SAS and Open Source Playing Nicely Together

Jim Box
Principal Data Scientist, HLS R&D
SAS Institute
SAS Environments

9.4 vs Viya
SAS Platforms

SAS 9.4 – Where you are now
SAS Platforms

SAS Viya – Where we are now
SAS and R

A long-term relationship
• PROC IML (Interactive Matrix Language)
• Code structure:
  – call ExportDataSetToR("SAS libname.dataset","R data frame")
  – Submit / R;
  – All your R code
  – Endsubmit;
  – call ImportDataSetFromR("<SAS dataset to write>","<R data frame>")
  – Quit;
PROC IML;
call ExportDataSetToR("Sashelp.Heart", "df" );

submit / R;
  summary(df)
endsubmit;
QUIT;
## SAS 9.4 and R

**PROC IML**

<table>
<thead>
<tr>
<th>Code</th>
<th>Log</th>
<th>Results</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Status</strong></td>
<td><strong>DeathCause</strong></td>
<td><strong>AgeCHDdiag</strong></td>
</tr>
<tr>
<td>Alive:3218</td>
<td>Cancer</td>
<td>: 539</td>
</tr>
<tr>
<td>Dead:1991</td>
<td>Cerebral Vascular Disease: 378</td>
<td>1st Qu.: 57.0</td>
</tr>
<tr>
<td>Coronary Heart Disease</td>
<td>: 605</td>
<td>Median: 63.0</td>
</tr>
<tr>
<td>Other</td>
<td>: 357</td>
<td>Mean : 63.3</td>
</tr>
<tr>
<td>Unknown</td>
<td>: 112</td>
<td>3rd Qu.: 70.0</td>
</tr>
<tr>
<td>NA's</td>
<td>: 3218</td>
<td>Max. : 90.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NA's : 3760</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>AgeAtStart</th>
<th>Height</th>
<th>Weight</th>
<th>Diastolic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min. : 28.00</td>
<td>Min. : 51.50</td>
<td>Min. : 67.0</td>
<td>Min. : 50.00</td>
</tr>
<tr>
<td>1st Qu.: 37.00</td>
<td>1st Qu.: 62.25</td>
<td>1st Qu.: 132.0</td>
<td>1st Qu.: 76.00</td>
</tr>
<tr>
<td>Median : 43.00</td>
<td>Median : 64.50</td>
<td>Median : 150.0</td>
<td>Median : 84.00</td>
</tr>
<tr>
<td>Mean : 44.07</td>
<td>Mean : 64.81</td>
<td>Mean : 153.1</td>
<td>Mean : 85.36</td>
</tr>
<tr>
<td>3rd Qu.: 51.00</td>
<td>3rd Qu.: 67.50</td>
<td>3rd Qu.: 172.0</td>
<td>3rd Qu.: 92.00</td>
</tr>
<tr>
<td>Max. : 62.00</td>
<td>Max. : 76.50</td>
<td>Max. : 300.0</td>
<td>Max. : 160.00</td>
</tr>
<tr>
<td>NA's : 6</td>
<td>NA's : 6</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Systolic</th>
<th>HRW</th>
<th>Smoking</th>
<th>AgeAtDeath</th>
<th>Cholesterol</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min. : 82.0</td>
<td>Min. : 67</td>
<td>Min. : 0.000</td>
<td>Min. : 35.00</td>
<td>Min. : 96.0</td>
</tr>
<tr>
<td>1st Qu.: 120.0</td>
<td>1st Qu.: 106</td>
<td>1st Qu.: 0.000</td>
<td>1st Qu.: 63.00</td>
<td>1st Qu.: 196.0</td>
</tr>
<tr>
<td>Median : 132.0</td>
<td>Median : 118</td>
<td>Median : 1.000</td>
<td>Median : 71.00</td>
<td>Median : 223.0</td>
</tr>
<tr>
<td>Mean : 136.9</td>
<td>Mean : 120</td>
<td>Mean : 9.367</td>
<td>Mean : 70.54</td>
<td>Mean : 227.4</td>
</tr>
<tr>
<td>3rd Qu.: 148.0</td>
<td>3rd Qu.: 131</td>
<td>3rd Qu.: 20.000</td>
<td>3rd Qu.: 79.00</td>
<td>3rd Qu.: 255.0</td>
</tr>
<tr>
<td>Max. : 300.0</td>
<td>Max. : 268</td>
<td>Max. : 60.000</td>
<td>Max. : 93.00</td>
<td>Max. : 568.0</td>
</tr>
<tr>
<td>NA's : 6</td>
<td>NA's : 36</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chol_Status</th>
<th>BP_Status</th>
<th>Weight_Status</th>
<th>Smoking_Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Borderline:1861</td>
<td>High : 2267</td>
<td>Normal : 1472</td>
<td>Heavy (16-25) : 1046</td>
</tr>
<tr>
<td>Desirable :1405</td>
<td>Normal : 2143</td>
<td>Overweight : 3550</td>
<td>Light (1-5) : 579</td>
</tr>
<tr>
<td>High : 1791</td>
<td>Optimal : 799</td>
<td>Underweight : 181</td>
<td>Moderate (6-15) : 576</td>
</tr>
<tr>
<td>NA's : 152</td>
<td>NA's : 6</td>
<td>Non-smoker : 2401</td>
<td>Very Heavy (&gt; 25) : 471</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NA's : 36</td>
</tr>
</tbody>
</table>
SAS 9.4 and R

