Paper SAS3528-2019

Time Travel into the Future of Clinical Trial Enrollment Design

Bahar Biller, Anup Mokashi, Ivan Oliveira, Sharmin Pathan, Jinxin Yi, and Jim Box,
SAS Institute Inc., Cary, NC

ABSTRACT

How might the clinical trial that you design today perform in the future? Will your design enable you to reach the target patient enrollment fast enough while staying within budget? This paper shows how SAS® Clinical Trial Enrollment Simulator, built on SAS® Optimization software and integrated with machine learning, gives pharmaceutical companies and clinical research organizations the power to predict the future performance of their clinical trial enrollment designs in real time. Furthermore, combined use of simulation, machine learning, and optimization creates the option to deploy enrollment simulations in real-time analytical portals. A numerical example is used to describe the development of a clinical trial enrollment process simulation and to outline the steps of using output data generated from this simulation to instantaneously predict patient enrollment for any given scenario and to recommend site activations to deliver target enrollment with high confidence.

INTRODUCTION

SAS recognizes how critical it is for clinical research organizations and pharmaceutical companies to have access to strategic decision-support tools to design better patient enrollment plans and accurately estimate cost. Cognizant (2015) reports that 80% of clinical trials fail to meet enrollment timelines and that one-third of phase III clinical trial study terminations stem from poor patient enrollment planning. Often the problem is caused by the lack of accuracy in gauging the time that it takes to reach target patient enrollment and in estimating the total cost of starting clinical research efforts in new countries, activating clinical research sites, and screening and enrolling patients in clinical trials. All of this leads to delays in getting medicines to the market and can result in significant budget shortfalls.

This paper presents the SAS technology that will help you overcome the three primary challenges of clinical trial enrollment planning (Handelsman 2012):

- The patient enrollment process consists of a long sequence of dynamic events.
- The hierarchical relationship among country startups, site activations, and patient screening and enrollment complicates the process of design and analysis of patient enrollment.
- Enrollment planning must be driven by country, site, and patient data sets, and the solution must be robust to the data uncertainty and scalable to any number of countries and sites.

SAS Clinical Trial Enrollment Simulator addresses clinical trial enrollment planning questions for SAS customers. It is made available through a web interface as software as a service. This paper has two objectives:

- to introduce you to SAS® Simulation Studio of SAS Optimization software and showcase how this technology enables you to create flexible, scalable, data-driven discrete-event stochastic simulation models of clinical trial enrollment processes

- to demonstrate how an integrated use of SAS Simulation Studio with machine learning and optimization will enable you to almost immediately make patient enrollment predictions and site-selection recommendations

You will learn how to exploit scalable and data-driven discrete-event stochastic simulation models of SAS Simulation Studio to develop risk-sensitive enrollment plans. Furthermore, you will be equipped with the capability to perform fast scenario analysis and make real-time site-selection decisions.

AN ILLUSTRATIVE CLINICAL TRIAL ENROLLMENT PROCESS

This section discusses how you can use SAS Simulation Studio to represent the illustrated process flow. As a realistic representation of a clinical trial enrollment process, consider the illustration in Figure 1, which demonstrates the need to estimate the number of patients who can be enrolled in a clinical trial within a specified time horizon (such as within the next 12 months). This time-dependent key performance indicator (KPI) is denoted throughout the paper as *Y(t)* to represent the total number patients enrolled in the clinical trial—summed across all site enrollments—by time *t* (measured in days).

There are three consecutive events that represent the clinical trial timeline and contribute to the construction of a risk profile for *Y(t)*:

- starting clinical research efforts in a country
- activating the clinical research sites in a country
- enrolling and tracking patients who arrive at each site

These events connect through a sequence of random subprocesses: country startup delay; site identification delay; site activation delay; site enrollment capacity; arrivals of patients to the site; and finally, the screening of each patient, which might result in the enrollment of the

