COVER PAGE

Table of Contents

Disclaimer

This user manual was developed by the Cortex team of SAS employees.

It is intended solely for the use of beta testing and may not be provided to any other person or entity without the express written consent of SAS Institute.

While every effort was made to ensure accuracy and completeness, neither SAS nor the report authors are able to warrant the degree of accuracy or completeness of this manual.

This user guideline was prepared on a best effort basis and is only intended to assist participants to play the game during the beta testing period. The reader should not rely solely on the manual's content to play the game.

SAS is committing to offer more reference links of further learning resources for each section in the future.

What is Cortex?

SAS and ERPsim Lab at HEC Montreal partnered to develop Cortex, an analytics simulation game, which is designed to help participants apply and consolidate different data analytics concepts in realistic, business settings. Cortex currently offers a business scenario called the Fundraising Scenario.

In the Fundraising Scenario, participants will target potential donors to maximize the donations to the fundraising campaign. This simulation game provides pedagogical flexibility and is adaptable to students in both undergrad and graduate programs.

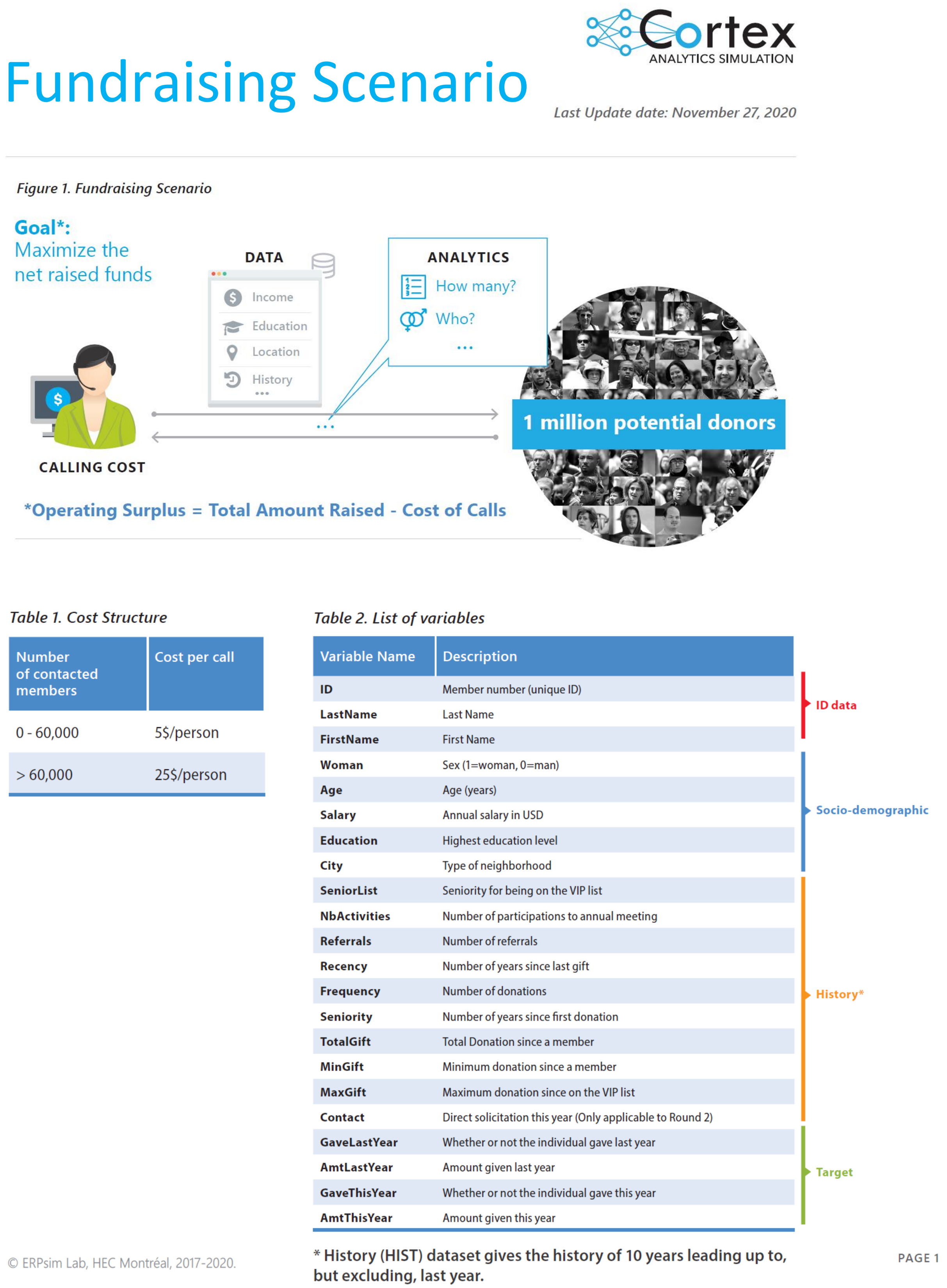
The Fundraising Scenario provides a complete experience that can be used in any teaching context with participants of varying levels of statistical knowledge and competency. Its hands-on approach helps the development of skills with tools commonly used in the industry.

The game is available for both Academic and Commercial use.

Fundraising Scenario

The fundraising scenario will place participants in a context where they will be working on a fundraising campaign for a foundation, which is a 12-year-old, not-for-profit charitable organization with a million members.

The foundation has decided to add a direct contact campaign to its list of marketing activities. Participants will predict how many and which individuals to target in the campaign. The objective is to fundraise the highest donation amount while managing the expenses of contacting donors.



Cortex ANALYTICS SIMULATION

Fundraising Scenario

Last Update date: November 27, 2020

Figure 1. Fundraising Scenario

Goal*:
Maximize the net raised funds
DATA
Income
Education
Location
History
...
ANALYTICS
How many?
Who?
...
CALLING COST
1 million potential donors
...

*Operating Surplus = Total Amount Raised - Cost of Calls

Table 1. Cost Structure

Number of contacted members	Cost per call
0 - 60,000	5$/person
> 60,000	25$/person

Table 2. List of variables

Variable Name	Description	
ID	Member number (unique ID)	ID data
LastName	Last Name	
FirstName	First Name	
Woman	Sex (1=woman, 0=man)	Socio-demographic
Age	Age (years)	
Salary	Annual salary in USD	
Education	Highest education level	
City	Type of neighborhood	
SeniorList	Seniority for being on the VIP list	History*
NbActivities	Number of participations to annual meeting	
Referrals	Number of referrals	
Recency	Number of years since last gift	
Frequency	Number of donations	
Seniority	Number of years since first donation	
TotalGift	Total Donation since a member	
MinGift	Minimum donation since a member	
MaxGift	Maximum donation since on the VIP list	
Contact	Direct solicitation this year (Only applicable to Round 2)	
GaveLastYear	Whether or not the individual gave last year	Target
AmtLastYear	Amount given last year	
GaveThisYear	Whether or not the individual gave this year	
AmtThisYear	Amount given this year	

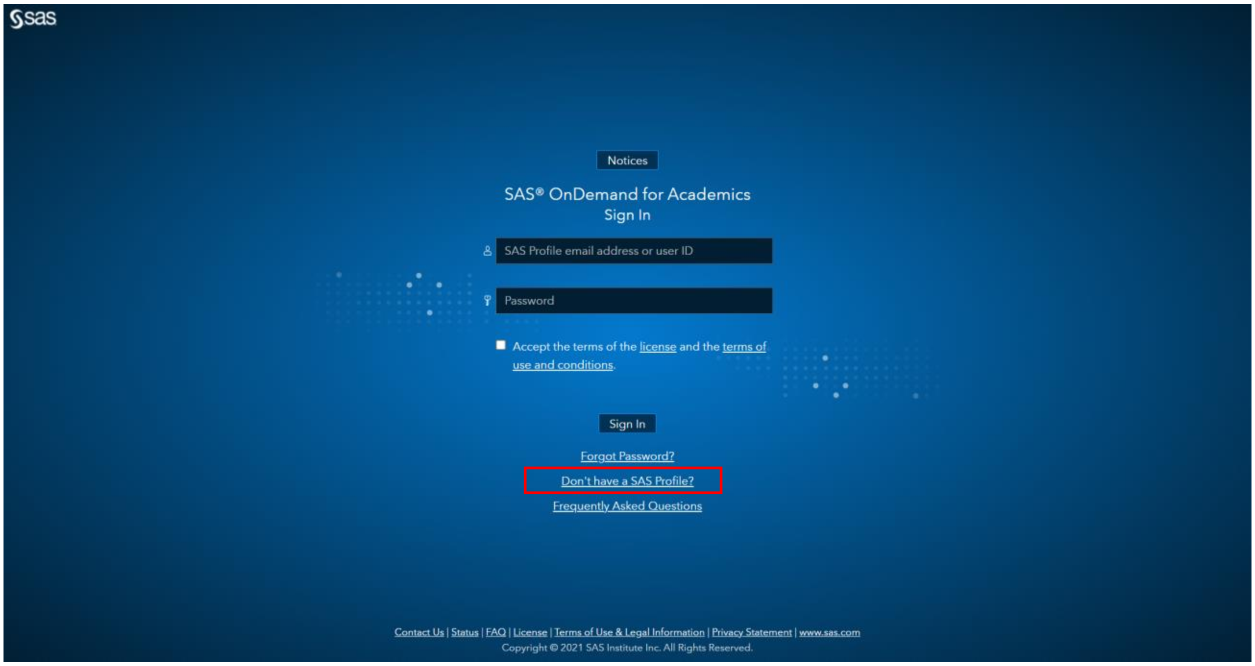
*** History (HIST) dataset gives the history of 10 years leading up to, but excluding, last year.**

PAGE 1

Source: https://erpsim.hec.ca/sites/default/files/cortex/GameSummary_Fundraising.pdf

How to play Cortex in SAS OnDemand for Academics (SODA)

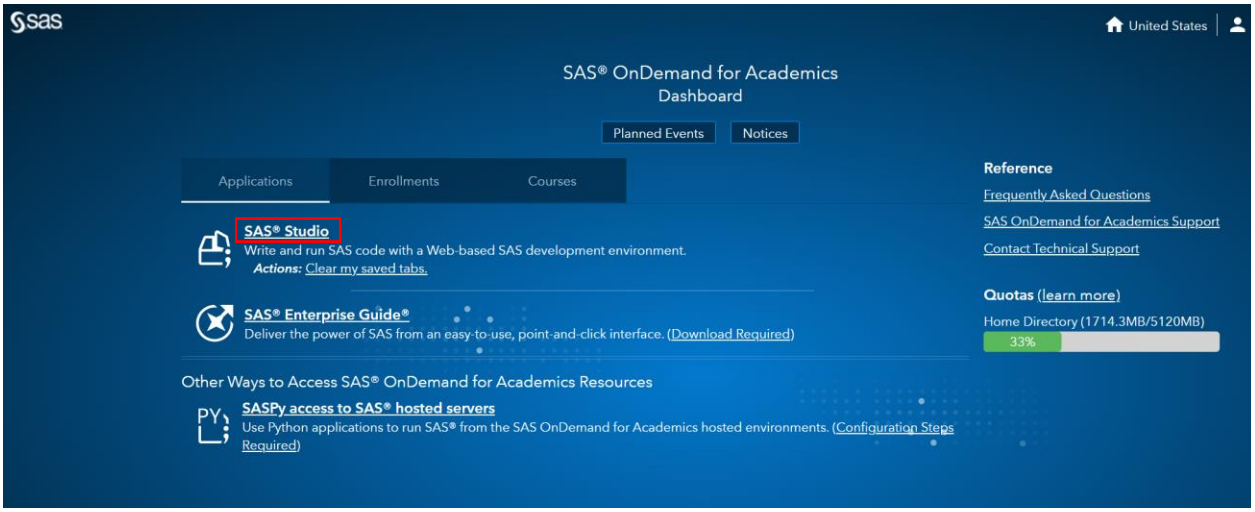
You can play the game by coding the projection model in SAS at SAS Studio. As a first step, please create your account for SODA. To register, visit https://odamid.oda.sas.com and click on **“Don't have a SAS Profile?”** to create your profile and register for an account.


sas
Notices
SAS® OnDemand for Academics
Sign In
SAS Profile email address or user ID
Password
Sign In
Forgot Password?
Don't have a SAS Profile?
Frequently Asked Questions
Contact Us | Status | FAQ | License | Terms of Use & Legal Information | Privacy Statement | www.sas.com

□ Accept the terms of the license and the terms of use and conditions.

Copyright © 2021 SAS Institute Inc. All Rights Reserved.