PROC IML

• Text output will be written to the Results window
• Graphical output needs to be pointed to a specific location – it will not (currently) show up in the SAS interface

```
119  p <- ggplot(teams, aes(x=Outcome, y=Score, color=Outcome)) + geom_boxplot()
120
121  png(file="/data/compute-landingzone/Projects/Open Source/boxplot.png",
122  width=600, height=350)
123  p
dev.off()
```
SAS 9.4 and R

Reading SAS Data into R

- R library: sas7bdat
  - Command: `df=read.sas7bdat(“heart.sas7bdat”)`

- R library: haven:
  - Command: `df= read_sas(“heart.sas7bdat”)`

- Haven also has a `write_sas` command to output datasets when done
SAS 9.4 and R

Reading SAS Data into R

```r
library(ggplot2)
library(sas7bdat)

df=read.sas7bdat("heart.sas7bdat")

plt = ggplot(df, aes(x=Status, y=AgeAtStart, fill=Status)) +
  geom_boxplot(outlier.colour="red", outlier.shape=8,
              outlier.size=4)

plt + stat_summary(fun=mean, geom='point', shape=23, size=4)
```
SAS Viya and R

PROC IML

• Works exactly the same as SAS 9.4. (runs in the compute engine)
library(swat)
Sys.setenv(CAS_CLIENT_SSL_CA_LIST='/etc/pki/tls/certs/ca.crt')
username <- rstudioapi::askForPassword("username")
password <- rstudioapi::askForPassword("password")

currentCASlib='PUBLIC'

#list available tables in caslib
sites <- defCasTable(session,caslib=currentCASlib,"StudySites")
head(sites)
dim(sites)
### SAS Viya and R

