STAT 440 – Homework 10

Students are encouraged to work together on homework. However, sharing or copying any part of the homework is an infraction of the University's rules on Academic Integrity.

Final submissions must be uploaded to our Compass 2g site on the Homework page. No email, hardcopy, or late submissions will be accepted.

Getting the program file ready

a. Create a folder on the hard drive with the following pathname – C:\440\hw10. Save all data files accompanying this assignment in that folder. If you cannot create the folder because you are working on a university computer and don't have permission, create the …\440\hw10 folder elsewhere.
b. Assign the library reference **hw10** to the folder 'C:\440\hw10'. Use this library as your permanent library for this assignment. If you could not create the folder, assign the library reference **hw10** to your …\440\hw10 folder.
   Note: If you are using a folder other than 'C:\440\hw10', you must change any pathname references in your program file to 'C:\440\hw10' before submitting your homework.

Submitting your work to Compass 2g

You are to submit two (and only two) files for your homework submission.

1. Your SAS program file which should be saved as **HW*n_YourNetID*.sas**. For example, my file for the HW1 assignment would be HW1_dunger.sas. All program statements and code should be included in one program file.

2. Your Report including all relevant output to address the exercises. For this homework, use ODS to send your results to a Rich Text Format (RTF) file called **HW*n_YourNetID*.rtf**. For example, my file for HW1 would be HW1_dunger.rtf. Only include your final set of output. Do not include output for every execution of your SAS program.

You have an unlimited number of submissions, but only the last one will be viewed and graded. Homework submissions must always come as a pair of files, as described above.

1. You have several options for investing money, and you want to write a program that will calculate the most lucrative choice. You plan to make a yearly deposit and the investment fund will return the given annual interest rate compounded according to the given frequency. (This is meant to be very similar to our examples and exercises from the notes.)

Option	Yearly Deposit	Annual Interest Rate	Compound Frequency	Times per Year
A	$1000	8.00%	Yearly	1
B	$1700	4.00%	Quarterly	4
C	$2000	3.50%	Quarterly	4
D	$2200	2.25%	Monthly	12
E	$2500	1.25%	Monthly	12
F	$2700	1.00%	Weekly	52

   a. Write a single DATA step to determine which is the most profitable by completing the following tasks.
      - Enter this data directly into the SAS program file using the DATALINES statement. Do not create a new raw data file to read from.
      - Calculate the final balance of each option after 25 years. Do not use a single compound interest formula; instead, show that this can be done using DO loops.
      - Save the results in a temporary SAS data set called **mostmoney_*NetID***.
      - The SAS data set should contain 6 observations with the five variables shown above and a sixth variable labeled "Total after 25 years."

   b. Print the data portion of the SAS data set **mostmoney_*NetID***. (Include results in the HW Report. After opening the Report file, highlight the highest total after 25 years in green and the lowest in orange.)

   c. Write a single DATA step to determine which one gets you to $30,000 the fastest.
      - You can essentially make a copy of your DATA step in part a, and edit it as needed.
      - Save the results in a temporary SAS data set called **save30_*NetID***.
      - The SAS data set should contain 6 observations with only three variables: Option, a new variable labeled "Years until $30K," and a new variable called Amount that gives the value of the account at the end of the year in which the $30,000 threshhold is crossed.

   d. Print the data portion of the SAS data set **save30_*NetID***. (Include results in the HW Report. After opening the Report file, highlight the lowest number of years (i.e., the fastest) in green and the highest (i.e., the slowest) in orange.)

2. You will be working with the raw data set **chicago_avg_temps 12-15.txt** which contains the 1260 average daily temperatures in Chicago from Sunday, January 1, 2012 to Saturday, June 13, 2015. The data was obtained from the University of Dayton archives: http://academic.udayton.edu/kissock/http/Weather/citylistUS.htm.

   a. Write a program to read in the data from **chicago_avg_temps 12-15.txt** into a permanent SAS data set called **chicago_*NetID*** .
      - The SAS data set should contain one observation for each value in the raw data set and the following variables:
        - Temp -- the value from the raw data set.
        - Weekday -- the day of the week (e.g., Sunday, Monday,…)
        - Date -- the actual date of the observed temperature (e.g. 01/01/2012).
      - Apply a format to the Date variable so values are written as January 1, 2012. That is, month, day, and year but no weekday name.

   b. Print the data portion of **chicago_*NetID*** from December 2014. (Include results in the HW Report.)

   You will now be working with the raw data set **hourly_temps.txt** which contains the daily temperatures for each hour of the day on two of the hottest days in Chicago in 2013. The 48 temperature values are listed out in several lines. The first 24 values represent temperatures from Hour 1 to Hour 24 on Day 1, and the next 24 values represent temperatures from Hour 1 to Hour 24 on Day 2.

   c. Write a program to read in the data from hourly_temps.txt into a permanent SAS data set with 48 observations called **hotdays_*NetID***.
      - The SAS data set should contain one observation for each value in the raw data set and the following variables:
        - Day -- a value of either 1 or 2.
        - Hour -- a value from 1 to 24.
        - Temp -- the value from the raw data set.

   d. Print a table containing the number of observations, median, mean, and standard deviation for temperatures for each of the two days individually. (Include results in the HW Report.)