After you successfully create your account, you should be able to sign on the Control Center at https://odamid.oda.sas.com, and click “SAS Studio” to start coding.


sas
United States
SAS® OnDemand for Academics
Dashboard
Planned Events
Notices
Applications
Enrollments
Courses
SAS® Studio
SAS® Enterprise Guide®
Other Ways to Access SAS® OnDemand for Academics Resources
SASPy access to SAS® hosted servers
Reference
Frequently Asked Questions
SAS OnDemand for Academics Support
Contact Technical Support
Quotas (learn more)
33%

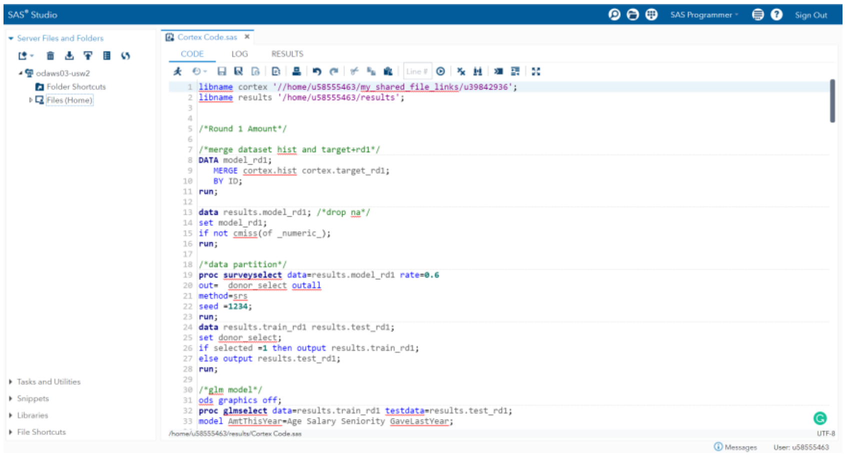
Write and run SAS code with a Web-based SAS development environment.
Actions: Clear my saved tabs.

Deliver the power of SAS from an easy-to-use, point-and-click interface. (Download Required)

Use Python applications to run SAS® from the SAS OnDemand for Academics hosted environments. (Configuration Steps Required)

Home Directory (1714.3MB/5120MB)

You will be provided a baseline model of SAS code to start with, you can modify the code to achieve the goal of the game – maximizing the donation amount.

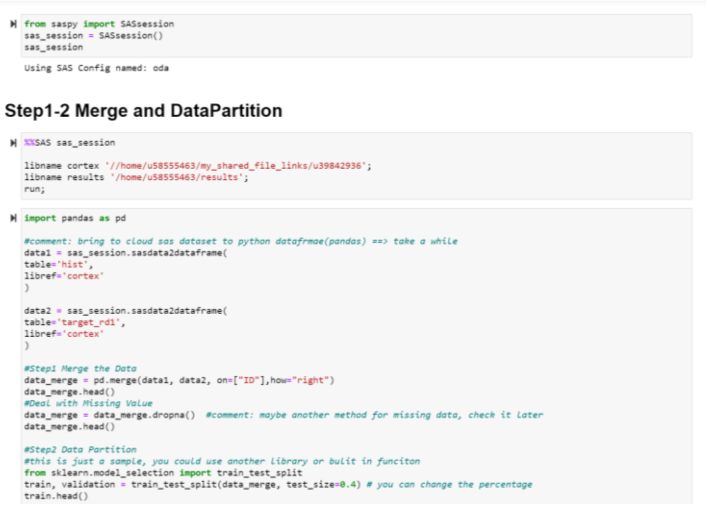


```
libname cortex '//home/u58555463/my_shared_file_links/u39842936';
libname results '/home/u58555463/results';


/*Round 1 Amount*/

/*merge dataset hist and target+rd1*/
DATA model_rd1;
   MERGE cortex.hist cortex.target_rd1;
   BY ID;
run;

data results.model_rd1; /*drop na*/
set model_rd1;
if not cmiss(of _numeric_);
run;

/*data partition*/
proc surveyselect data=results.model_rd1 rate=0.6
out=  donor_select outall
method=srs
seed =1234;
run;
data results.train_rd1 results.test_rd1;
set donor_select;
if selected =1 then output results.train_rd1;
else output results.test_rd1;
run;

/*glm model*/
ods graphics off;
proc glmselect data=results.train_rd1 testdata=results.test_rd1;
model AmtThisYear=Age Salary Seniority GaveLastYear;
```

How to play Cortex in Python Command

You can play the game by coding the projection model in Python. You will need to first establish connection between SODA and Python Command in order to access datasets in SAS terminal, more details are listed below. We will provide the baseline model of Python code in Jupyter Notebook. Python players will be expected to install Jupyter Notebook to continue the game.

After you successfully open the Jupyter Notebook, you can download the Python code of the baseline model from Github, you can modify the code to achieve the goal of the game – maximizing the donation amount.



```
from saspy import SASsession
sas_session = SASsession()
sas_session
```

```
Using SAS Config named: oda
```

Step1-2 Merge and DataPartition

```
%%SAS sas_session

libname cortex '//home/u58555463/my_shared_file_links/u39842936';
libname results '/home/u58555463/results';
run;
```

```
import pandas as pd

#comment: bring to cloud sas dataset to python datafrmae(pandas) ==> take a while
data1 = sas_session.sasdata2dataframe(
table='hist',
libref='cortex'
)

data2 = sas_session.sasdata2dataframe(
table='target_rd1',
libref='cortex'
)

#Step1 Merge the Data
data_merge = pd.merge(data1, data2, on=["ID"],how="right")
data_merge.head()
#Deal with Missing Value
data_merge = data_merge.dropna()  #comment: maybe another method for missing data, check it later
data_merge.head()

#Step2 Data Partition
#this is just a sample, you could use another library or bulit in funciton
from sklearn.model_selection import train_test_split
train, validation = train_test_split(data_merge, test_size=0.4) # you can change the percentage
train.head()
```

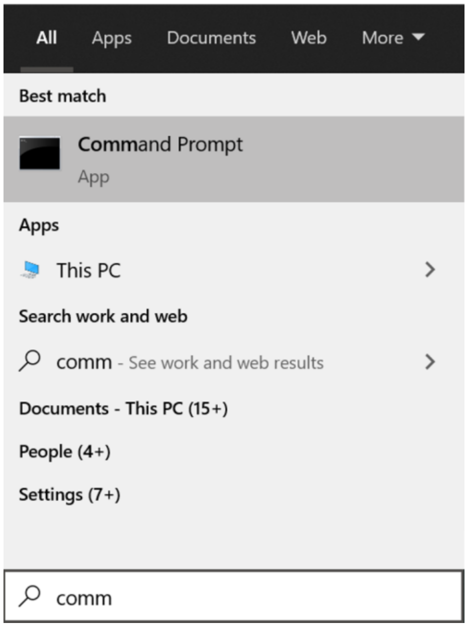
SASPy Installation and Configuration

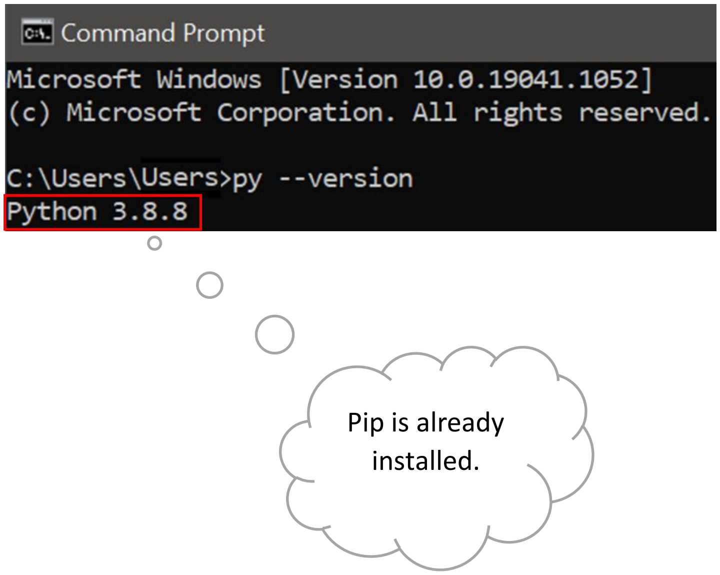
To install the SASPy in your Python Command, you need to ensure that you meet the prerequisite:

1. Java version 1.8.0_162 or higher
2. Python 3.3 or higher
3. SASPy 3.3.4 or higher

Step 1: Install PIP (Windows)

You need to open Command Prompt and check your Python version by entering code **py --version**, if it is higher than 3.4, pip is already installed.

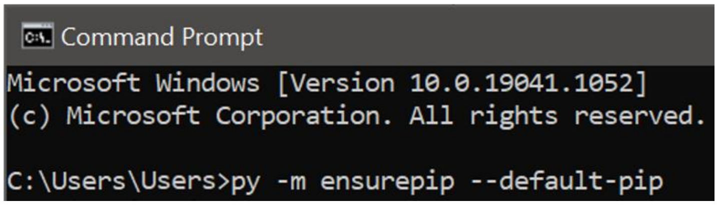




```
Microsoft Windows [Version 10.0.19041.1052]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Users>py --version
Python 3.8.8
```

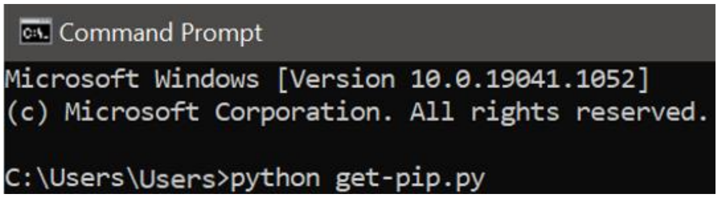
Pip is already installed.

If pip is not installed, you should first try to bootstrap it from the standard library by entering the code **py -m ensurepip --default-pip** in Command Prompt.



```
Microsoft Windows [Version 10.0.19041.1052]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Users>py -m ensurepip --default-pip
```

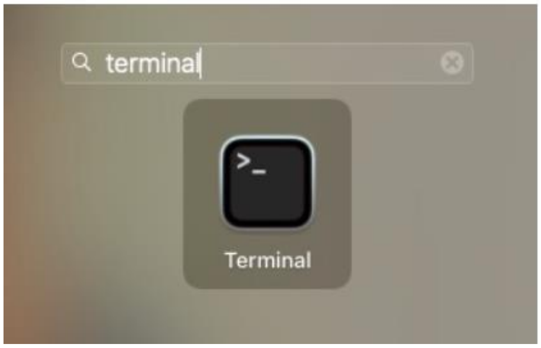
If that still does not allow you to run code **python -m pip**, you need to download get-pip.py, and run code **python get-pip.py** in the Command Prompt. This will install or upgrade pip. Additionally, it will install setup tools and wheel if they are not installed already.

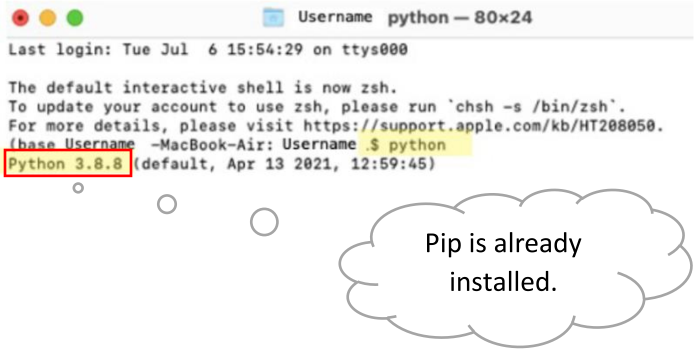


```
Microsoft Windows [Version 10.0.19041.1052]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Users>python get-pip.py
```

Step 1: Install PIP (MACOs)

You need to open Terminal and check your Python version, if it is higher than 3.4, pip is already installed.

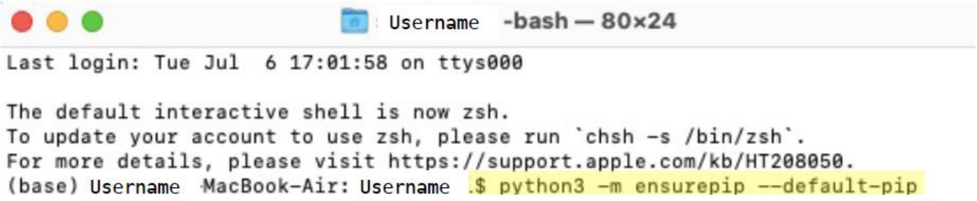




```
Last login: Tue Jul  6 15:54:29 on ttys000

The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
(base Username  -MacBook-Air: Username .$ python
Python 3.8.8 (default, Apr 13 2021, 12:59:45)
```

Pip is already installed.

If pip is not installed, you should first try to bootstrap it from the standard library by entering the code **python3 -m ensurepip -- default-pip** in Terminal.



```
Last login: Tue Jul  6 17:01:58 on ttys000

The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
(base) Username  MacBook-Air: Username .$ python3 -m ensurepip --default-pip
```

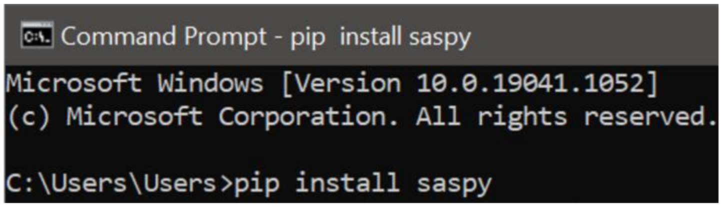
If that still does not allow you to run code **python -m pip**, you need to download get-pip.py, and run code **python get-pip.py** in the Command Prompt. This will install or upgrade pip. Additionally, it will install setup tools and wheel if they are not installed already.

Link for Reference

Step 2: Use PIP to Install SASPy (Windows)

This package installs just like any other Python package. It is a pure Python package and works with Python 3.x installations. To install the latest version using pip, you should execute the code ***pip install saspy*** in the Command Prompt.

If you prefer conda install, you can use that from the conda-forge channel.

