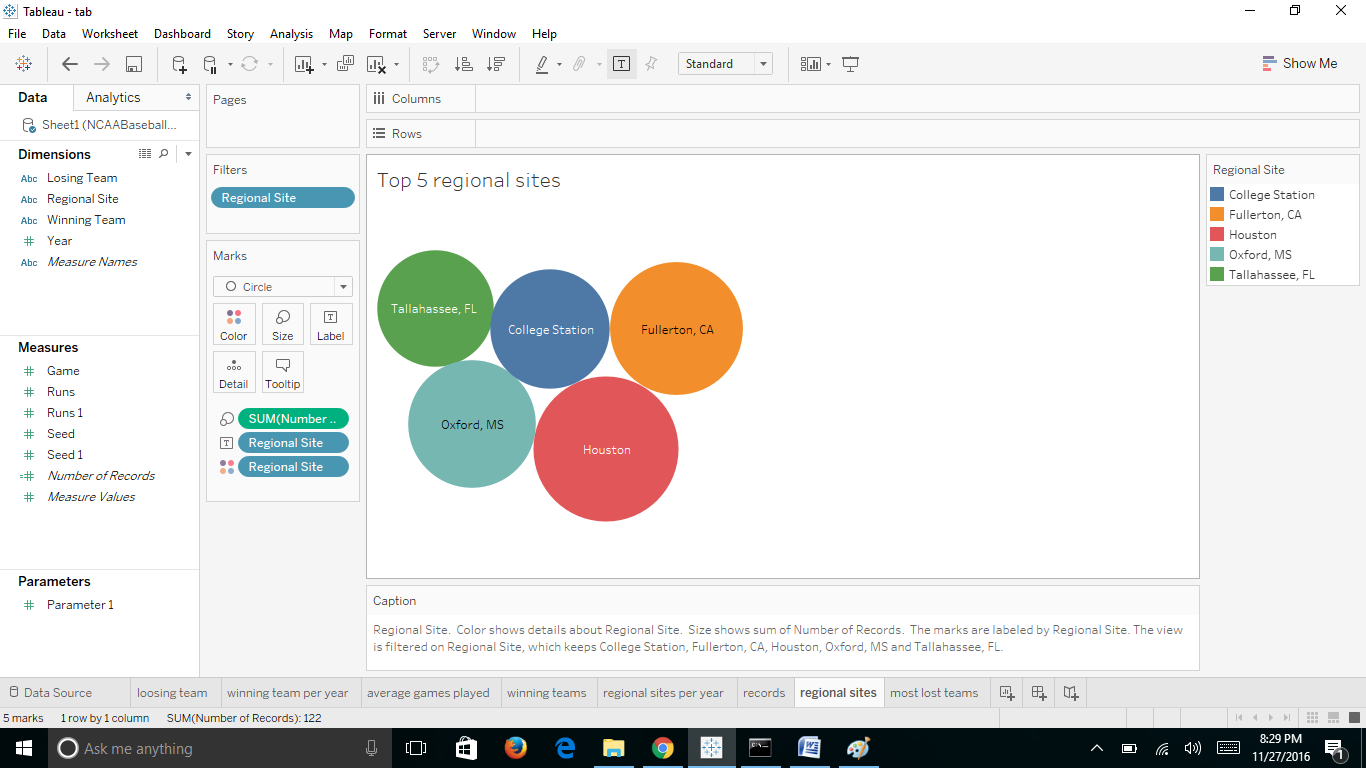
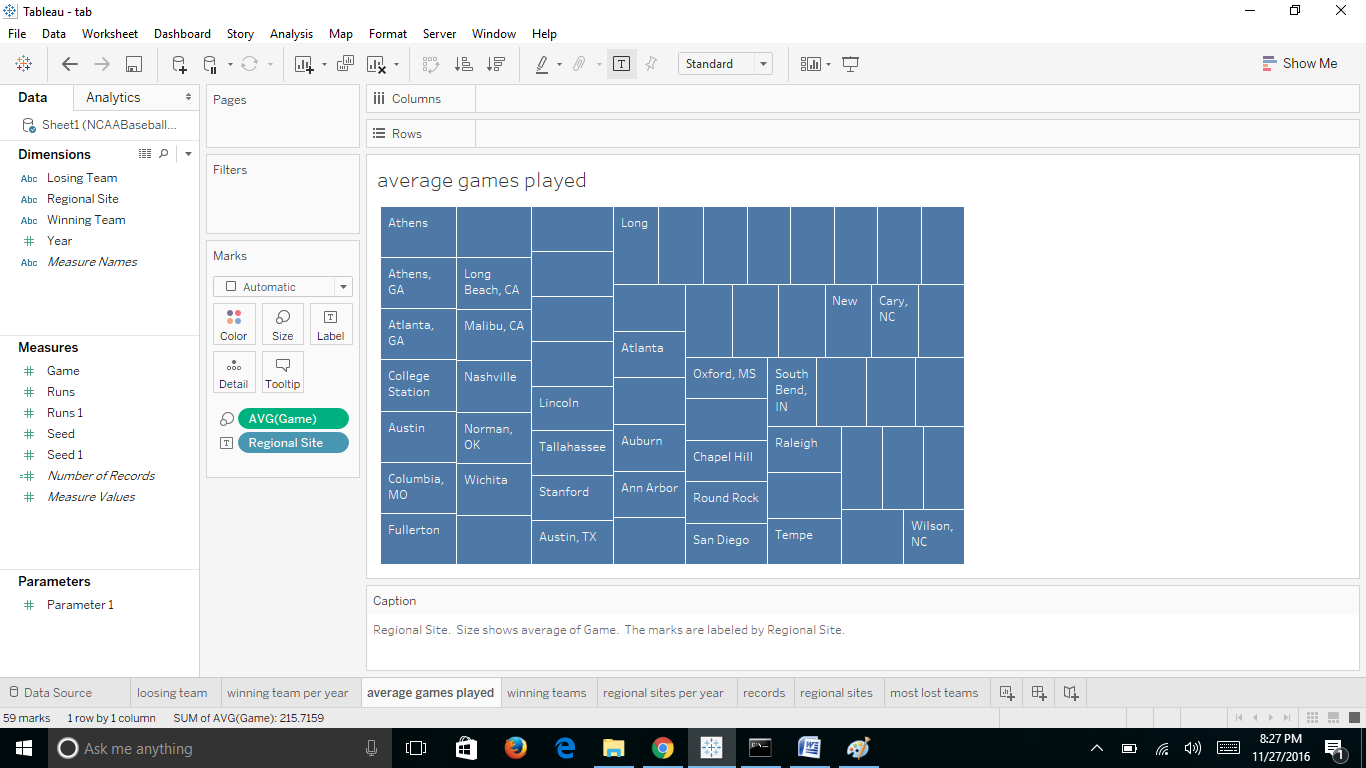
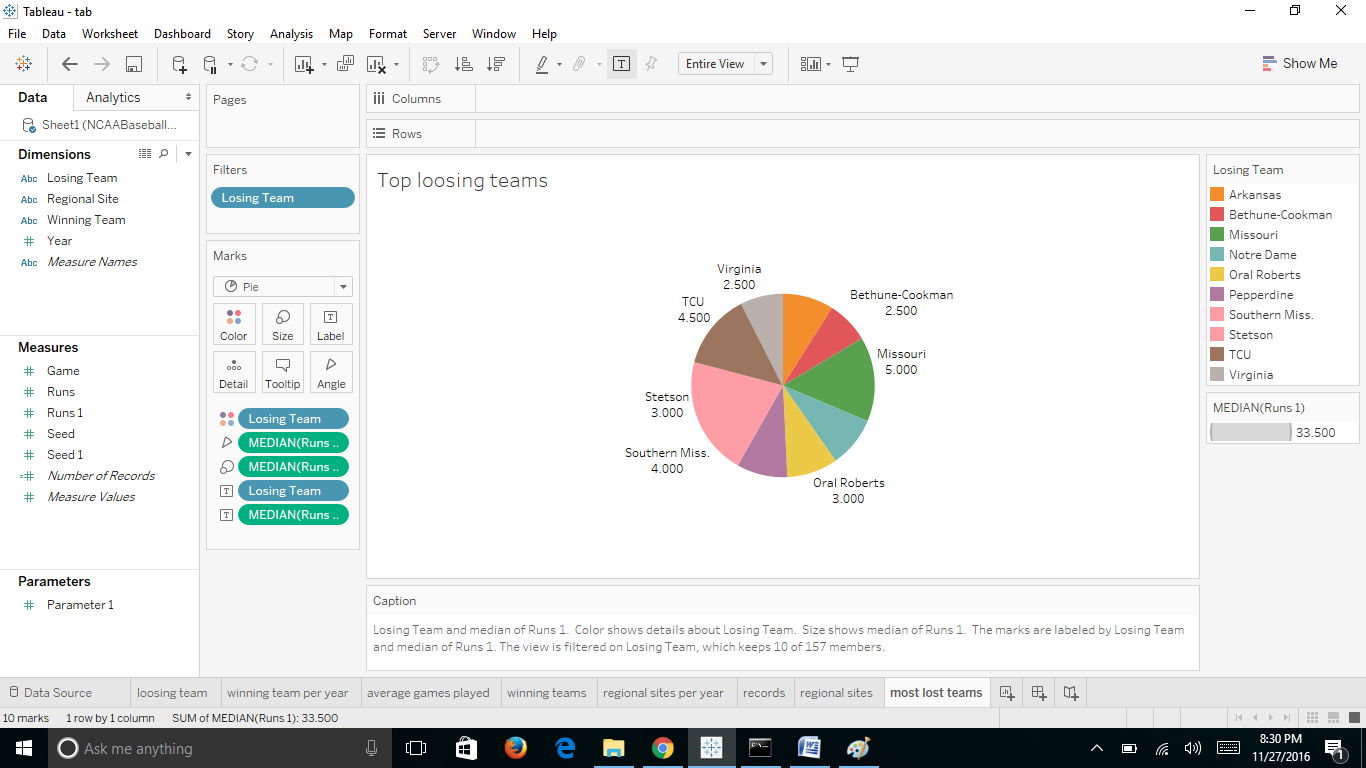
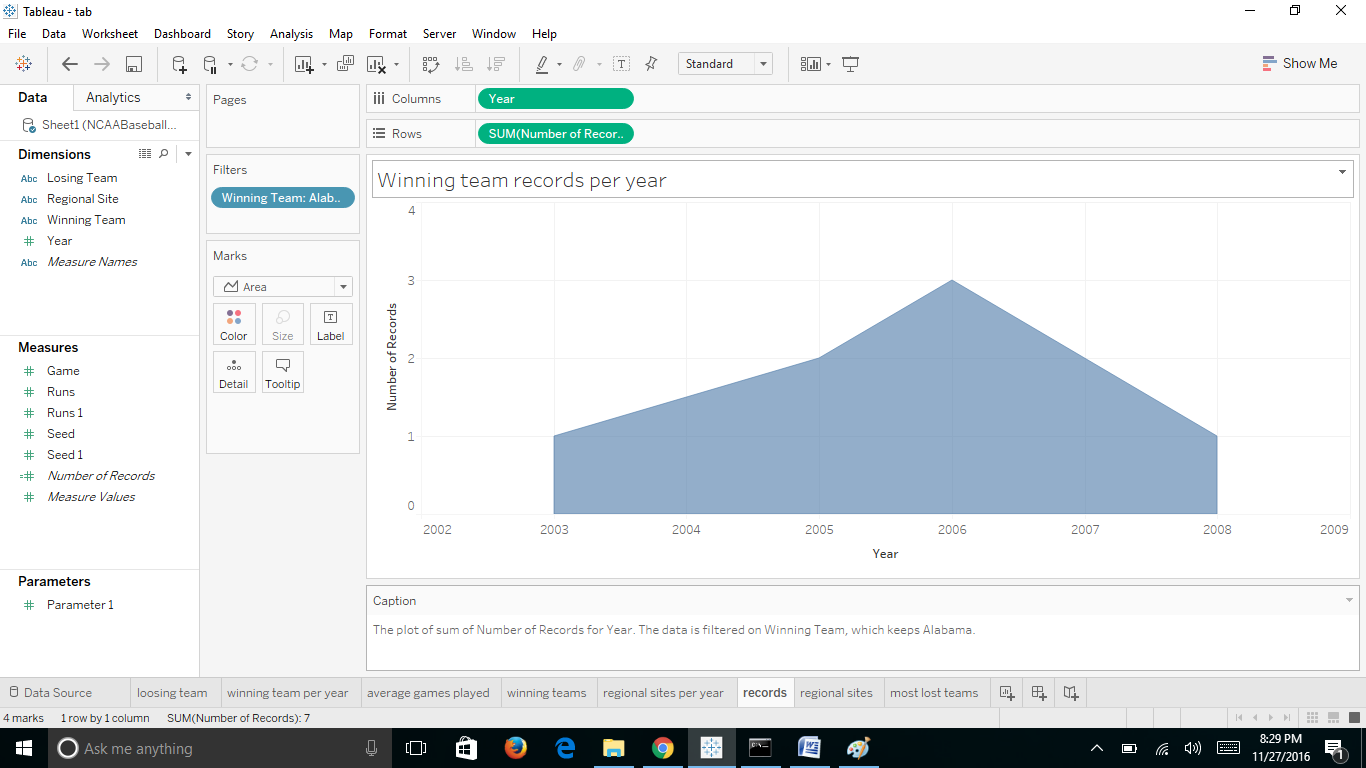
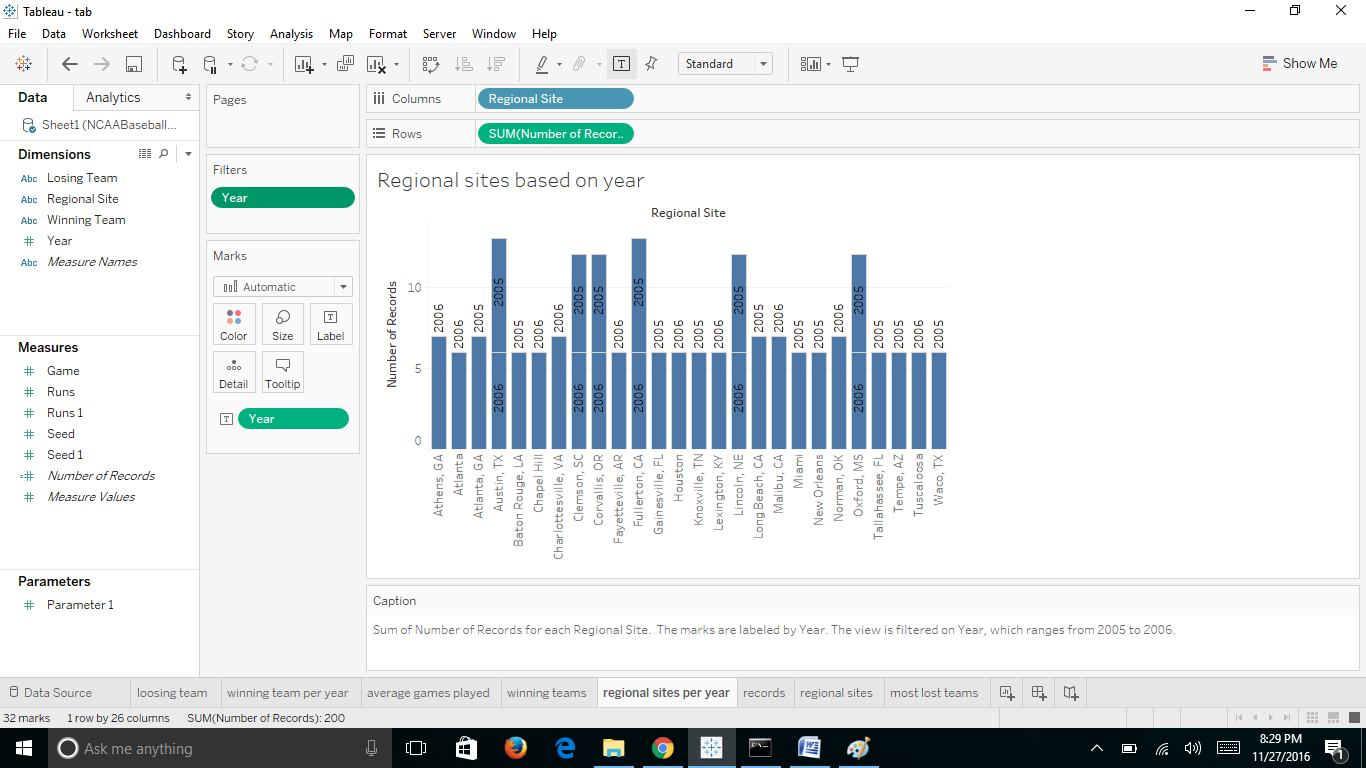
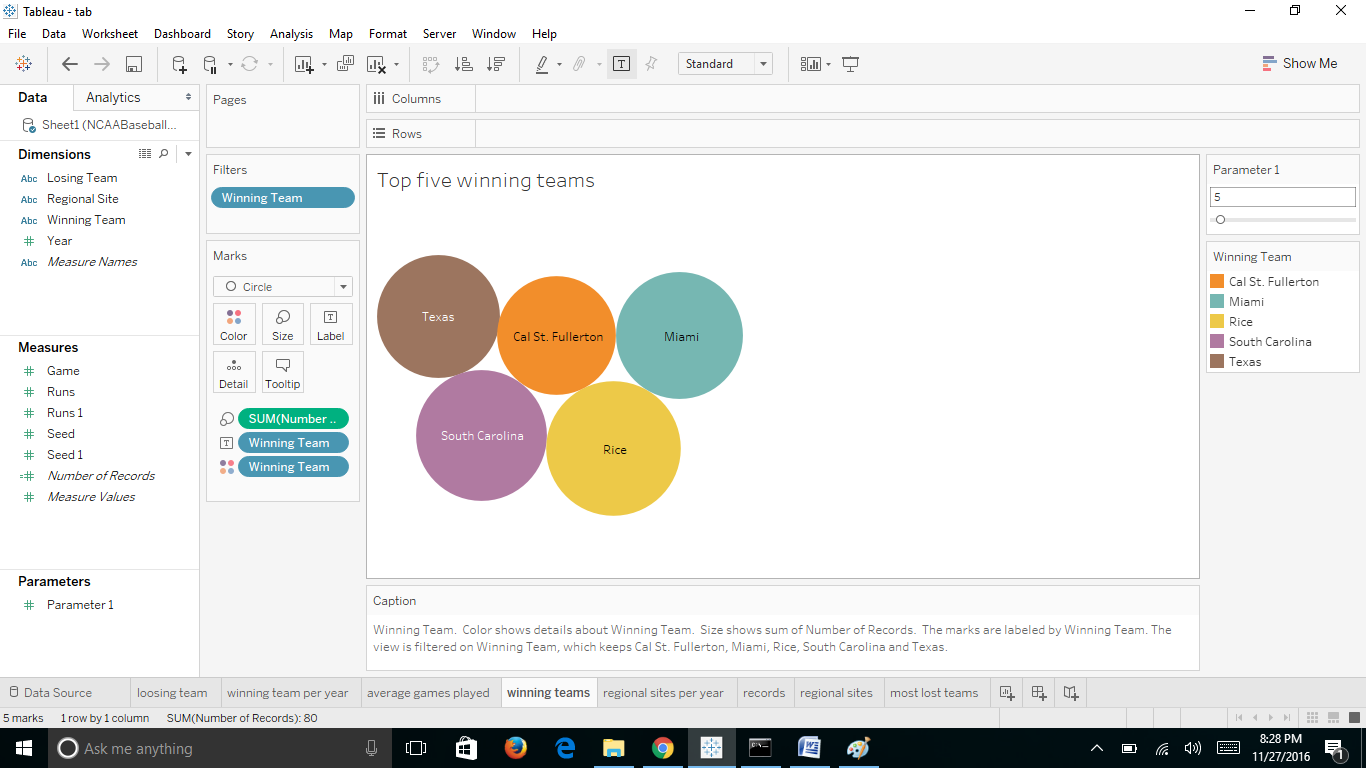
Tableau lets you ask rapid questions of your data by letting you iteratively create and modify live, interactive charts, reports and dashboards in minutes. These views are fundamentally more useful for analysis than those provided by pre-canned reports and traditional dashboards. Tableau gives you interactive visual tables, picture-perfect data displays, side-by-side comparisons, and graphic encodings using color, size and shape. Without any programming or training, users can see and understand data like they’ve never been able to before.