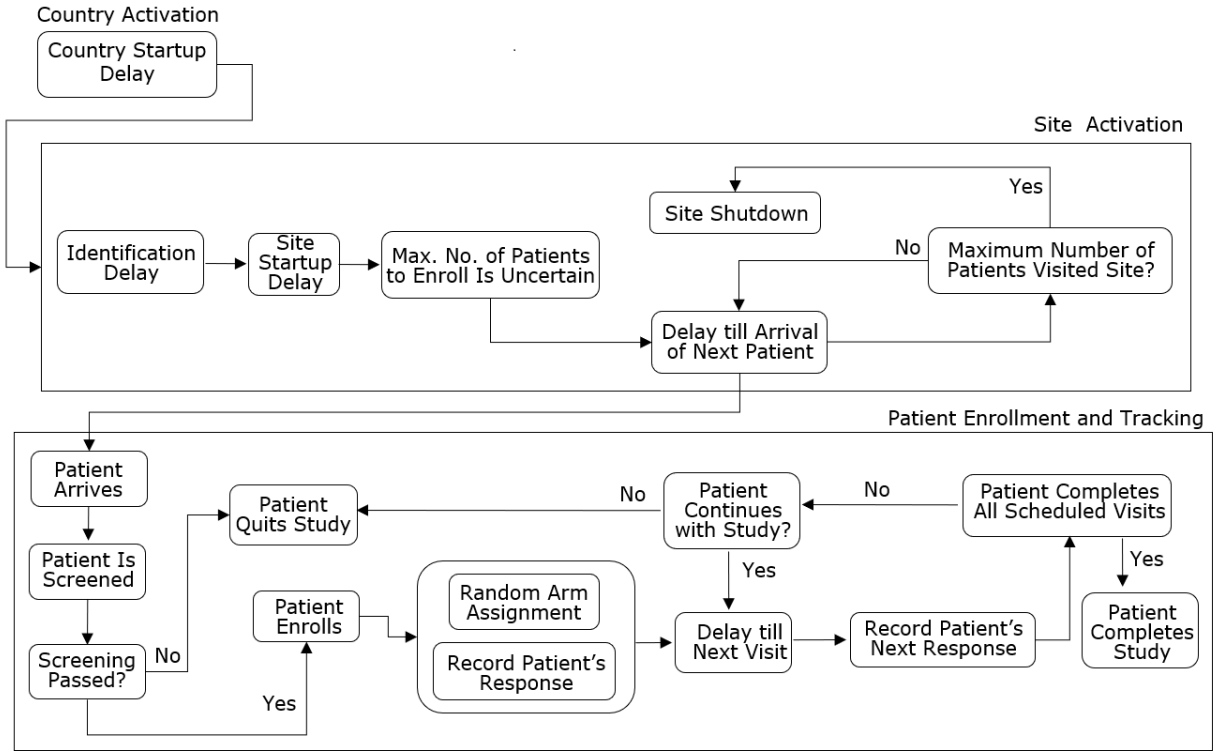


Figure 1. A High-Level View of Clinical Trial Enrollment Process Flow

patient in the clinical trial. Notice that a patient is enrolled in the clinical trial after passing the screening and before the execution of the patient's response model and the visit schedule. Therefore, generation of any patient-specific data from the patient's response model and visit schedule takes place after the KPI of interest in this paper is updated. It is for this reason that the focus is naturally on country-specific and site-specific events.

The software that is used to create the clinical trial enrollment process flow in Figure 1 is SAS Simulation Studio, which is a Java-based application for building and working with discrete-event simulation models (Hughes, Pratt, and Biller 2018). SAS Simulation Studio models dynamic system operations as a discrete sequence of events, each of which occurs at a specific point in time and triggers a change in system state. Furthermore, objects move within the discrete-event simulation as entities. The three types of entities present in clinical trials are listed in Table 1 along with their attributes.

Entity	Attributes
Country	Startup delay Number of sites
Site	Startup delay Identification delay Site enrollment capacity Site patient enrollment rate Patient screening failure probability
Patient	Arm identification Patient arm assignment Patient site visit schedule Patient dropout probability

Table 1. Clinical Trial Enrollment Simulation Entities and Attributes

In the planning phase, there is uncertainty about country-specific startup delay, site-specific startup and identification delays, enrollment capacity, patient enrollment rate, and patient screening failure probability listed in Table 1. Information is provided by expert users for minimum, most likely, and maximum values for each country and site attributes. Table 2 lists these attributes.

Source of Uncertainty	Expert Opinion 1	Expert Opinion 2	Expert Opinion 3
Country startup delay	Minimum	Most likely	Maximum
Site startup delay	Minimum	Most likely	Maximum
Identification delay	Minimum	Most likely	Maximum
Enrollment capacity	Minimum	Most likely	Maximum
Site enrollment rate	Minimum	Most likely	Maximum
Screen failure probability	Minimum	Most likely	Maximum

Table 2. Information Elicited from Expert Users for Enrollment Simulation Inputs

SAS considers any stochastic simulation to consist of system logic and simulation inputs. For a clinical trial enrollment simulation, the process flow in Figure 1 plays the role of the system logic, and the information in Table 2 leads to the construction of the probabilistic models to represent the simulation inputs. Sampling realizations of system inputs and applying the system logic in SAS Simulation Studio enable you to generate predictions of KPIs, such as the number of patients who are expected to enroll in the clinical trial within the next sponsor-specified number of days. Traditional simulation output analysis quantifies the uncertainty

about the values that are predicted for enrollment within a given time horizon. Finally, integration with machine learning and optimization enables fast scenario analysis and real-time site selection to attain a target patient enrollment within the minimum number of days.