**R Studio connection & SWAT**

```r
> head(sites)

<table>
<thead>
<tr>
<th>Replication</th>
<th>Site</th>
<th>Start_Flag</th>
<th>StartUp</th>
<th>Cost</th>
<th>StartDelay</th>
<th>CountryDelay</th>
<th>FirstPatient</th>
<th>LastPatient</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Chenango Memorial Hospital</td>
<td>1</td>
<td>22886 5500</td>
<td>60</td>
<td>32</td>
<td>22891</td>
<td>23298</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Derry Medical Center</td>
<td>1</td>
<td>22890 4500</td>
<td>64</td>
<td>36</td>
<td>22894</td>
<td>23300</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Lassen General Hospital</td>
<td>1</td>
<td>22891 6000</td>
<td>65</td>
<td>37</td>
<td>22895</td>
<td>23295</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Sacred Heart Hospital</td>
<td>1</td>
<td>22894 5000</td>
<td>68</td>
<td>40</td>
<td>22898</td>
<td>23298</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Shasta Regional Medical Center</td>
<td>1</td>
<td>22879 5500</td>
<td>53</td>
<td>25</td>
<td>22883</td>
<td>23297</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>St Eligus Hospital</td>
<td>1</td>
<td>22897 7500</td>
<td>71</td>
<td>43</td>
<td>22902</td>
<td>23295</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>N_Screened</th>
<th>N_Enroll</th>
<th>P_Fail</th>
<th>N_Complete</th>
<th>P_Comp</th>
<th>StudyVisits</th>
<th>ScreenCost</th>
<th>VisitCost</th>
<th>TotalCost</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>93</td>
<td>84</td>
<td>0.09677419</td>
<td>68</td>
<td>0.8095238</td>
<td>236</td>
<td>62775</td>
<td>206516</td>
</tr>
<tr>
<td>2</td>
<td>110</td>
<td>97</td>
<td>0.11818182</td>
<td>86</td>
<td>0.8865979</td>
<td>277</td>
<td>82500</td>
<td>275200</td>
</tr>
<tr>
<td>3</td>
<td>93</td>
<td>80</td>
<td>0.13978495</td>
<td>64</td>
<td>0.8000000</td>
<td>223</td>
<td>60450</td>
<td>187200</td>
</tr>
<tr>
<td>4</td>
<td>118</td>
<td>108</td>
<td>0.08474576</td>
<td>90</td>
<td>0.8333333</td>
<td>309</td>
<td>85550</td>
<td>293580</td>
</tr>
<tr>
<td>5</td>
<td>100</td>
<td>81</td>
<td>0.19000000</td>
<td>64</td>
<td>0.7901235</td>
<td>230</td>
<td>70000</td>
<td>201600</td>
</tr>
<tr>
<td>6</td>
<td>81</td>
<td>69</td>
<td>0.14814815</td>
<td>51</td>
<td>0.7391304</td>
<td>194</td>
<td>53460</td>
<td>151470</td>
</tr>
</tbody>
</table>
```
SAS Viya and R

R Studio connection & SWAT

```r
site51 <- cas.dataStep.runCode(session,
    code="
data Site51;
set PUBLIC.StudySites;
if Replication = ' 51';
run;"
)
results <- cas.table.fetch(session,
    table=list(name="site51")
)
results
```
## SAS Viya and R

### R Studio connection & SWAT

<table>
<thead>
<tr>
<th><em>Index</em></th>
<th>Replication</th>
<th>Site</th>
<th>Start_Flag</th>
<th>StartUp</th>
<th>Cost</th>
<th>StartDelay</th>
<th>CountryDelay</th>
<th>FirstPatient</th>
<th>LastPatient</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>Chenango Memorial Hospital</td>
<td>1</td>
<td>22903</td>
<td>5500</td>
<td>77</td>
<td>37</td>
<td>22908</td>
<td>23320</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>Derry Medical Center</td>
<td>1</td>
<td>22895</td>
<td>4500</td>
<td>69</td>
<td>29</td>
<td>22899</td>
<td>23322</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>Lassen General Hospital</td>
<td>1</td>
<td>22902</td>
<td>6000</td>
<td>76</td>
<td>36</td>
<td>22906</td>
<td>23318</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>Sacred Heart Hospital</td>
<td>1</td>
<td>22895</td>
<td>5000</td>
<td>69</td>
<td>29</td>
<td>22899</td>
<td>23322</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>Shasta Regional Medical Center</td>
<td>1</td>
<td>22899</td>
<td>5500</td>
<td>73</td>
<td>33</td>
<td>22903</td>
<td>23322</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>St Eligus Hospital</td>
<td>1</td>
<td>22907</td>
<td>7500</td>
<td>81</td>
<td>41</td>
<td>22912</td>
<td>23323</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>Tower Medical Group</td>
<td>1</td>
<td>22898</td>
<td>6550</td>
<td>72</td>
<td>32</td>
<td>22904</td>
<td>23319</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>Twin Pines Medical Center</td>
<td>1</td>
<td>22899</td>
<td>7000</td>
<td>73</td>
<td>33</td>
<td>22905</td>
<td>23320</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>Western Regional Hospital</td>
<td>1</td>
<td>22897</td>
<td>8000</td>
<td>71</td>
<td>31</td>
<td>22901</td>
<td>23325</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>Wexler Medical Center</td>
<td>1</td>
<td>22897</td>
<td>4250</td>
<td>71</td>
<td>31</td>
<td>22901</td>
<td>23325</td>
</tr>
</tbody>
</table>
SAS and Python