In our project, tableau is used to find the data visualization for NCAA baseball analysis. It generates graphs taking regional sites, winning teams, losing teams, runs as parameters, all graphs are drawn between these parameters. Based on these graphs, insights have been generated which helps the stakeholders in many ways. For example, the stakeholders can have sample analysis of the teams which won in the previous years and have a chance of winning in the coming year.

Worksheets of Tableau:







Programming languages: Python

Program: python program for displaying the analysis of the data based on the year

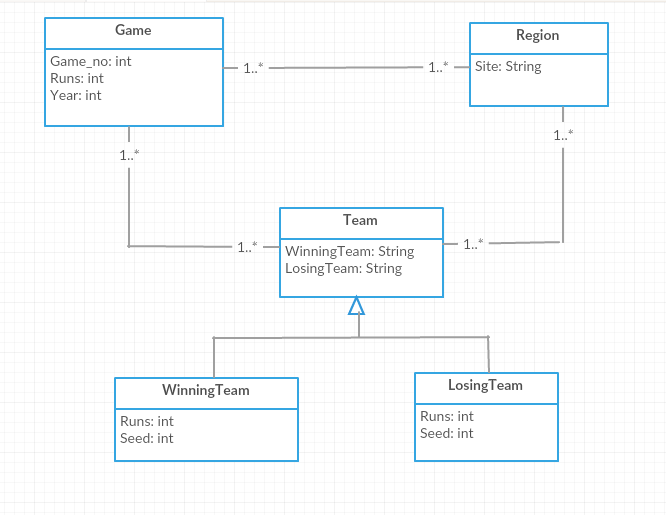
Tools and environments: SAS Enterprise guide and Tableau Desktop

Text files: “anal.txt” is used to store the analyzed data which is the output of the program.