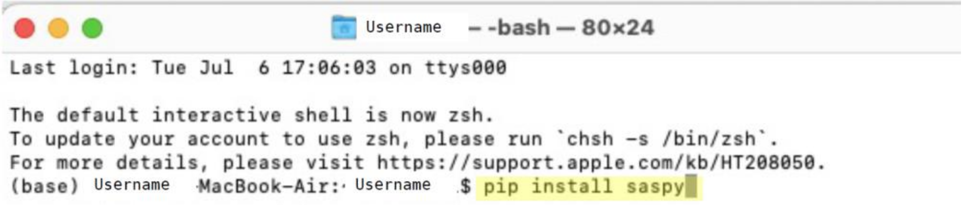


```
Microsoft Windows [Version 10.0.19041.1052]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Users>pip install saspy
```

Step 2: Use PIP to Install SASPy (MACOs)

This package installs just like any other Python package. It is a pure Python package and works with Python 3.x installations. To install the latest version using pip, you should execute the code ***pip install saspy*** in the Command Prompt.

If you prefer conda install, you can use that from the conda-forge channel.



```
Last login: Tue Jul  6 17:06:03 on ttys000

The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
(base) Username MacBook-Air:~ Username $ pip install saspy
```

Link for Reference

Step 3: Configure Python to access SODA (Windows)

Following the steps below you should be able to configure Python to access SOAD:

1. Create a text editor named ***sascfg_personal.py*** containing the information based on your SOAD Home Region.
2. Delete the file extension when saving file, if the Notepad in your Windows does not allow you to do so, download **Notepad++** for file creation.

3. You can simply copy and paste the entire code block below. Be sure to uncomment only the 'iomhost' key for your home region. *** US Region is uncommented as an example.** You can leave the others commented out or delete them.

   NOTE: You will likely need to change the JAVA location

```
SAS_config_names=['oda']
oda = {'java' : 'C:\\Program Files (x86)\\Common Files\\Oracle\\Java\\javapath\\java.exe',
#US Home Region
'iomhost' : ['odaws01-usw2.oda.sas.com','odaws02-usw2.oda.sas.com','odaws03-
usw2.oda.sas.com','odaws04-usw2.oda.sas.com'],
#European Home Region
#'iomhost' : ['odaws01-euw1.oda.sas.com','odaws02-euw1.oda.sas.com'],
#Asia Pacific Home Region
#'iomhost' : ['odaws01-apse1.oda.sas.com','odaws02-apse1.oda.sas.com'],
'iomport' : 8591,
'authkey' : 'oda',
'encoding' : 'utf-8'
}
```

4. Save ***sascfg_personal.py*** to your SASPy installation location.

5. Run the following commands in the command prompt window to get the full pathname of SASPy installation location.

```
import saspy, os
print(saspy.__file__.replace('__init__.py', 'sascfg_personal.py’))
```

6. Create a text editor named ***_authinfo*** using the coding block below as a template. You will need to change ***ODA_EMAIL/ODA_USERNAME*** and ***ODA_PASSWORD*** to your SODA credentials.

```
oda user ODA_EMAIL/ODA_USERNAME password ODA_PASSWORD
```

7. Save ***_authinfo*** to your user's home directory *C:\Users\'YOUR_USERNAME'* on Windows.

Step 3: Configure Python to access SODA (MACOs)

Following the steps below you should be able to configure Python to access OAD:

1. Create a text editor named ***sascfg_personal.py*** containing the information based on your SODA Home Region.

2. MacOS users could use built-in software TextEdit to delete the file extension when saving file.

3. You can simply copy and paste the entire code block below. Be sure to uncomment only the 'iomhost' key for your home region. *** US Region is uncommented as an example.** You can leave the others commented out or delete them.
   NOTE: You will likely need to change the JAVA location. For MAC users, your Java path usually is “/usr/bin/java”

   ```
   SAS_config_names=['oda']
   oda = {'java' : 'C:\\Program Files (x86)\\Common Files\\Oracle\\Java\\javapath\\java.exe',
   #US Home Region
   'iomhost' : ['odaws01-usw2.oda.sas.com','odaws02-usw2.oda.sas.com','odaws03-
   usw2.oda.sas.com','odaws04-usw2.oda.sas.com'],
   #European Home Region
   #'iomhost' : ['odaws01-euw1.oda.sas.com','odaws02-euw1.oda.sas.com'],
   #Asia Pacific Home Region
   #'iomhost' : ['odaws01-apse1.oda.sas.com','odaws02-apse1.oda.sas.com'],
   'iomport' : 8591,
   'authkey' : 'oda',
   'encoding' : 'utf-8'
   }
   ```

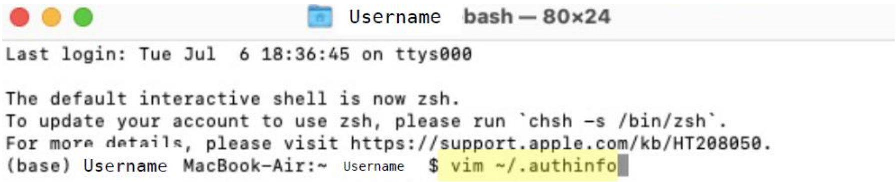
4. Save ***sascfg_personal.py*** to your SASPy installation location.

5. Use Terminal to get the full pathname of SASPy installation location:

   1. First type “python” and hit Enter
   2. Run the following commands

   ```
   import saspy, os
   print(saspy.__file__.replace('__init__.py', 'sascfg_personal.py’))
   ```

6. Open Terminal and Type command **vim ~/.authinfo**



```
Last login: Tue Jul  6 18:36:45 on ttys000

The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
(base) Username MacBook-Air:~ Username  $ vim ~/.authinfo
```

7. Press " a " to enter edit mode.

8. Type in the following text, using the coding block below as a template. You will need to change ***ODA_EMAIL/ODA_USERNAME*** and ***ODA_PASSWORD*** to your SODA credentials.

```
oda user ODA_EMAIL/ODA_USERNAME password ODA_PASSWORD
```

9. Press "ESC" and then press ":wq" to save and quit.

Link for Reference

Step 4: Connect to the SODA Servers from Python (Windows and MACOs)

This step should be performed any time that you want to connect to hosted SAS servers.

From a Python prompt or from another Python interface, like Jupyter Notebook, use the following commands to confirm a connection to SODA.

```
import saspy
sas_session = saspy.SASsession()
sas_session
```

Note: If you encounter, **"*None of the requested encryption algorithms are supported by both peers: AES*"**, please confirm you are running the correct version of Java (1.8.0_162 or greater) and Contact Us for additional assistance.

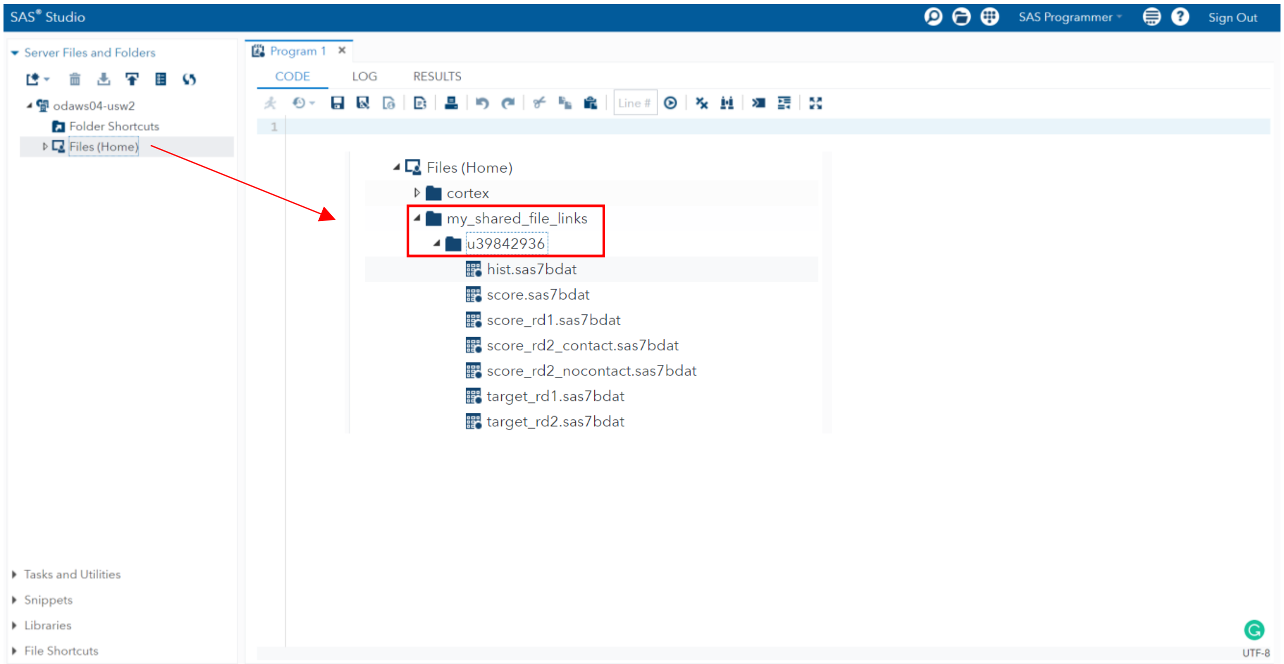
Link for Reference

Access to Dataset

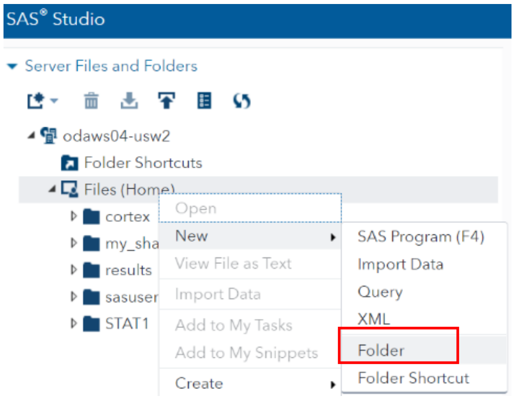
In order to access the dataset in the SAS terminal, you need to follow these steps:

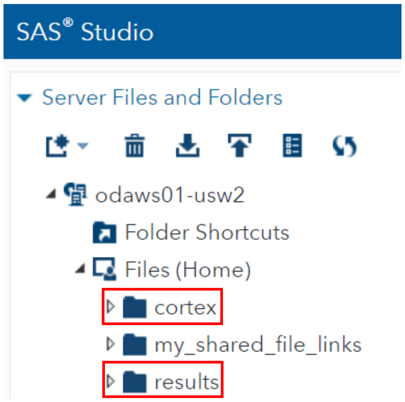
1. Sign on the Control Center at https://odamid.oda.sas.com.
2. Look for the Enroll in a course link in the "Enrollments" section near the bottom of the page. Click this link to start the enrollment.
3. Enter the course code: 83fd3afd-8964-4f5b-a612-7b665ed69104

Then you can open your SAS Studio interface, you can see all the cortex dataset in the “u39842936” folder under the“my_shared_file_links” folder.

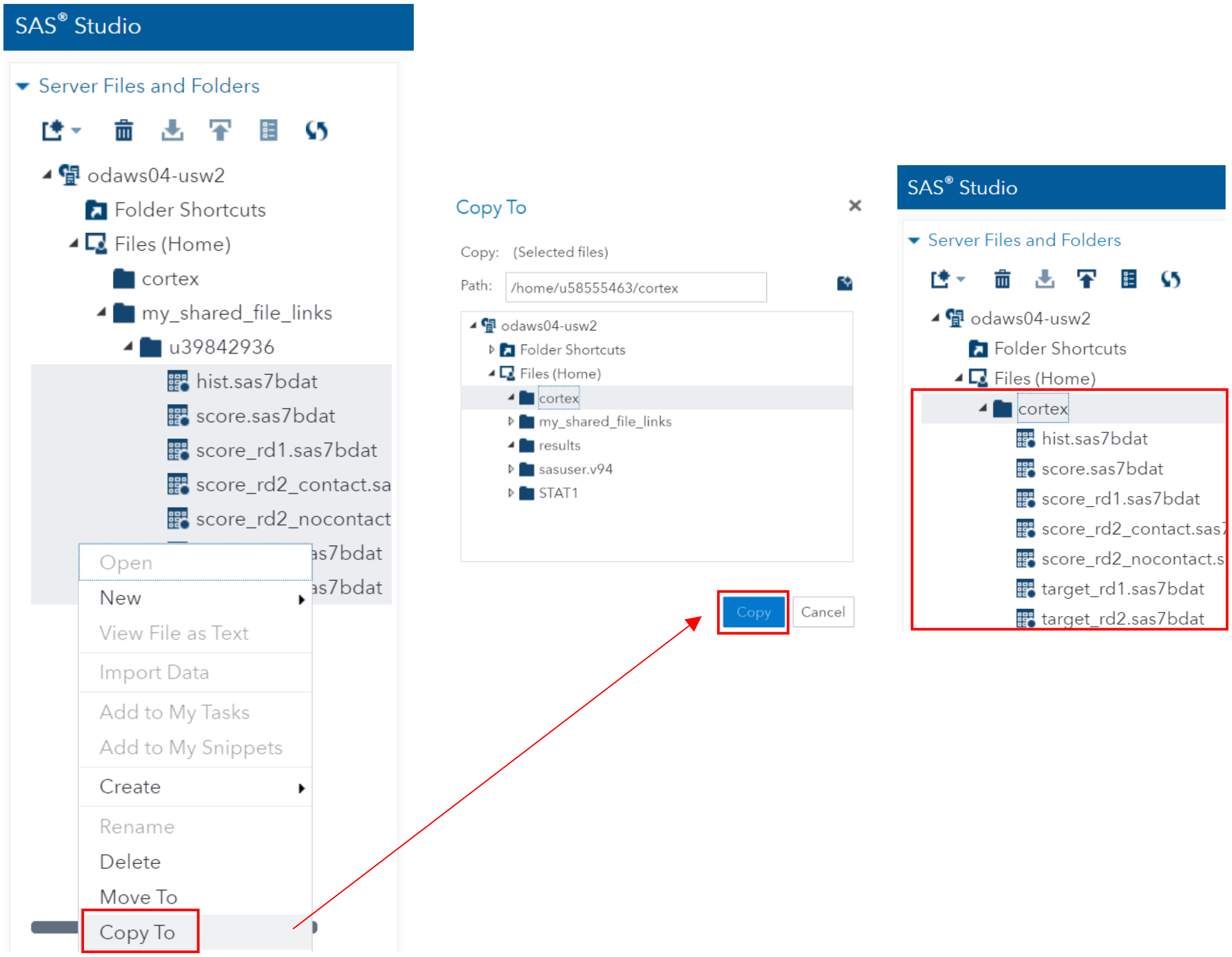


After that, you need to create a folder named “cortex” with all datasets inside. Then create another folder named “results”.