A big focus for SAS
SAS Viya and Python

Write Python Programs in SAS Studio

```python
1. print("Hello World!")
```
SAS Viya and Python

PROC Python

• Run Python code within a SAS session
• Call most SAS functions within Python statements
• Submit SAS code from Python
• Move data between SAS datasets and Pandas dataframes
• Transfer values between SAS macro variables and Python variables
SAS Viya and Python

PROC Python – Run SAS in Python

```sas
%let language = 'python';

proc python;
submit;
lang = SAS.symget('language')
ver = 3.8
SAS.submit("data work.test; language={}; version={}; run;
".format(lang,ver))

var3 = SAS.sasfnc("upcase","hello world")
print( var3)

py_var = 'Inside python'
SAS.symput('macrovar', py_var)

endsubmit;
run;
%
put &macrovar;

proc print data=test;
run;
```
SAS Viya and Python

PROC Python – Run SAS in Python

```plaintext
95  endsSubmit;
96  run:
97  data work.test; language='python'; version=3.8; run;
   NOTE: The data set WORK.TEST has 1 observations and 2 variables.
   NOTE: DATA statement used (Total process time):
         real time   0.00 seconds
         cpu time   0.01 seconds

>>> HELLO WORLD

>>> 

   NOTE: PROCEDURE PYTHON used (Total process time):
         real time   0.00 seconds
         cpu time   0.01 seconds

99  %put &=macrovar;
    MACROVAR=Inside python
100
101
```
SAS Viya and Python

PROC Python – Run SAS in Python

<table>
<thead>
<tr>
<th>Obs</th>
<th>language</th>
<th>version</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>python</td>
<td>3.8</td>
</tr>
</tbody>
</table>
SAS Viya and Python

PROC Python – run Python in SAS

```python
proc python;
submit;

import pandas as pd
URL = "https://en.wikipedia.org/wiki/List_of_Super_Bowl_champions"

tables = pd.read_html(URL, attrs = {'class' : 'wikitable sortable'})
sb = tables[0]
sb.head()

ds = SAS.df2sd(sb, 'work.sb')

endsubmit;
run;
```

Use Pandas to read in a table from wiki page

Output to SAS dataset
SAS and Open Source

Use Cases

• Sometimes a Python method or an R library can make your work easier

• PROC PYTHON and PROC IML make it easy to leverage Python and R, but still be able to use SAS when it makes sense

• SAS Viya makes it easy to mix and match programming languages to solve problems in the most efficient way possible
# SAS and Open Source