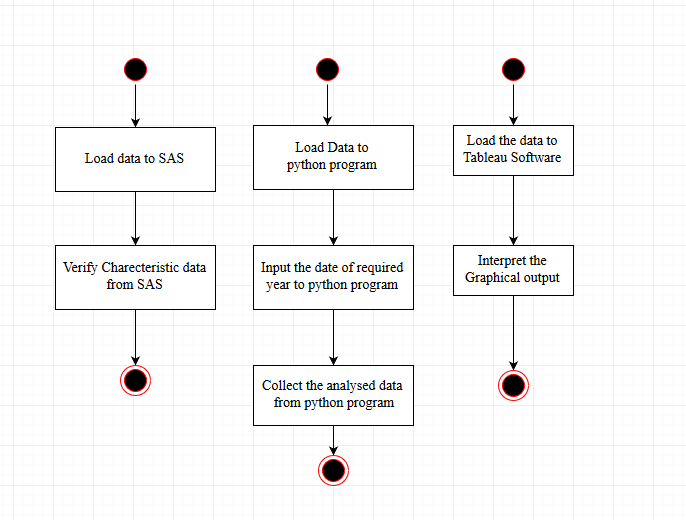
In addition to this, one more file i.e., NCAABaseball.xls or NCAABaseball.csv which is the dataset file used in the project.