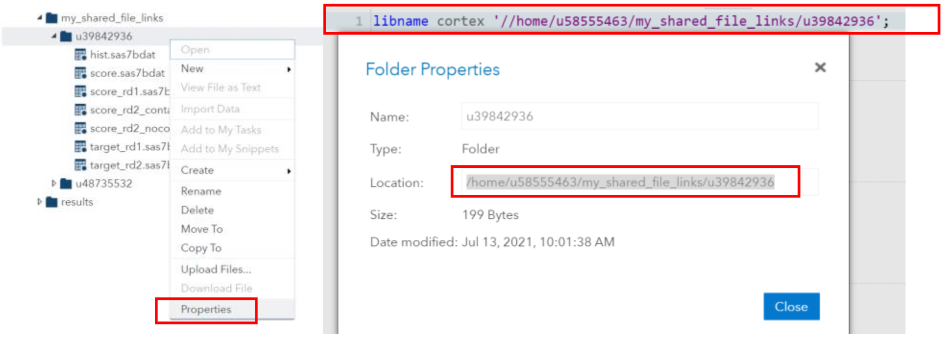
You should copy all the datasets from “u39842936” folder to the “cortex” folder, this is important because you do not have the authority to do any edit/change to the datasets in “u39842936” folder.



Copy: (Selected files)

Path: /home/u58555463/cortex

Copy the path of “u39842936” folder and create “cortex” library by code: libname cortex '/***path***'



```
libname cortex '//home/u58555463/my_shared_file_links/u39842936';
```

Folder Properties

Name: u39842936

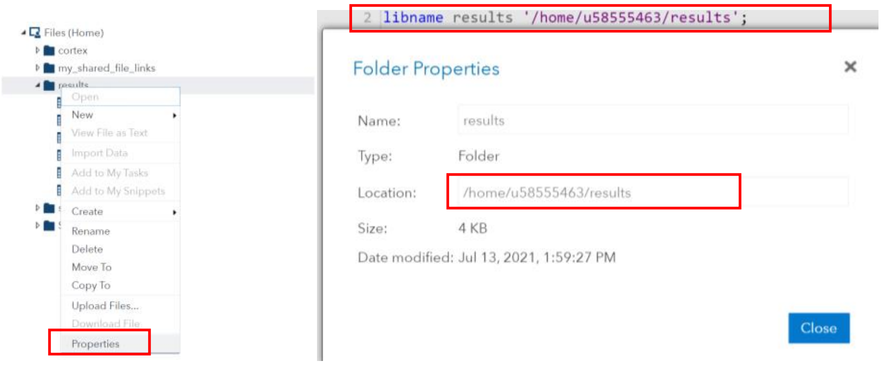
Type: Folder

Location: /home/u58555463/my_shared_file_links/u39842936

Size: 199 Bytes

Date modified: Jul 13, 2021, 10:01:38 AM

Copy the path of “results” folder and create “results” library by code: libname cortex '/***path***', then you can restore any interim dataset to the “work” or “results” folder/libraries.



```
libname results '/home/u58555463/results';
```

Folder Properties

Name: results

Type: Folder

Location: /home/u58555463/results

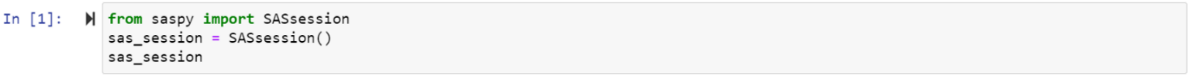
Size: 4 KB

Date modified: Jul 13, 2021, 1:59:27 PM

Link for Reference

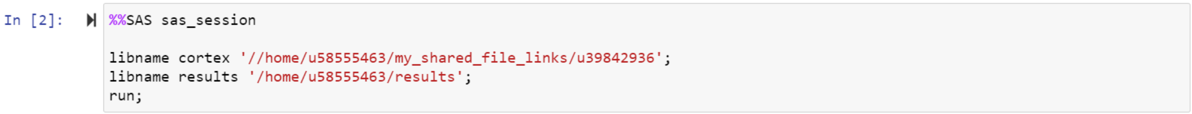
You can then start coding in the SAS Studio.

If you want to use Python Command for coding, you need to connect your SAS terminal to the Python Command. In this case, we will use Jupyter Notebook as an example.



```
In [1]:  from saspy import SASsession
         sas_session = SASsession()
         sas_session
```

Then you need to add this code ***%%SAS sas_session*** to create libraries in SAS language to access datasets in SAS terminal.



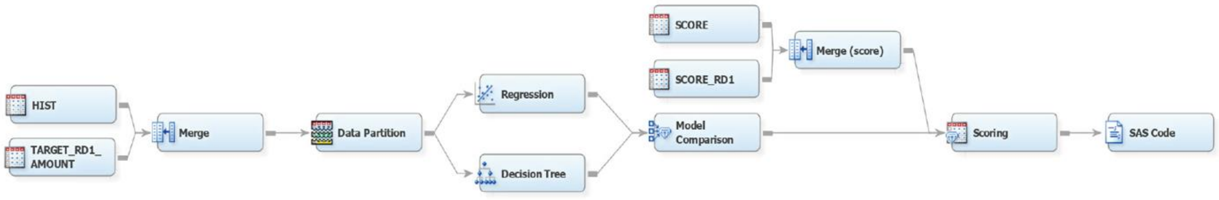
```
In [2]:  %%SAS sas_session

         libname cortex '//home/u58555463/my_shared_file_links/u39842936';
         libname results '/home/u58555463/results';
         run;
```

Round 1 - Game Summary

In Round 1, your mission is to predict the amount that each member would give next year to the foundation. You would then use this prediction to call the most valuable members (i.e., members who are predicted to give the most to the foundation).

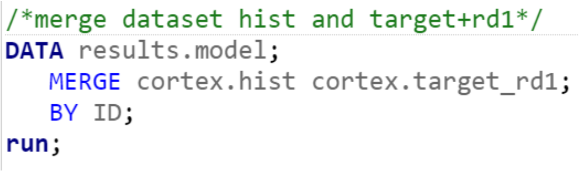
Round 1 - Model Description (SAS)

There are 6 steps for game completion: Data Merge, Data Partition, Model Building, Model Comparison, Scoring Data and Export Results.



Step 1: Data Merge

The first step is to merge two datasets into one for better modification.

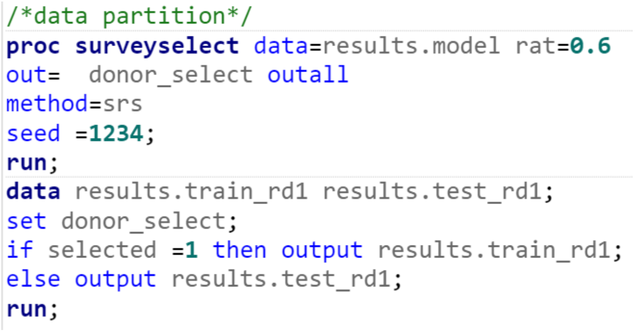


```
/*merge dataset hist and target+rd1*/
DATA results.model;
   MERGE cortex.hist cortex.target_rd1;
   BY ID;
run;
```

Link for Reference

Step 2: Data Partition

The second step is data partition, this will help the model to increase its performance, please note that this is a default model, you can change the training and validation percentage for a better model performance.

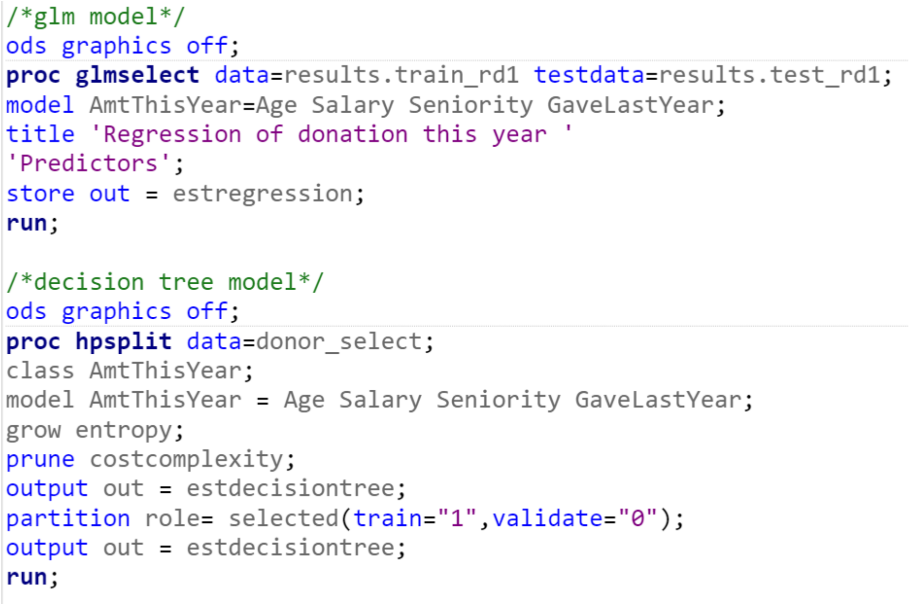


```
/*data partition*/
proc surveyselect data=results.model rat=0.6
out=  donor_select outall
method=srs
seed =1234;
run;
data results.train_rd1 results.test_rd1;
set donor_select;
if selected =1 then output results.train_rd1;
else output results.test_rd1;
run;
```

Link for Reference

Step 3: Model Building

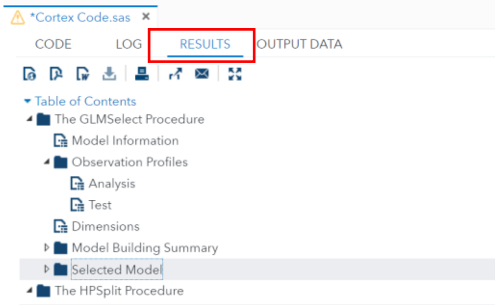
In this step, linear regression and decision tree models are used for analysis, you can also choose other model methods (i.e., logistic regression, HP forest, gradient boosting) to predict the amount donated. We recommend you perform a descriptive analysis to select independent variables for prediction.



```
/*glm model*/
ods graphics off;
proc glmselect data=results.train_rd1 testdata=results.test_rd1;
model AmtThisYear=Age Salary Seniority GaveLastYear;
title 'Regression of donation this year '
'Predictors';
store out = estregression;
run;

/*decision tree model*/
ods graphics off;
proc hpsplit data=donor_select;
class AmtThisYear;
model AmtThisYear = Age Salary Seniority GaveLastYear;
grow entropy;
prune costcomplexity;
output out = estdecisiontree;
partition role= selected(train="1",validate="0");
output out = estdecisiontree;
run;
```

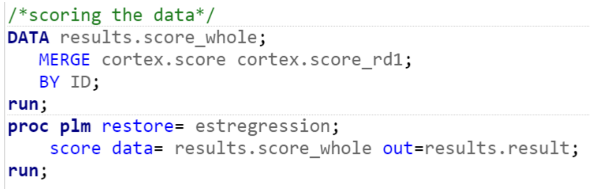
Step 4: Model Comparison

When comparing models, you can click RESULTS tab for comparison, SAS will compute statistics automatically, you can then choose different criteria to determine which model fits better for prediction. In the default model, linear regression model is selected for scoring.



Step 5: Scoring Data

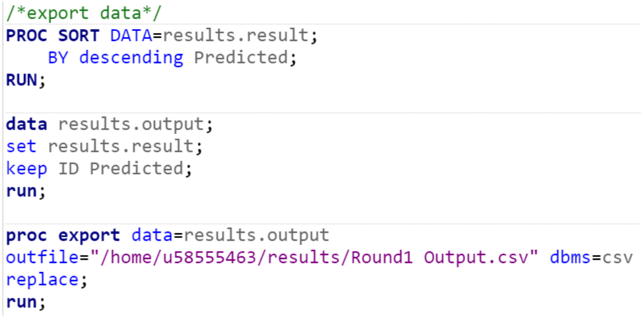
Scoring new data to compute predictions for an existing model is a fundamental stage in the analytics life cycle, after you score out the dataset, you can export the model result for upload.



```
/*scoring the data*/
DATA results.score_whole;
   MERGE cortex.score cortex.score_rd1;
   BY ID;
run;
proc plm restore= estregression;
    score data= results.score_whole out=results.result;
run;
```

Step 6: Export Results

Following code will help you to export the model result into a CSV file. You can sort the data by predicted amount in the descending order before export the file, this will help you determine number of donors you want to contact.