## Use Case: Pull Data from Wikipedia

<table>
<thead>
<tr>
<th>Game</th>
<th>Date/Season</th>
<th>Winning team</th>
<th>Score</th>
<th>Losing team</th>
<th>Venue</th>
<th>City</th>
<th>Attendance</th>
<th>Referee</th>
<th>Ref</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>January 15, 1967 (1966 AFL/1966 NFL)</td>
<td>Green Bay Packers&lt;sup&gt;A&lt;/sup&gt; (1, 1–0)</td>
<td>35–10</td>
<td>Kansas City Chiefs&lt;sup&gt;A&lt;/sup&gt; (1, 0–1)</td>
<td>Los Angeles Memorial Coliseum</td>
<td>Los Angeles, California&lt;sup&gt;1&lt;/sup&gt;</td>
<td>61,946</td>
<td>Norm Schachter</td>
<td>[7][8]</td>
</tr>
<tr>
<td>II</td>
<td>January 14, 1968 (1967 AFL/1967 NFL)</td>
<td>Green Bay Packers&lt;sup&gt;A&lt;/sup&gt; (2, 2–0)</td>
<td>33–14</td>
<td>Oakland Raiders&lt;sup&gt;B&lt;/sup&gt; (1, 0–1)</td>
<td>Miami Orange Bowl</td>
<td>Miami, Florida&lt;sup&gt;3&lt;/sup&gt;</td>
<td>75,546</td>
<td>Jack Vest</td>
<td>[9][8]</td>
</tr>
<tr>
<td>III</td>
<td>January 12, 1969 (1968 AFL/1968 NFL)</td>
<td>New York Jets&lt;sup&gt;A&lt;/sup&gt; (1, 1–0)</td>
<td>16–7</td>
<td>Baltimore Colts&lt;sup&gt;N&lt;/sup&gt; (1, 0–1)</td>
<td>Miami Orange Bowl (2)</td>
<td>Miami, Florida&lt;sup&gt;2&lt;/sup&gt;</td>
<td>75,389</td>
<td>Tom Bell</td>
<td>[10][8]</td>
</tr>
<tr>
<td>IV</td>
<td>January 11, 1970 (1969 AFL/1969 NFL)</td>
<td>Kansas City Chiefs&lt;sup&gt;A&lt;/sup&gt; (2, 1–0)</td>
<td>23–7</td>
<td>Minnesota Vikings&lt;sup&gt;N&lt;/sup&gt; (1, 0–1)</td>
<td>Tulane Stadium</td>
<td>New Orleans, Louisiana</td>
<td>80,562</td>
<td>John McDonough</td>
<td>[11][8]</td>
</tr>
<tr>
<td>V</td>
<td>January 17, 1971 (1970)</td>
<td>Baltimore Colts&lt;sup&gt;A&lt;/sup&gt; (2, 1–0)</td>
<td>16–13</td>
<td>Dallas Cowboys&lt;sup&gt;N&lt;/sup&gt; (1, 0–1)</td>
<td>Miami Orange Bowl (3)</td>
<td>Miami, Florida&lt;sup&gt;3&lt;/sup&gt;</td>
<td>79,204</td>
<td>Norm Schachter</td>
<td>[12][8]</td>
</tr>
<tr>
<td>VI</td>
<td>January 16, 1972 (1971)</td>
<td>Dallas Cowboys&lt;sup&gt;N&lt;/sup&gt; (2, 1–0)</td>
<td>24–3</td>
<td>Miami Dolphins&lt;sup&gt;A&lt;/sup&gt; (1, 0–1)</td>
<td>Tulane Stadium (2)</td>
<td>New Orleans, Louisiana (2)</td>
<td>81,023</td>
<td>Jim Tunney</td>
<td>[13][8]</td>
</tr>
<tr>
<td>VII</td>
<td>January 14, 1973 (1972)</td>
<td>Miami Dolphins&lt;sup&gt;A&lt;/sup&gt; (2, 1–0)</td>
<td>14–7</td>
<td>Washington Redskins&lt;sup&gt;N&lt;/sup&gt; (1, 0–1)</td>
<td>Los Angeles Memorial Coliseum (2)</td>
<td>Los Angeles, California&lt;sup&gt;2&lt;/sup&gt;</td>
<td>90,182</td>
<td>Tom Bell</td>
<td>[14][8]</td>
</tr>
</tbody>
</table>

SAS and Open Source

Use Case: Python grabs data

```python
proc python;
submit;

import pandas as pd
URL = "https://en.wikipedia.org/wiki/List_of_Super_Bowl_champions"

tables = pd.read_html(URL, attrs = {'class': 'wikitable sortable'})
sb = tables[0]
sb.head()

ds = SAS.df2sd(sb, 'work.sb')
endsubmit;
run;
```
data SB2;
set SB;
  SB = _N_;
  Season = SB + 1965;
  if Attendance in ('TBD', 'Attendance') then delete;
  fans = input(Attendance, 8.);

  Winner = substr('Winning team'n, 1, index('Winning team'n, '(')-2);
  Loser = substr('Losing team'n, 1, index('Losing team'n, '(')-2);
  WS = input(substr(Score, 1, 2.), 2.);
  points = trim(left(scan (score, 1, ', ')));
  LS = input(substr(points, 6, 2), 2.);

  OT = index(Score, "OT")>0;
  paren = index(City, '(');
  bracket = index(City, '[');
  if paren then locale = substr(City, 1, paren-1);
    else if bracket then locale = substr(City, 1, bracket-1);
    else locale = trim(left(City));

  City1 = scan(locale, 1, ', ');
  State = scan(locale, 2, ', ');
run;
SAS and Open Source

Use Case: SAS Reports Output

Proc SQL;