**UML Diagrams:**

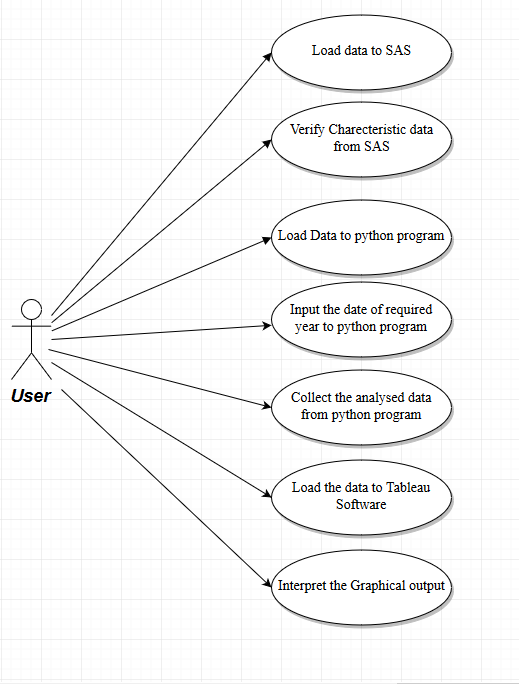
**Class Diagram:**



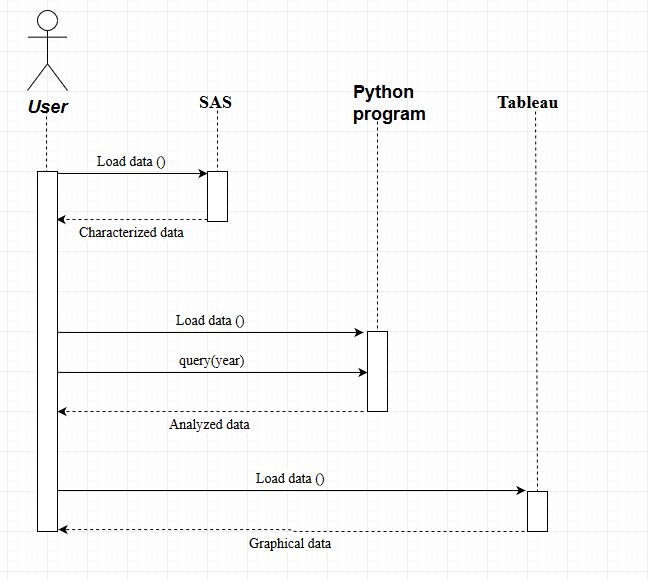
**Activity Diagram:**



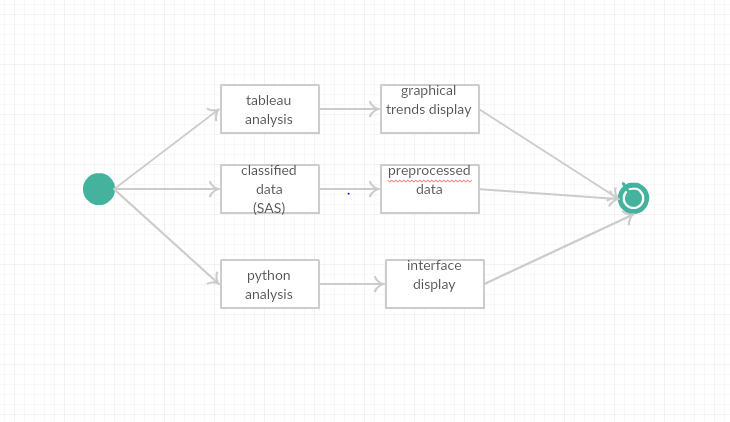
**Use Case Diagram:**



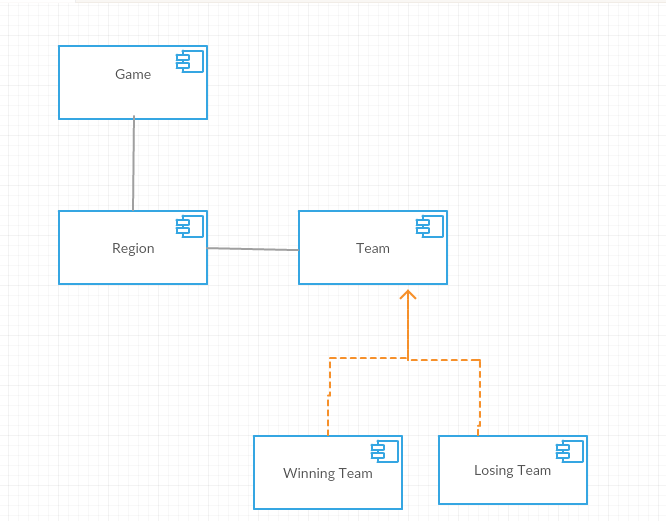
**Sequence Diagram:**



**State Diagram:**



**Component Diagram:**

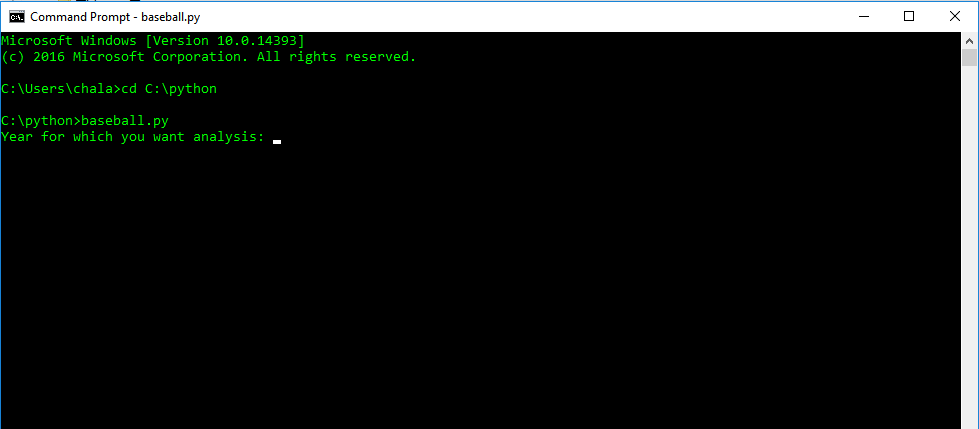
****

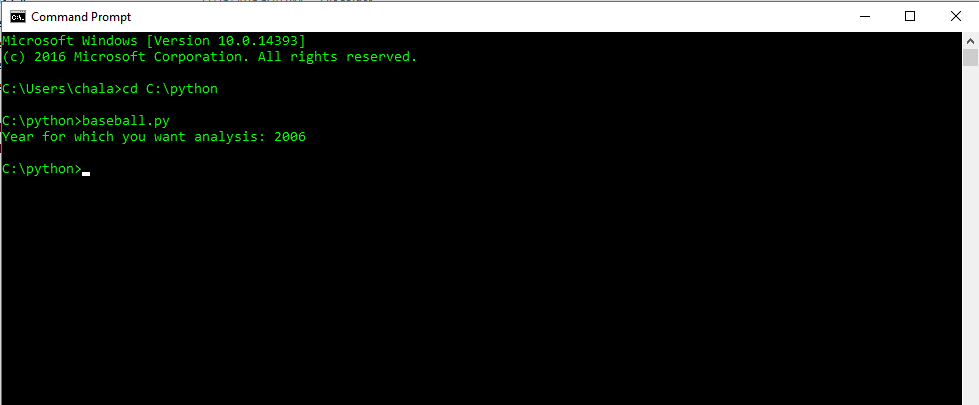
**Results:**

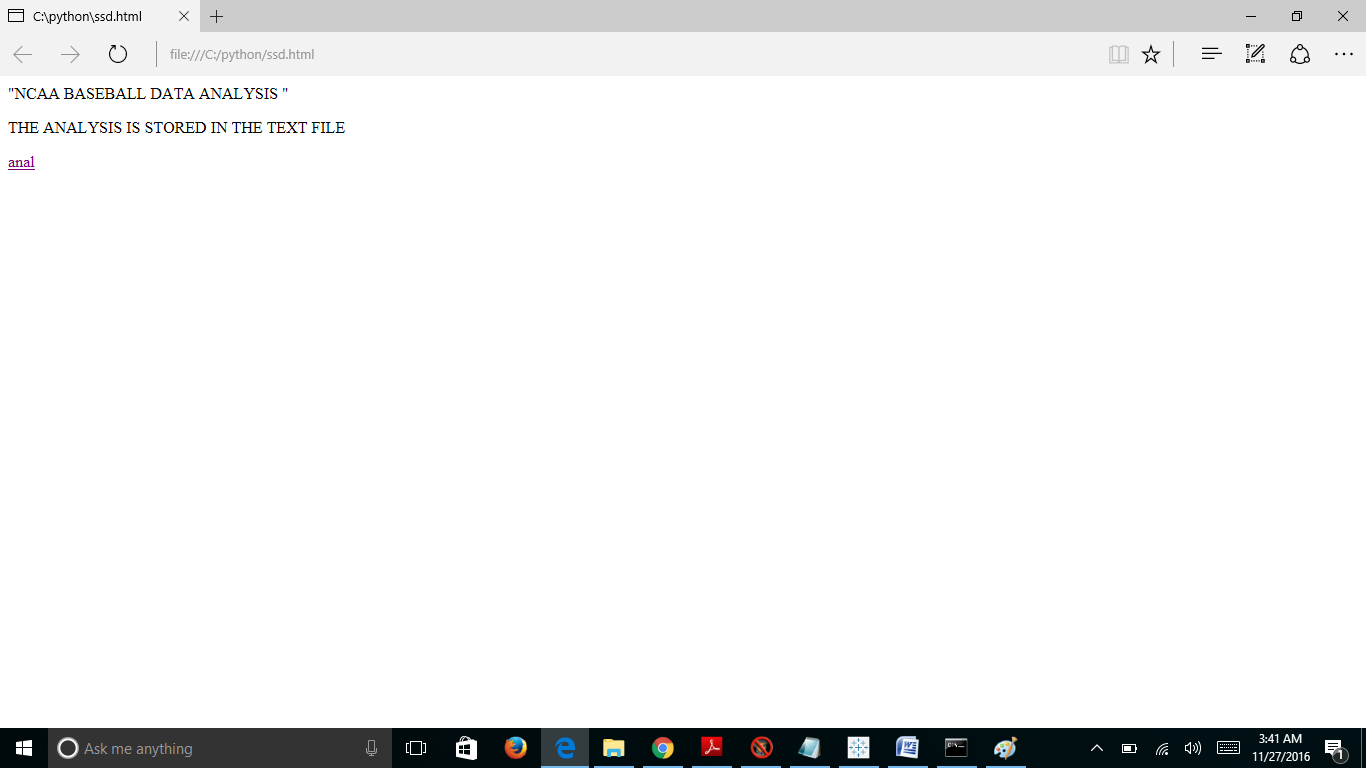
Based on the SAS Enterprise guide analysis, it is clear that there are no missing values in the data and the dataset is perfect except for the date attribute.