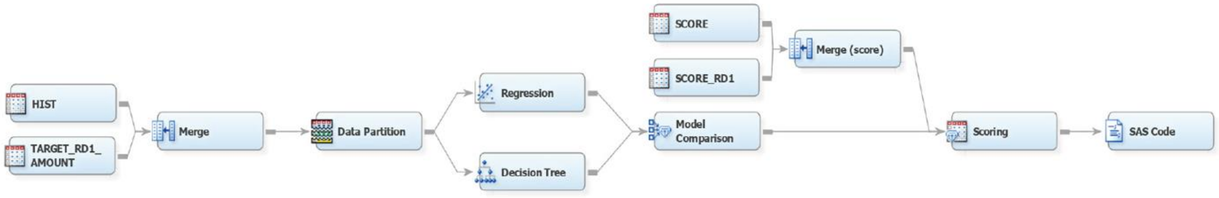


```
/*export data*/
PROC SORT DATA=results.result;
    BY descending Predicted;
RUN;

data results.output;
set results.result;
keep ID Predicted;
run;

proc export data=results.output
outfile="/home/u58555463/results/Round1 Output.csv" dbms=csv
replace;
run;
```

Round 1 - Model Description (Python)

NOTE: You need to install ***saspy, pandas***, ***numpy***, and ***sklearn*** python packages to run the baseline model in your python command.

There are 6 steps for game completion: Data Merge, Data Partition, Model Building, Model Comparison, Scoring Data and Export Results.



Step 1: Data Merge

The first step is to merge two datasets into one for better modification.

Step 2: Data Partition

The second step is data partition, this will help the model to increase its performance, please note that this is a default model, you can change the training and validation percentage for a better model performance.

Step 3: Model Building

In this step, linear regression and decision tree models are used for analysis, you can also choose other model methods (i.e., logistic regression, HP forest, gradient boosting) to predict the amount donated. We recommend you perform a descriptive analysis to select independent variables for prediction.

Step 4: Model Comparison

When comparing models, you need to choose a criteria and calculate it manually to determine which model fits better for prediction since Python will not compute the statistical results automatically. In the default model, we compare the MSE of two model, and the linear regression model is selected for scoring.

Step 5: Scoring Data

Scoring new data to compute predictions for an existing model is a fundamental stage in the analytics life cycle, after you score out the dataset, you can export the model result for upload.

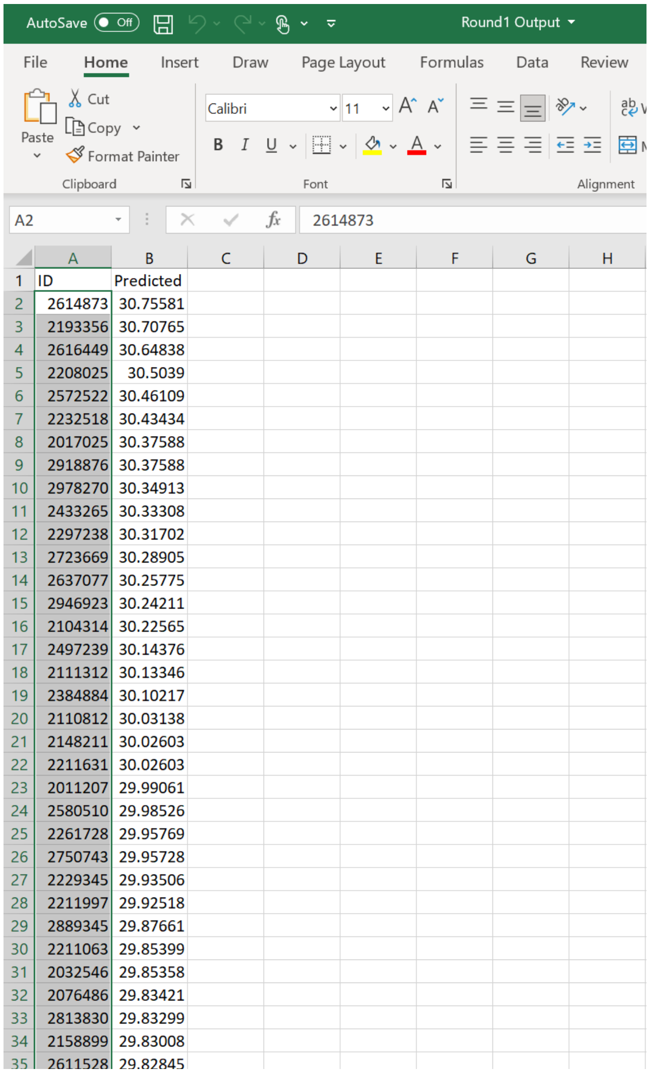
Step 6: Export Results

Following code will help you to export the model result into a CSV file. You can sort the data by predicted amount in the descending order before export the file, this will help you determine number of donors you want to contact.

Round 1 - Upload Decisions

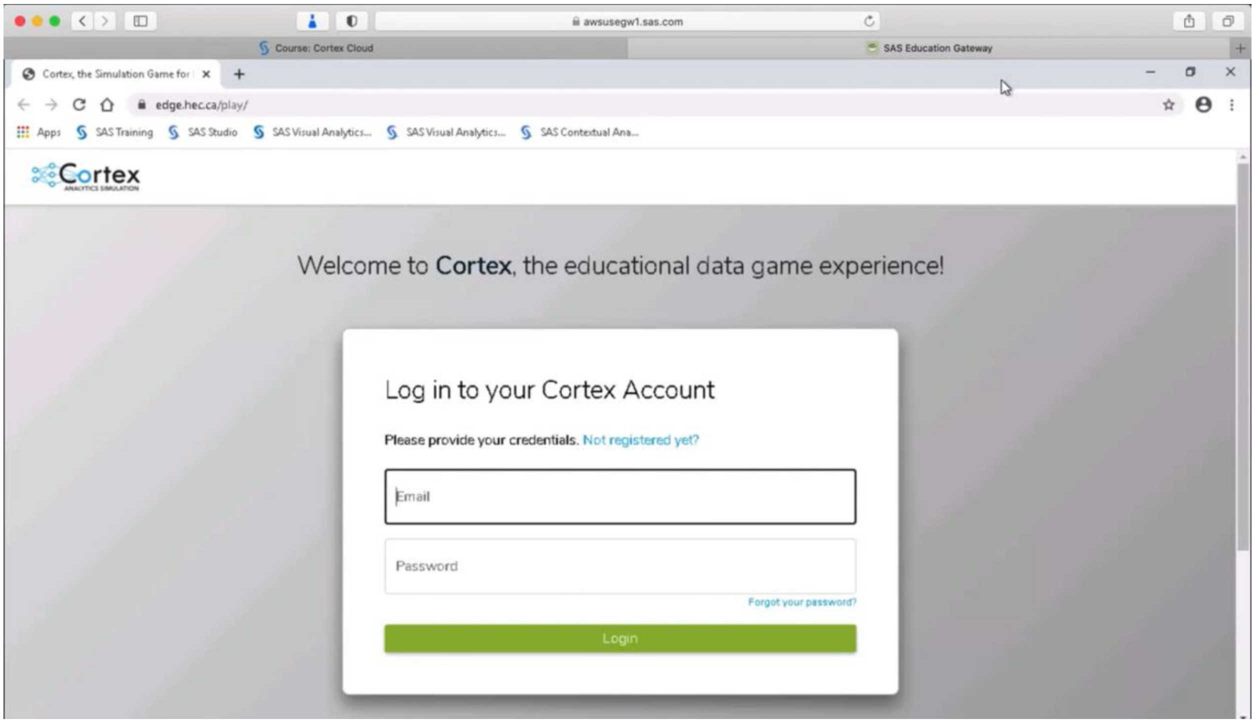
After you export the result, you need to create a new CSV file, select the number of people you want to contact in the file exported, and copy the ID list ONLY to the new CSV file. You need to **SAVE** and **CLOSE** the new CSV file before uploading it to the game leaderboard.

Select ID numbers ONLY, paste them to a new CSV file.



	A	B
1	ID	Predicted
2	2614873	30.75581
3	2193356	30.70765
4	2616449	30.64838
5	2208025	30.5039
6	2572522	30.46109
7	2232518	30.43434
8	2017025	30.37588
9	2918876	30.37588
10	2978270	30.34913
11	2433265	30.33308
12	2297238	30.31702
13	2723669	30.28905
14	2637077	30.25775
15	2946923	30.24211
16	2104314	30.22565
17	2497239	30.14376
18	2111312	30.13346
19	2384884	30.10217
20	2110812	30.03138
21	2148211	30.02603
22	2211631	30.02603
23	2011207	29.99061
24	2580510	29.98526
25	2261728	29.95769
26	2750743	29.95728
27	2229345	29.93506
28	2211997	29.92518
29	2889345	29.87661
30	2211063	29.85399
31	2032546	29.85358
32	2076486	29.83421
33	2813830	29.83299
34	2158899	29.83008
35	2611528	29.82845

Then you need to log in to the Leaderboard: https://edge.hec.ca/play/

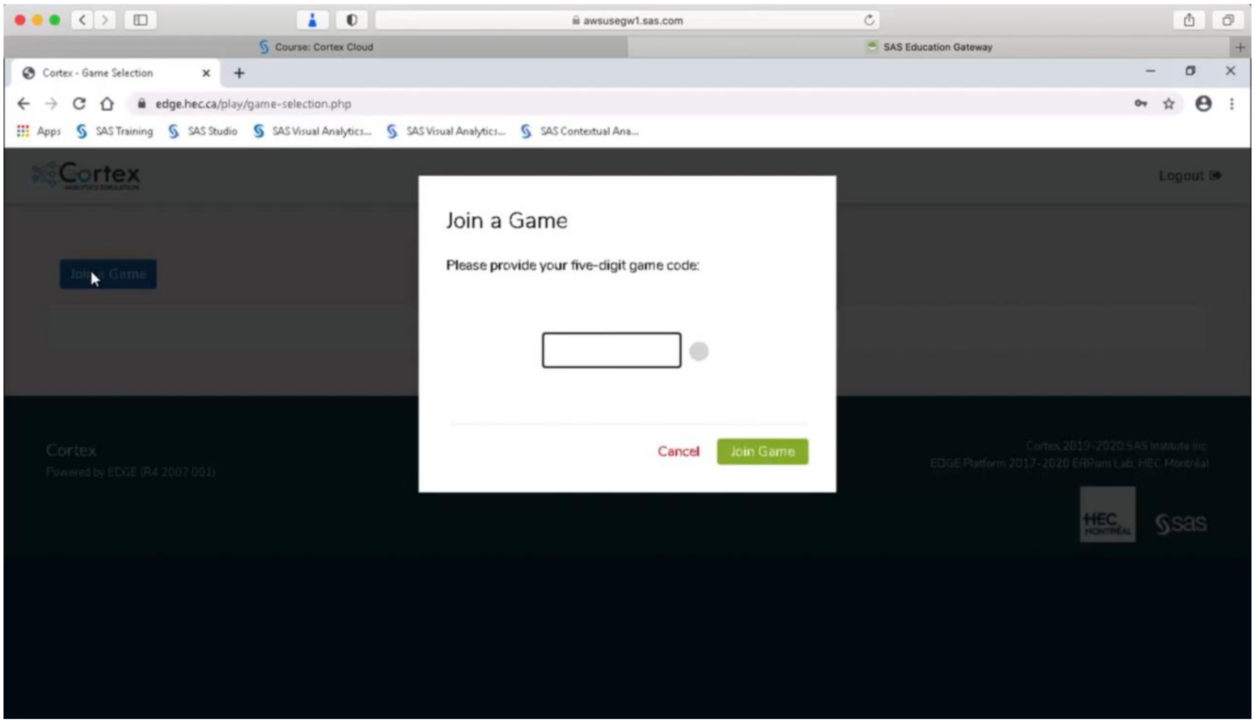


Welcome to **Cortex**, the educational data game experience!

Log in to your Cortex Account

Please provide your credentials. Not registered yet?

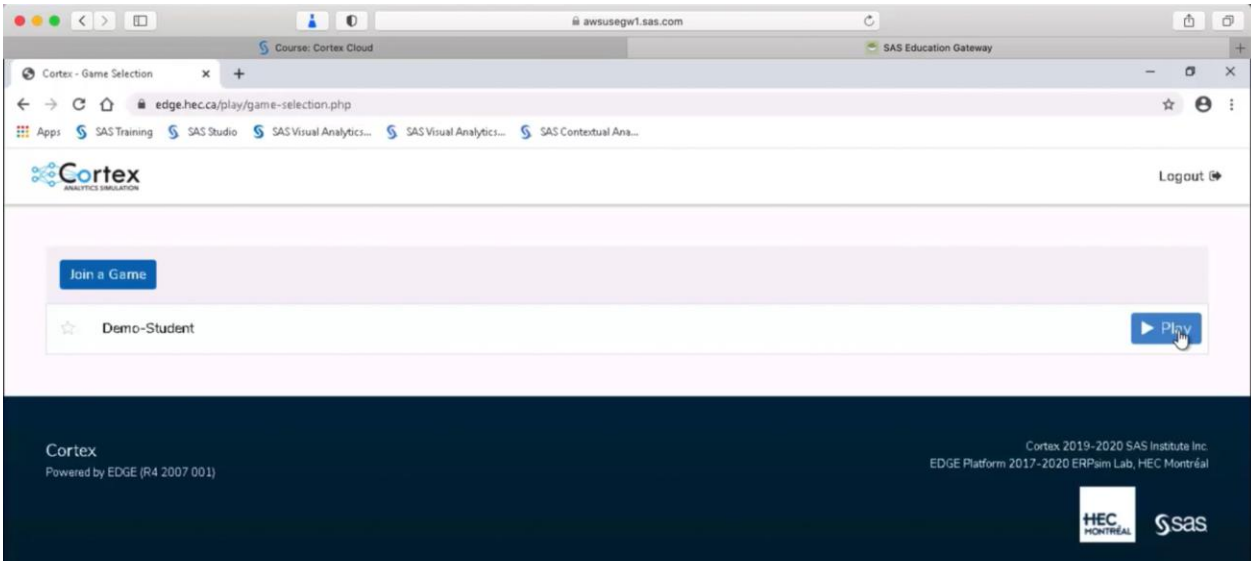
Click **Join a Game** button and enter the Five-Digit Game Code that you should have received from your instructor.