Select SB as Superbowl "SuperBowl Number"
,Season "Season"
,City1 as City "SB City"
,State "SB State"
,fans as Attendance "Attendance" format = comma9.
,Winner "Winning Team"
,Loser "Losing Team"
,points as Score "Score"
,WS "Winning Score"
,LS "Losing Score"
,OT "Overtime"
from SB2;
quit.
## SAS and Open Source

### Use Case: SAS Reports Output

<table>
<thead>
<tr>
<th>SuperBowl Number</th>
<th>Season</th>
<th>SB City</th>
<th>SB State</th>
<th>Attendance</th>
<th>Winning Team</th>
<th>Losing Team</th>
<th>Score</th>
<th>Winning Score</th>
<th>Losing Score</th>
<th>Overtime</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1966</td>
<td>Los Angeles</td>
<td>California</td>
<td>61,945</td>
<td>Green Bay Packers</td>
<td>Kansas City Chiefs</td>
<td>35–10</td>
<td>35</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1967</td>
<td>Miami</td>
<td>Florida</td>
<td>75,546</td>
<td>Green Bay Packers</td>
<td>Oakland Raiders</td>
<td>33–14</td>
<td>33</td>
<td>14</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1968</td>
<td>Miami</td>
<td>Florida</td>
<td>75,389</td>
<td>New York Jets</td>
<td>Baltimore Colts</td>
<td>16–7</td>
<td>16</td>
<td>7</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>1969</td>
<td>New Orleans</td>
<td>Louisiana</td>
<td>80,562</td>
<td>Kansas City Chiefs</td>
<td>Minnesota Vikings</td>
<td>23–7</td>
<td>23</td>
<td>7</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>1970</td>
<td>Miami</td>
<td>Florida</td>
<td>79,204</td>
<td>Baltimore Colts</td>
<td>Dallas Cowboys</td>
<td>16–13</td>
<td>16</td>
<td>13</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>1971</td>
<td>New Orleans</td>
<td>Louisiana</td>
<td>81,023</td>
<td>Dallas Cowboys</td>
<td>Miami Dolphins</td>
<td>24–3</td>
<td>24</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>1972</td>
<td>Los Angeles</td>
<td>California</td>
<td>90,182</td>
<td>Miami Dolphins</td>
<td>Washington Redskins</td>
<td>14–7</td>
<td>14</td>
<td>7</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>1973</td>
<td>Houston</td>
<td>Texas</td>
<td>71,882</td>
<td>Miami Dolphins</td>
<td>Minnesota Vikings</td>
<td>24–7</td>
<td>24</td>
<td>7</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>1974</td>
<td>New Orleans</td>
<td>Louisiana</td>
<td>80,997</td>
<td>Pittsburgh Steelers</td>
<td>Minnesota Vikings</td>
<td>16–6</td>
<td>16</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>1975</td>
<td>Miami</td>
<td>Florida</td>
<td>80,187</td>
<td>Pittsburgh Steelers</td>
<td>Dallas Cowboys</td>
<td>21–17</td>
<td>21</td>
<td>17</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>1976</td>
<td>Pasadena</td>
<td>California</td>
<td>103,438</td>
<td>Oakland Raiders</td>
<td>Minnesota Vikings</td>
<td>32–14</td>
<td>32</td>
<td>14</td>
<td>0</td>
</tr>
<tr>
<td>12</td>
<td>1977</td>
<td>New Orleans</td>
<td>Louisiana</td>
<td>76,400</td>
<td>Dallas Cowboys</td>
<td>Denver Broncos</td>
<td>27–10</td>
<td>27</td>
<td>10</td>
<td>0</td>
</tr>
</tbody>
</table>
Conclusion

SAS & Open Source

• SAS has been open source friendly for years with R in PROC IML
• Python integration is a key goal
• SAS Viya makes it easy to mix and match programming languages to solve problems in the most efficient way possible
• There are several other integration points that are more for modelers and data scientists
  – Saspy in Python
  – Open Source nodes in Viya Modelling pipelines
Thanks

Jim Box
Principal Data Scientist, Life Sciences at SAS
Durham, North Carolina, United States - Contact info

Samiul Haque
1st
Transforming data into intelligence through cross-disciplinary collaboration | Machine Learning | Data Science | Analytics
Cary, North Carolina, United States - Contact info

sas.com