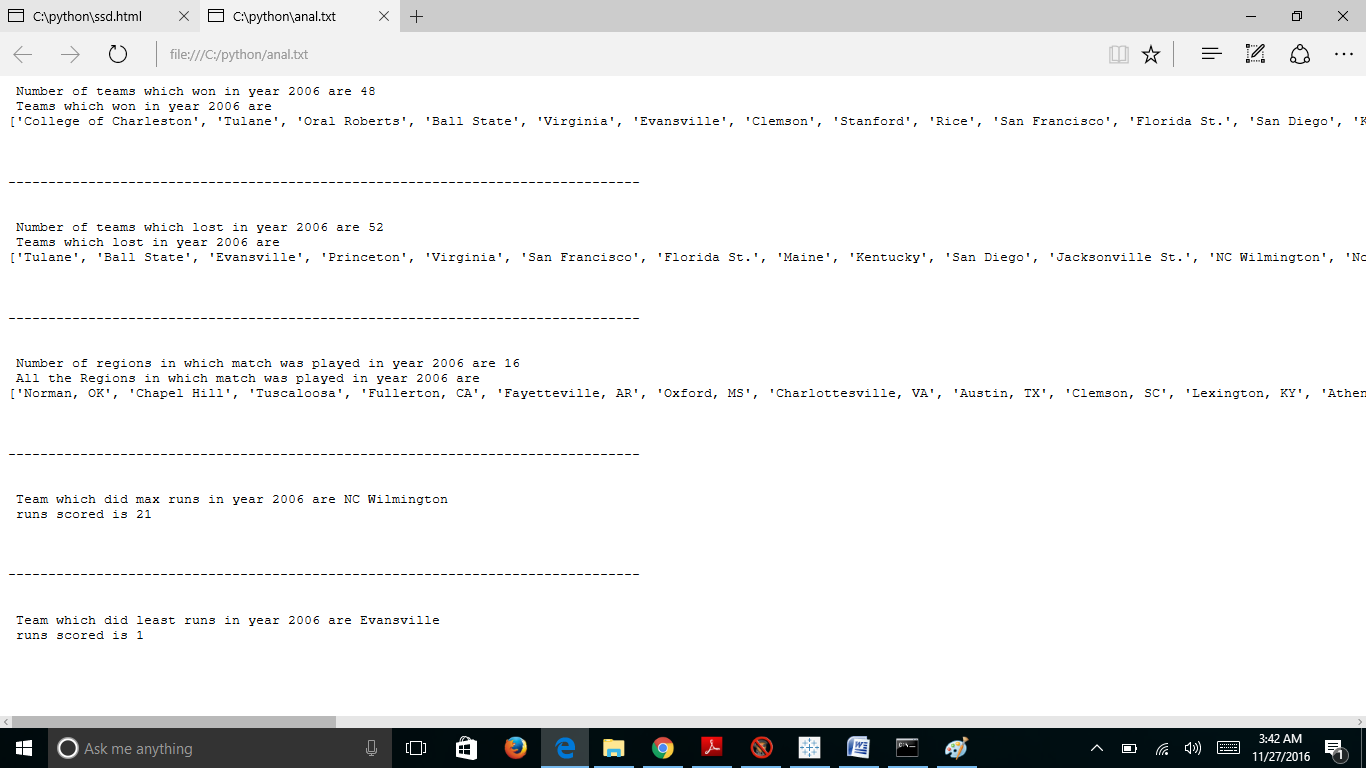
Tableau desktop gives clear graphs of regional sites in which games are most played, top winning teams, losing teams and their runs with respective to winning teams and the number of games played per year as shown in the Worksheets.

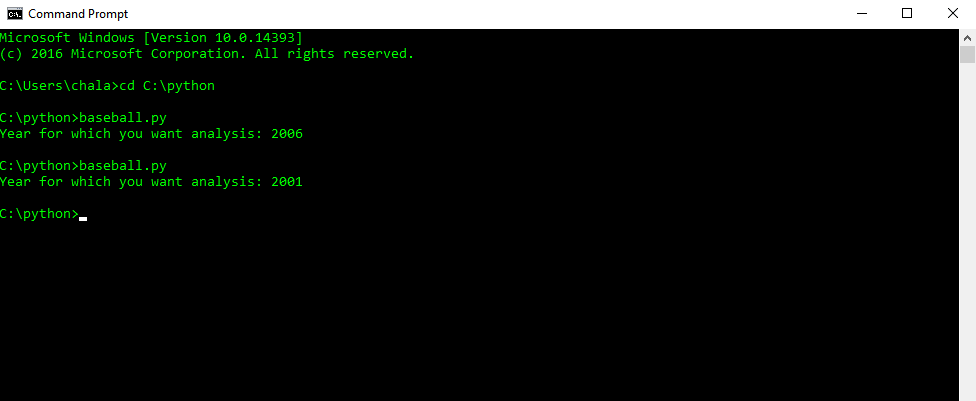
Python program displays the year-wise analysis of data based on the year which we select as shown below.