Join a Game

Please provide your five-digit game code:

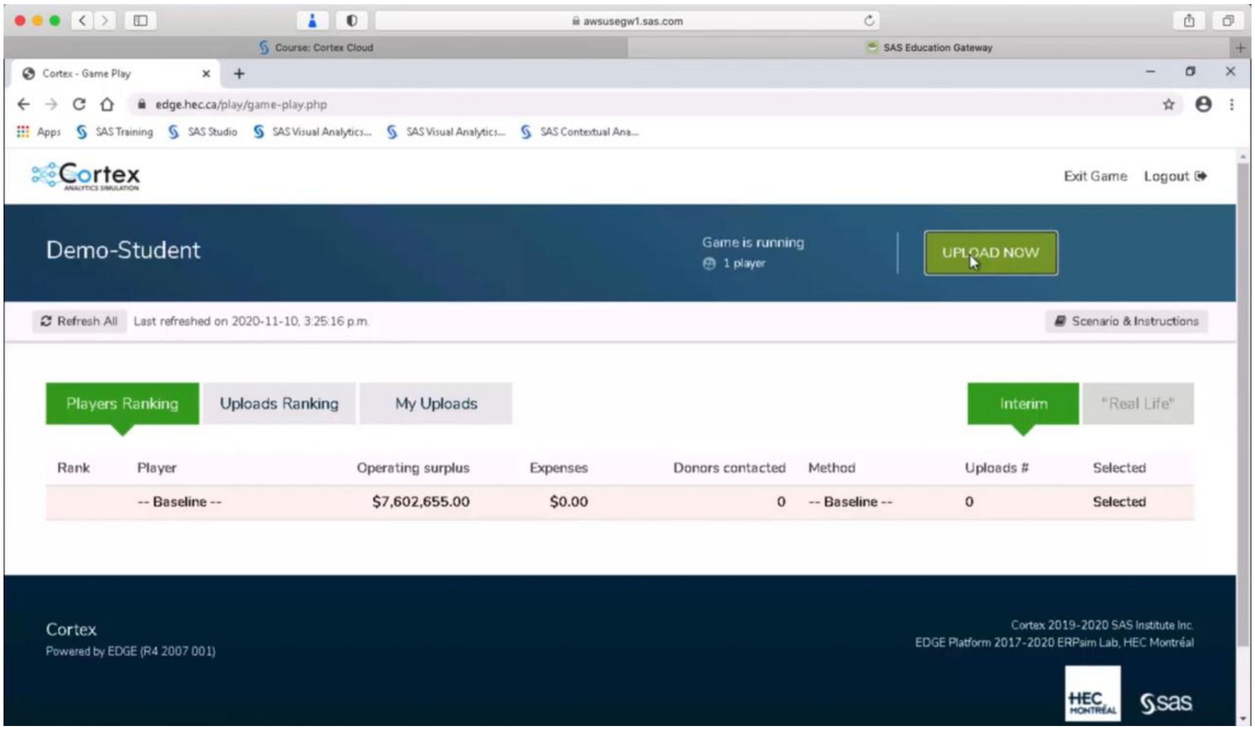
Click **Play** button to start the game.



Cortex
Powered by EDGE (R4 2007 001)

Cortex 2019-2020 SAS Institute Inc.
EDGE Platform 2017-2020 ERPsim Lab, HEC Montréal

Click **UPLOAD NOW** button to upload CSV file.



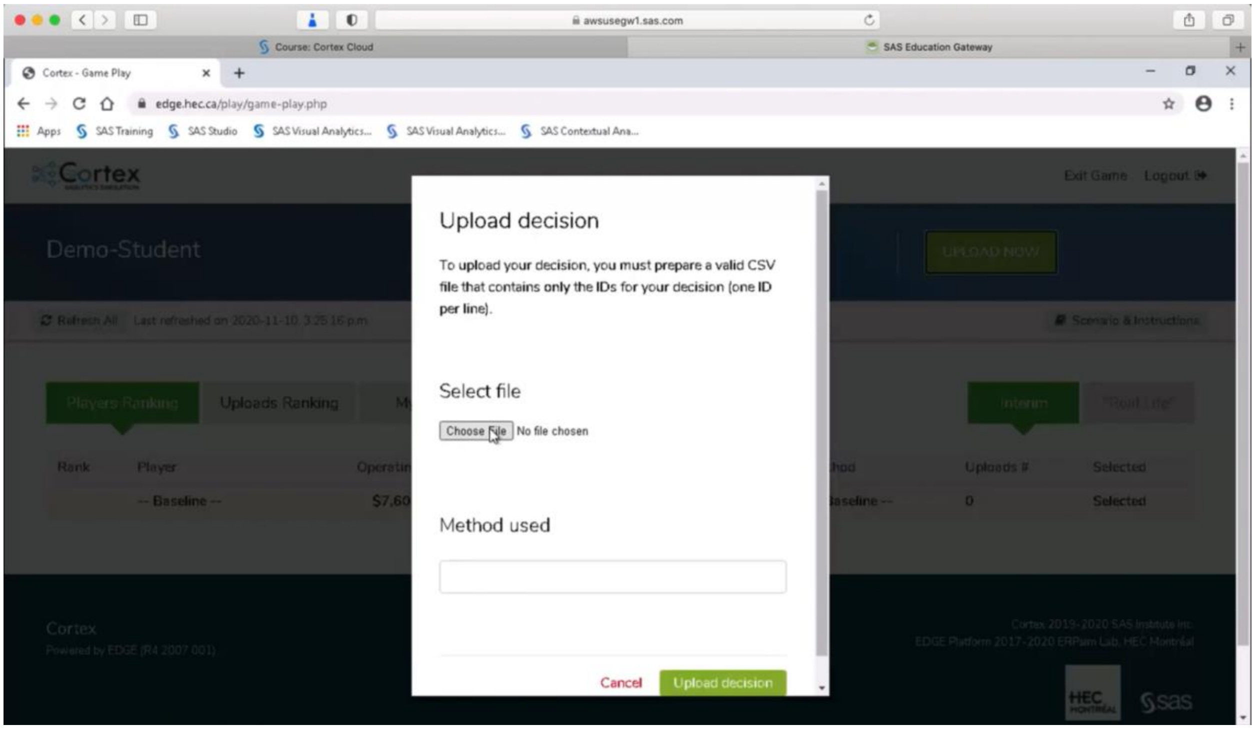
Last refreshed on 2020-11-10, 3:25:16 p.m.

Rank	Player	Operating surplus	Expenses	Donors contacted	Method	Uploads #	Selected
	-- Baseline --	$7,602,655.00	$0.00	0	-- Baseline --	0	Selected

Cortex
Powered by EDGE (R4 2007 001)

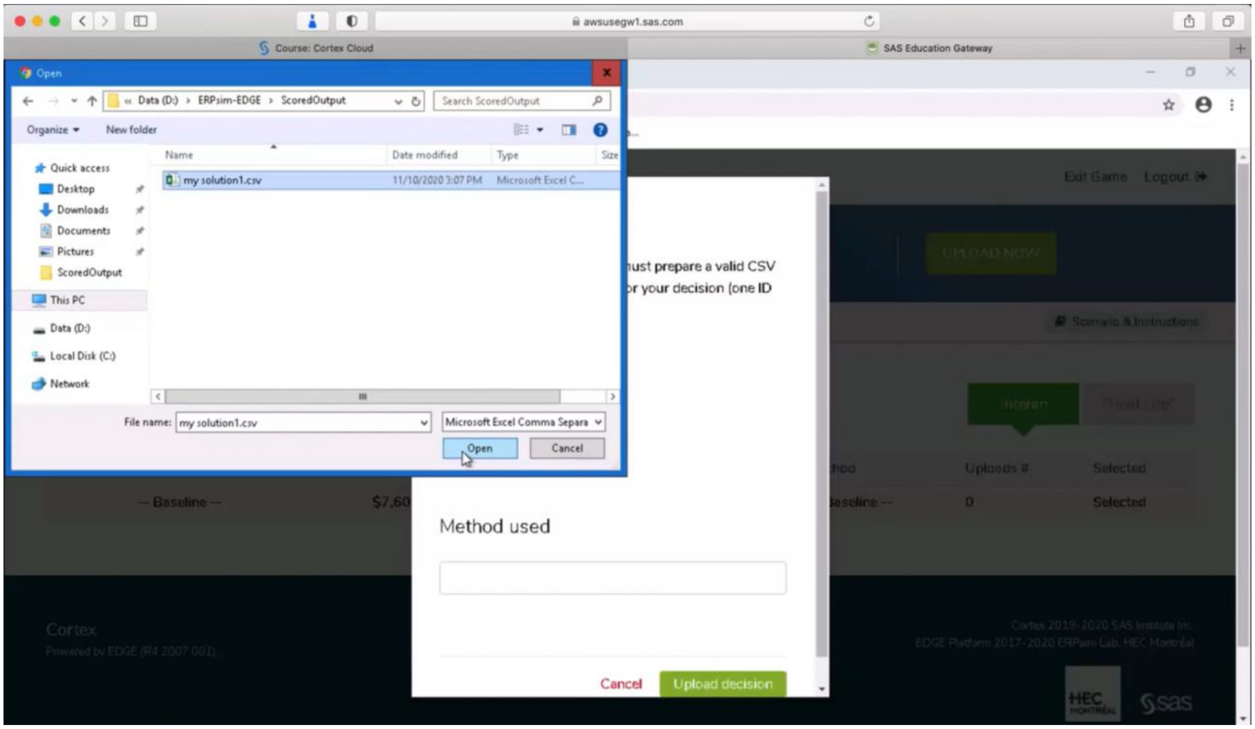
Cortex 2019-2020 SAS Institute Inc.
EDGE Platform 2017-2020 ERPsim Lab, HEC Montréal

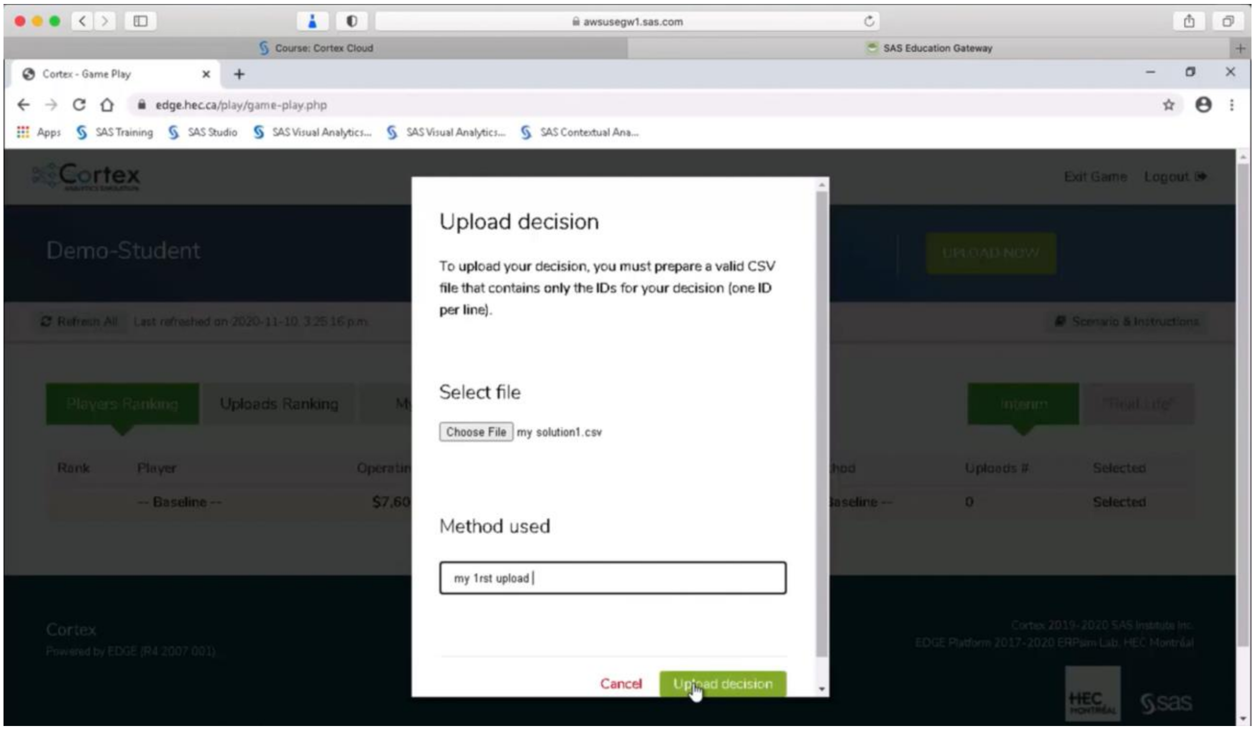
Click **Choose File** button to select your new CSV file, you can name your upload under the **Method used**, and click **Upload decision** button to complete your upload.



Upload decision

To upload your decision, you must prepare a valid CSV file that contains only the IDs for your decision (one ID per line).





Upload decision

To upload your decision, you must prepare a valid CSV file that contains only the IDs for your decision (one ID per line).

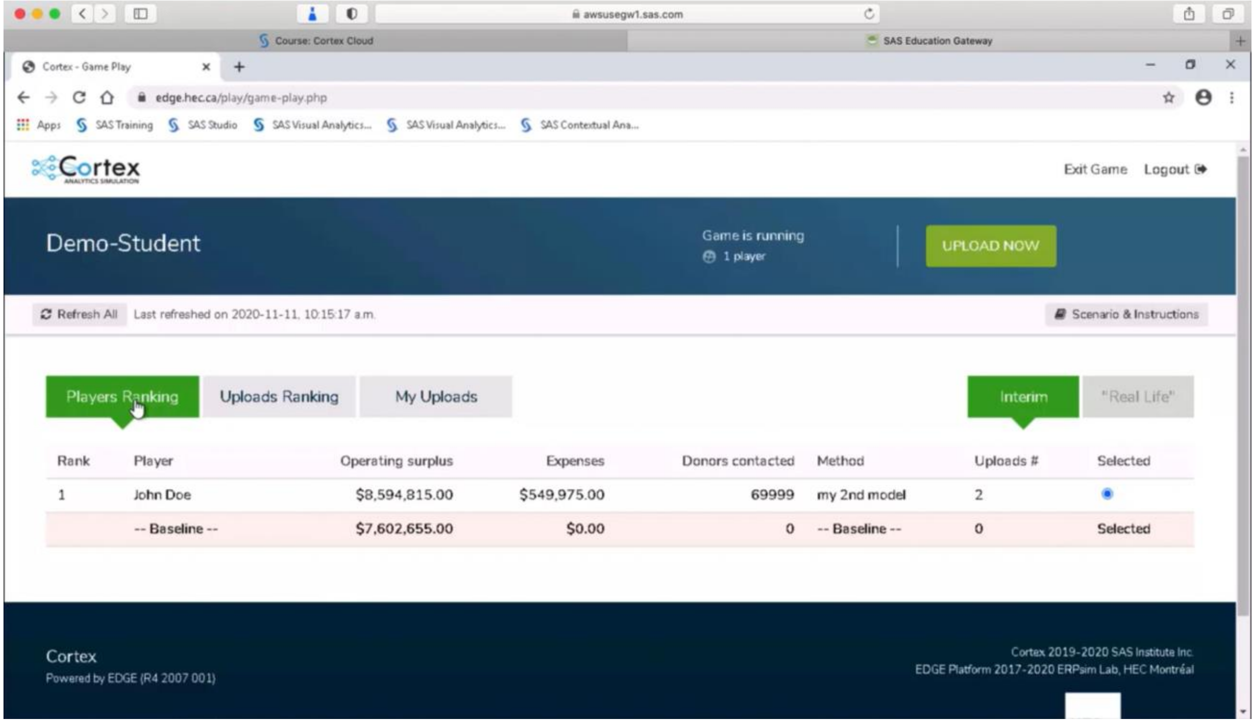
Select file

Choose File my solution1.csv

Method used

my 1rst upload

In the main leaderboard: **Players Ranking** tab, you can check the result of the model you choose to upload as well as results uploaded by other players.



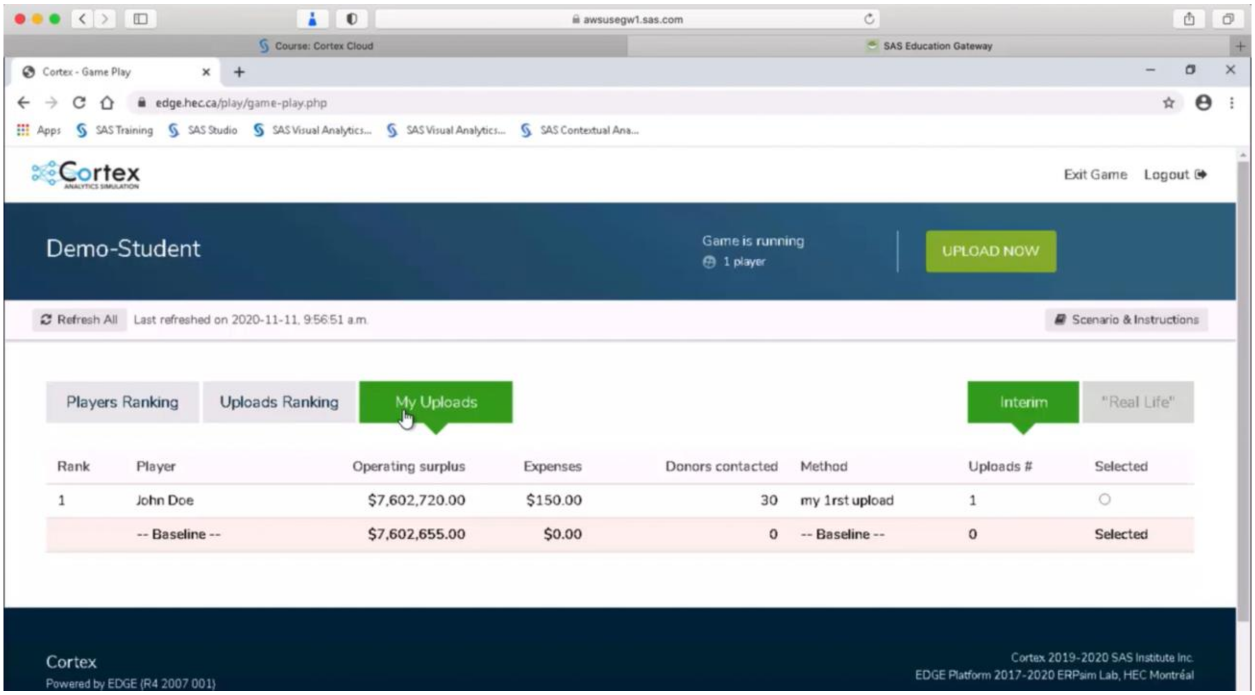
Last refreshed on 2020-11-11, 10:15:17 a.m.

Rank	Player	Operating surplus	Expenses	Donors contacted	Method	Uploads #	Selected
1	John Doe	$8,594,815.00	$549,975.00	69999	my 2nd model	2	●
	-- Baseline --	$7,602,655.00	$0.00	0	-- Baseline --	0	Selected

Cortex
Powered by EDGE (R4 2007 001)

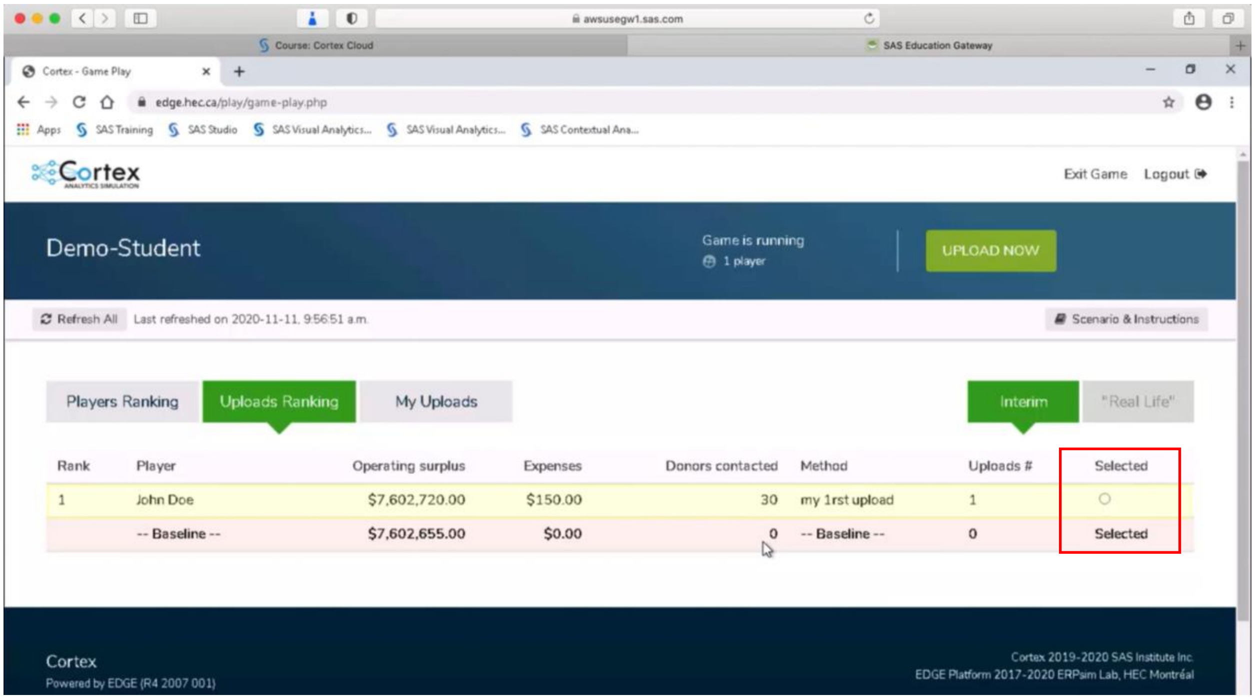
Cortex 2019-2020 SAS Institute Inc.
EDGE Platform 2017-2020 ERPsim Lab, HEC Montréal

In the **My Uploads** tab, you can see results of all your uploads in the chronological order.



Rank	Player	Operating surplus	Expenses	Donors contacted	Method	Uploads #	Selected
1	John Doe	$7,602,720.00	$150.00	30	my 1rst upload	1	○
	-- Baseline --	$7,602,655.00	$0.00	0	-- Baseline --	0	Selected

In the **Uploads Ranking** tab to see your uploads' result based on the Operating Surplus ranking. You can then **Select** one of your uploads with the highest operating surplus and upload to the main Leaderboard, where other players can also see your result.



Rank	Player	Operating surplus	Expenses	Donors contacted	Method	Uploads #	Selected
1	John Doe	$7,602,720.00	$150.00	30	my 1rst upload	1	○
	-- Baseline --	$7,602,655.00	$0.00	0	-- Baseline --	0	Selected

Round 2 - Game Summary

In Round 2, the foundation still needs help to increase the net amount of the donations. To do so, the effectiveness of the modelling approach should be improved.

One way to improve is to adopt a two-stage modelling approach as below:

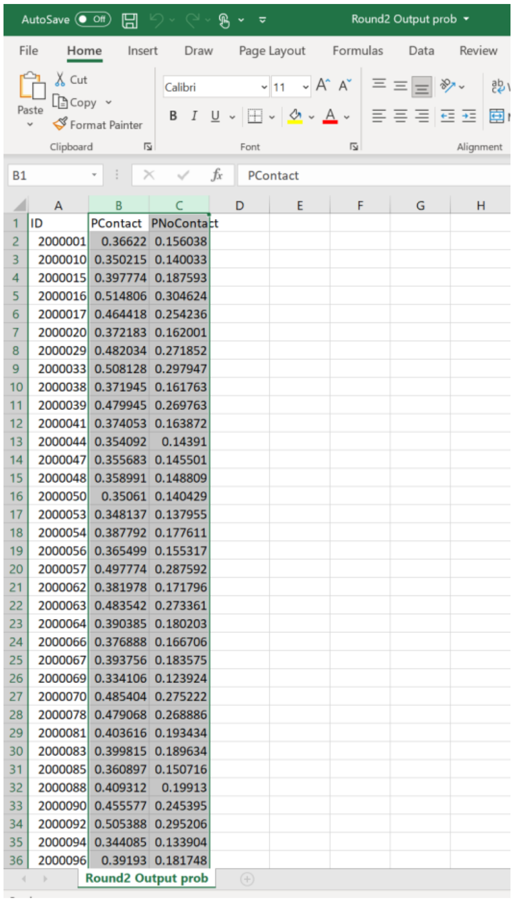
1. Fit a model to determine the probability P that an individual will give
2. Keep only the data of those who gave, fit a model for M (the amount gave)
3. Use both models to make predictions on the population
4. Compute P*M to determine the 'expected donation' of each individual

There are many approaches to 2-stage modelling, but in most cases these steps are required to calculate the value of the uplift (hence the uplift modelling):

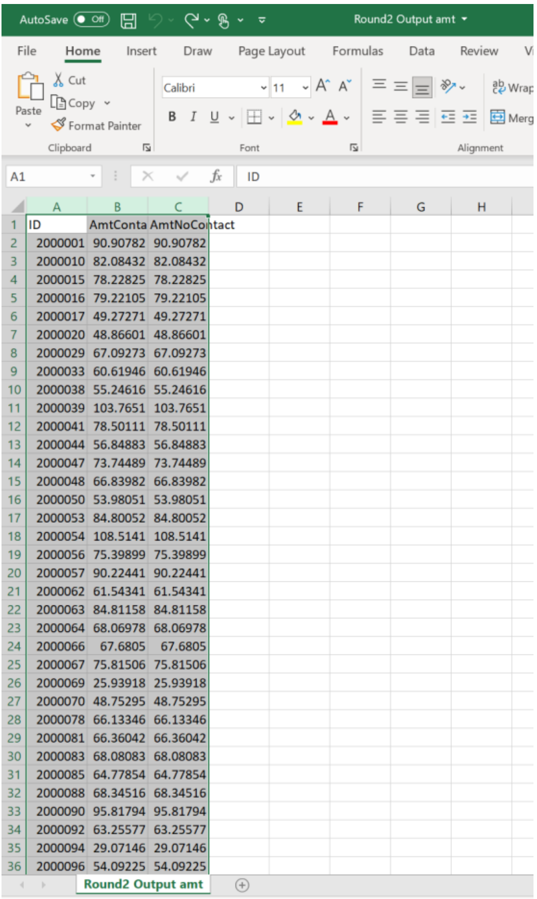
- Predict the value if a person receives a treatment (here called or contacted)
- Predict the value if a person does not receive a treatment (here not called or not contacted)
- Compute the difference between the two (i.e., the uplift generated by the treatment or targeted action: here the call)

Round 2 - Upload Decisions

Copy and paste these five columns into a new CSV file.