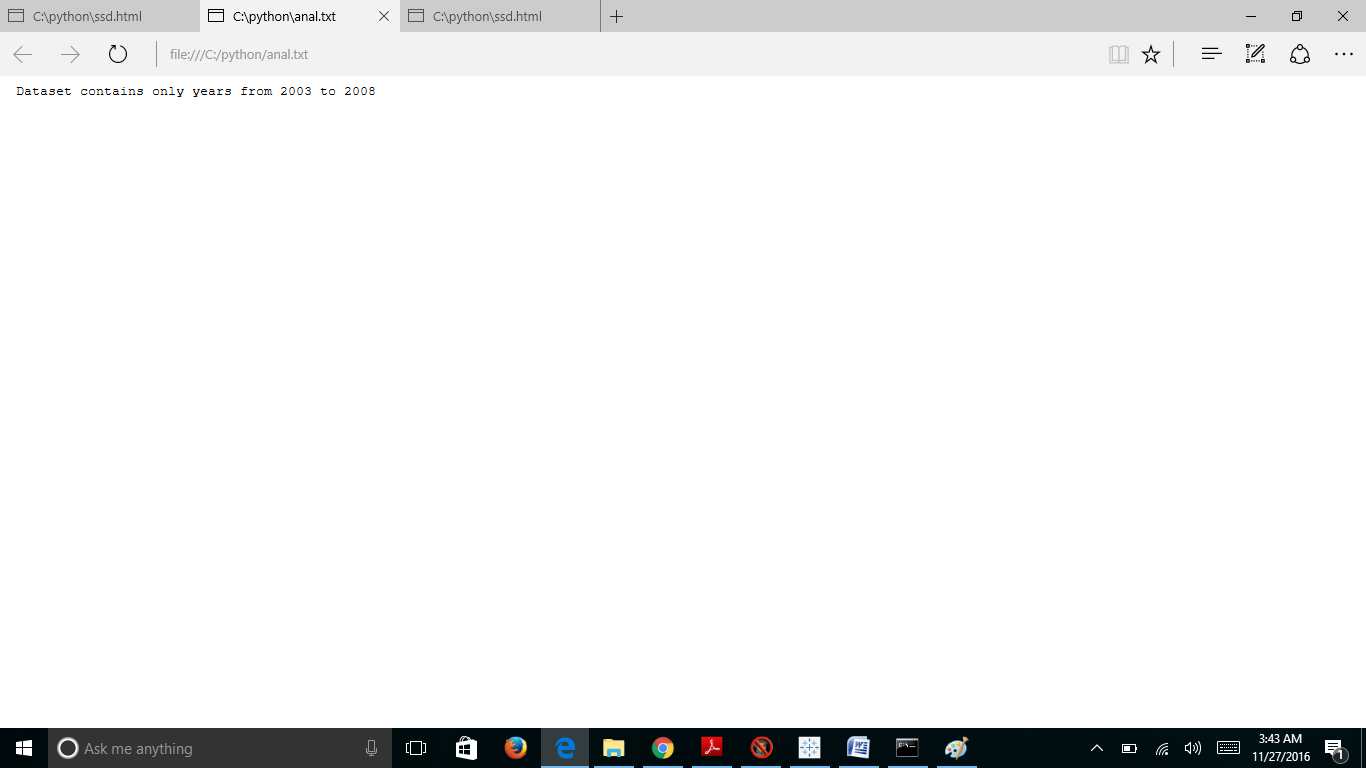






A 





Thus the program displays the analysis of dataset for any of the years from 2003 to 2008.

**Conclusions:**

Based on the tableau desktop visualizations, we can draw conclusions such as one of the top five winning teams such as Texas, Miami, Cal St. Fullerton, Rice and South Carolina may perform well in the coming years. This is based on a single visualization that is the top five winning teams’ visualization for the years 2003 to 2008. As we observed the data analysis of 2009, it is given that Texas performed very well defeating Boston and created a record of longest game.

Similar conclusions can be drawn from the rest of the visualizations which might be true. But these conclusions might not always be true which depends on several other factors. For example, Fresno state is the champions of the 2008 NCAA college world series but it was eliminated in the Irvine regional after losing to UC Irvine and San Diego. If we draw conclusion based on the 2008 that Fresco state may perform well then it would have been a wrong conclusion. Thus our visualizations might not always give good insights, as they might also depend on several other factors which are not included in the dataset.

In this NCAA baseball dataset, though we visualized data for 2003-2008, we might not get good results because in 2009 several new teams may add up in the competition or new coaches might be recruited which may change a losing team to a winning team in the next year. Although all the insights from these visualizations might not be true, these results help the stakeholders to get an overall view of the teams.

However, we can get even better visualizations if we consider more number of years rather than considering years only from 2003 to 2008.In addition the visualizations also depend on other factors which can be included in the dataset which can give a better conclusions.

**Reference:**

The conclusion about Fresno and Texas states is taken from the following reference:

[*https://en.wikipedia.org/wiki/2009\_NCAA\_Division\_I\_Baseball\_Tournament*](https://en.wikipedia.org/wiki/2009_NCAA_Division_I_Baseball_Tournament)