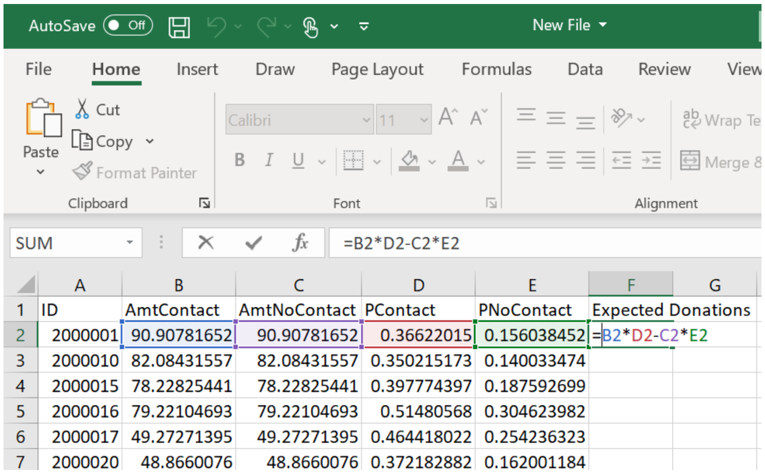


	A	B	C
1	ID	PContact	PNoContact
2	2000001	0.36622	0.156038
3	2000010	0.350215	0.140033
4	2000015	0.397774	0.187593
5	2000016	0.514806	0.304624
6	2000017	0.464418	0.254236
7	2000020	0.372183	0.162001
8	2000029	0.482034	0.271852
9	2000033	0.508128	0.297947
10	2000038	0.371945	0.161763
11	2000039	0.479945	0.269763
12	2000041	0.374053	0.163872
13	2000044	0.354092	0.14391
14	2000047	0.355683	0.145501
15	2000048	0.358991	0.148809
16	2000050	0.35061	0.140429
17	2000053	0.348137	0.137955
18	2000054	0.387792	0.177611
19	2000056	0.365499	0.155317
20	2000057	0.497774	0.287592
21	2000062	0.381978	0.171796
22	2000063	0.483542	0.273361
23	2000064	0.390385	0.180203
24	2000066	0.376888	0.166706
25	2000067	0.393756	0.183575
26	2000069	0.334106	0.123924
27	2000070	0.485404	0.275222
28	2000078	0.479068	0.268886
29	2000081	0.403616	0.193434
30	2000083	0.399815	0.189634
31	2000085	0.360897	0.150716
32	2000088	0.409312	0.19913
33	2000090	0.455577	0.245395
34	2000092	0.505388	0.295206
35	2000094	0.344085	0.133904
36	2000096	0.39193	0.181748



	A	B	C
1	ID	AmtConta	AmtNoContact
2	2000001	90.90782	90.90782
3	2000010	82.08432	82.08432
4	2000015	78.22825	78.22825
5	2000016	79.22105	79.22105
6	2000017	49.27271	49.27271
7	2000020	48.86601	48.86601
8	2000029	67.09273	67.09273
9	2000033	60.61946	60.61946
10	2000038	55.24616	55.24616
11	2000039	103.7651	103.7651
12	2000041	78.50111	78.50111
13	2000044	56.84883	56.84883
14	2000047	73.74489	73.74489
15	2000048	66.83982	66.83982
16	2000050	53.98051	53.98051
17	2000053	84.80052	84.80052
18	2000054	108.5141	108.5141
19	2000056	75.39899	75.39899
20	2000057	90.22441	90.22441
21	2000062	61.54341	61.54341
22	2000063	84.81158	84.81158
23	2000064	68.06978	68.06978
24	2000066	67.6805	67.6805
25	2000067	75.81506	75.81506
26	2000069	25.93918	25.93918
27	2000070	48.75295	48.75295
28	2000078	66.13346	66.13346
29	2000081	66.36042	66.36042
30	2000083	68.08083	68.08083
31	2000085	64.77854	64.77854
32	2000088	68.34516	68.34516
33	2000090	95.81794	95.81794
34	2000092	63.25577	63.25577
35	2000094	29.07146	29.07146
36	2000096	54.09225	54.09225

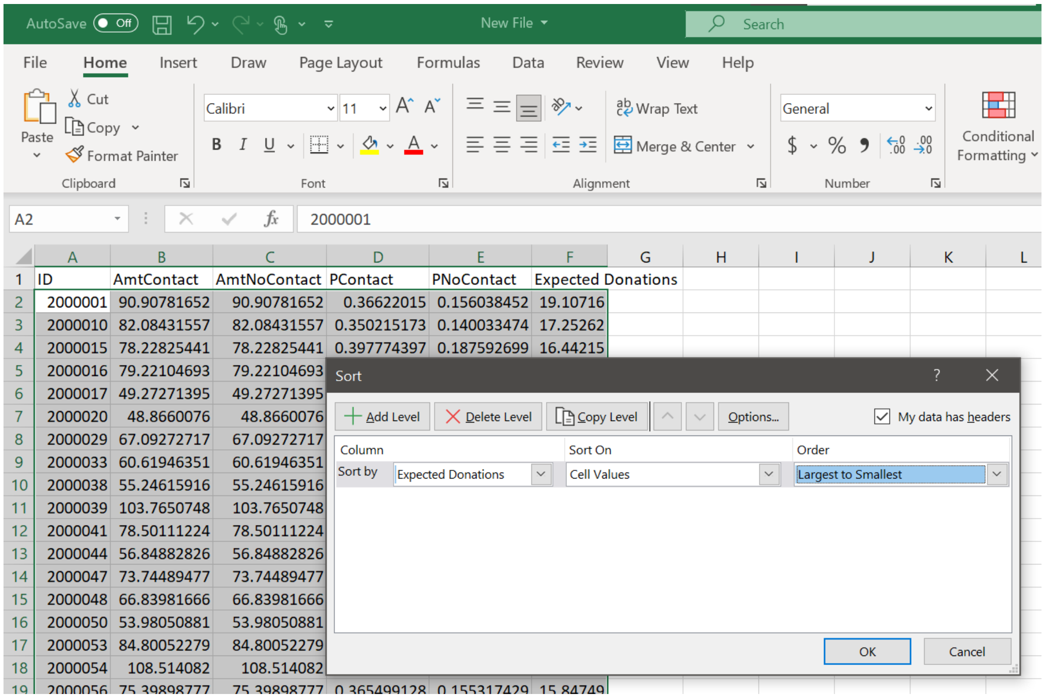
Create a new column names “Expected Donations” and calculate it.



```
=B2*D2-C2*E2
```

	A	B	C	D	E	F
1	ID	AmtContact	AmtNoContact	PContact	PNoContact	Expected Donations
2	2000001	90.90781652	90.90781652	0.36622015	0.156038452	=B2*D2-C2*E2
3	2000010	82.08431557	82.08431557	0.350215173	0.140033474	
4	2000015	78.22825441	78.22825441	0.397774397	0.187592699	
5	2000016	79.22104693	79.22104693	0.51480568	0.304623982	
6	2000017	49.27271395	49.27271395	0.464418022	0.254236323	
7	2000020	48.8660076	48.8660076	0.372182882	0.162001184	

Sort all columns by descending amount of Expected Donations.



A2 2000001

Sort

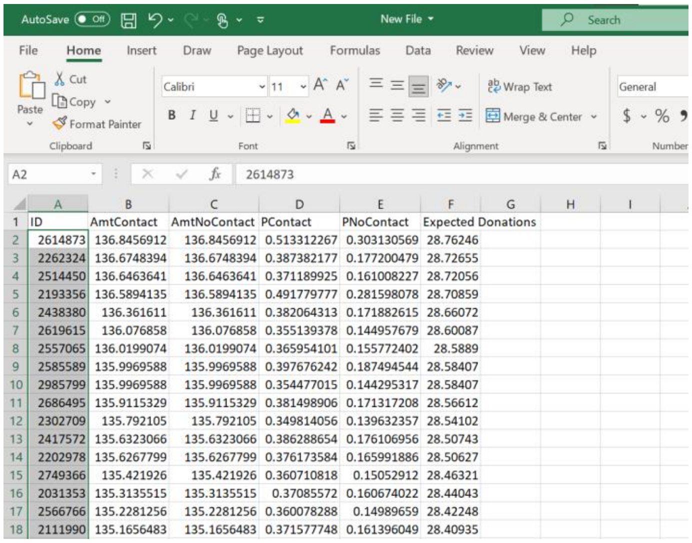
☑ My data has headers

Column	Sort On	Order
Sort by Expected Donations	Cell Values	Largest to Smallest

	A	B	C	D	E	F
1	ID	AmtContact	AmtNoContact	PContact	PNoContact	Expected Donations
2	2000001	90.90781652	90.90781652	0.36622015	0.156038452	19.10716
3	2000010	82.08431557	82.08431557	0.350215173	0.140033474	17.25262
4	2000015	78.22825441	78.22825441	0.397774397	0.187592699	16.44215
5	2000016	79.22104693	79.22104693			
6	2000017	49.27271395	49.27271395			
7	2000020	48.8660076	48.8660076			
8	2000029	67.09272717	67.09272717			
9	2000033	60.61946351	60.61946351			
10	2000038	55.24615916	55.24615916			
11	2000039	103.7650748	103.7650748			
12	2000041	78.50111224	78.50111224			
13	2000044	56.84882826	56.84882826			
14	2000047	73.74489477	73.74489477			
15	2000048	66.83981666	66.83981666			
16	2000050	53.98050881	53.98050881			
17	2000053	84.80052279	84.80052279			
18	2000054	108.514082	108.514082			
19	2000056	75.39898777	75.39898777	0.365499128	0.155317429	15.84749

Select the number of people ID you want to contact and keep this information in the CSV file for upload.

ONLY keep ID numbers in the CSV file for upload.



A2 2614873

	A	B	C	D	E	F
1	ID	AmtContact	AmtNoContact	PContact	PNoContact	Expected Donations
2	2614873	136.8456912	136.8456912	0.513312267	0.303130569	28.76246
3	2262324	136.6748394	136.6748394	0.387382177	0.177200479	28.72655
4	2514450	136.6463641	136.6463641	0.371189925	0.161008227	28.72056
5	2193356	136.5894135	136.5894135	0.491779777	0.281598078	28.70859
6	2438380	136.361611	136.361611	0.382064313	0.171882615	28.66072
7	2619615	136.076858	136.076858	0.355139378	0.144957679	28.60087
8	2557065	136.0199074	136.0199074	0.365954101	0.155772402	28.5889
9	2585589	135.9969588	135.9969588	0.397676242	0.187494544	28.58407
10	2985799	135.9969588	135.9969588	0.354477015	0.144295317	28.58407
11	2686495	135.9115329	135.9115329	0.381498906	0.171317208	28.56612
12	2302709	135.792105	135.792105	0.349814056	0.139632357	28.54102
13	2417572	135.6323066	135.6323066	0.386288654	0.176106956	28.50743
14	2202978	135.6267799	135.6267799	0.376173584	0.165991886	28.50627
15	2749366	135.421926	135.421926	0.360710818	0.15052912	28.46321
16	2031353	135.3135515	135.3135515	0.37085572	0.160674022	28.44043
17	2566766	135.2281256	135.2281256	0.360078288	0.14989659	28.42248
18	2111990	135.1656483	135.1656483	0.371577748	0.161396049	28.40935

Then you can follow the same step mentioned in Round 1 to upload your model.

DATASET MODIFICATION:

DATA EXPLORATION: Learn how to explore the data and the variables in the dataset, which will help you with improving your models and predictions.

MODEL COMPARISON: When adding models or changing the properties of existing models, check the results of the 'Model Comparison' node to see if those changes were effective and which model performed best.

REMOVING VARIABLES: You can choose to remove some of the variables from your models. For example, if a variable has too many missing values, you might decide not to consider it as part of your models.

TRANSFORM VARIABLES: For some models, the geometry of the data is important. If that's the case, transforming some variables to make them more symmetrical can improve performance.

REPLACING MISSING VARIABS: In this dataset, some data is coded as missing, but this is the donation history when an individual has never donated. These values are not actually missing. They are from an extraction that found no record of donation.

For example, if an individual never donated, the total donations for that individual are 0.

We can therefore replace the values coded as missing with their true value (e.g., replace blank with zero).