As an illustrative example, this paper presents an example application where a single country with 10 sites is considered. The inputs of Table 3 describe uncertainty for the numerical example at the most basic level.

Simulation Inputs	Characterization	Simulation Inputs	Characterization
Country startup delay	TRI (5,10,30)	Site 1 startup delay	TRI (90,105,120)
Identification delay	TRI (0,5,15)	Site 2 startup delay	TRI (30,45,60)
Enrollment capacity	TRI (1,150,300)	Site 3 startup delay	TRI (90,105,120)
Screen failure chance	TRI (0.00,0.15,0.30)	Site 4 startup delay	TRI (120,135,150)
Site 1 enrollment rate	TRI (0.25,0.50,0.75)	Site 5 startup delay	TRI (150,180,190)
Site 2 enrollment rate	TRI (0.15,0.30,0.45)	Site 6 startup delay	TRI (90,120,150)
Site 3 enrollment rate	TRI (0.20,0.40,0.60)	Site 7 startup delay	TRI (90,120,150)
Site 4 enrollment rate	TRI (0.25,0.50,0.75)	Site 8 startup delay	TRI (75,90,105)
Site 5 enrollment rate	TRI (0.35,0.70,1.05)	Site 9 startup delay	TRI (150,180,210)
Site 6 enrollment rate	TRI (0.15,0.30,0.45)	Site 10 startup delay	TRI (75,90,105)
Site 7 enrollment rate	TRI (0.35,0.70,1.05)	Site 9 enrollment rate	TRI (0.40,0.80,1.20)
Site 8 enrollment rate	TRI (0.25,0.50,0.75)	Site 10 enrollment rate	TRI (0.35,0.70,1.05)

Table 3. Representative Numerical Example: Simulation Inputs (unit of time: day)

As is the common practice, the three-parameter triangular distribution (denoted by TRI in Table 3) is used to capture the uncertainty associated with each of the six sources of randomness (Elkins et al. 2007). Despite Table 2 having only six entries, note that those starting from the third row are repeated for each of the 10 countries, resulting in 51 different stochastic inputs to be modeled for the single-country, 10-site setting. The following questions are to be answered via SAS Simulation Studio and its integrated use with machine learning and optimization:

1. Under the country, site, and patient assumptions of Table 3, how many patients could be enrolled in this clinical trial within the next 12 months, and how much risk is in this prediction?
2. Which of the stochastic inputs in Table 3 has the highest impact on the mean enrollment?
3. How fast would patient enrollment increase with time, and what would be the value of patient enrollment at a specific point in time and at specific values of the simulation inputs in Table 3?
4. What is the optimal set of sites to activate in order to enroll at least φ patients in the minimum amount of time?

Answering these questions can be challenging, especially under high levels of uncertainty. The rest of the paper demonstrates the power of SAS Simulation Studio to help you find answers. For the first time, you will also see the most recent SAS technology for scenario analysis and site selection in real time so that you are better equipped to answer your sponsor's what-if questions almost immediately. First, the paper describes how you can use SAS Simulation Studio for KPI generation and uncertainty quantification to answer the first two questions. Then, it builds on machine learning and optimization to address the last two questions and deliver on-demand performance prediction and site selection.

KPI GENERATION, RISK CALCULATION, AND SENSITIVITY ANALYSIS

CLINICAL TRIAL ENROLLMENT SIMULATION: LOGIC AND INPUTS

SAS Simulation Studio captures the logic in Figure 1 by using the entities and attributes shown in Table 1 in drag-and-drop construction. The power of SAS Simulation Studio stems from scalable, data-driven, flexible modeling of dynamic and stochastic systems by building on this construction method. Figure 2 illustrates a simplified view of how the blocks available in SAS Simulation Studio for drag-and-drop construction lead to a clinical trial enrollment simulation model and experimental design view. The scalability, data-driven nature, and flexibility of this simulation model are discussed next.

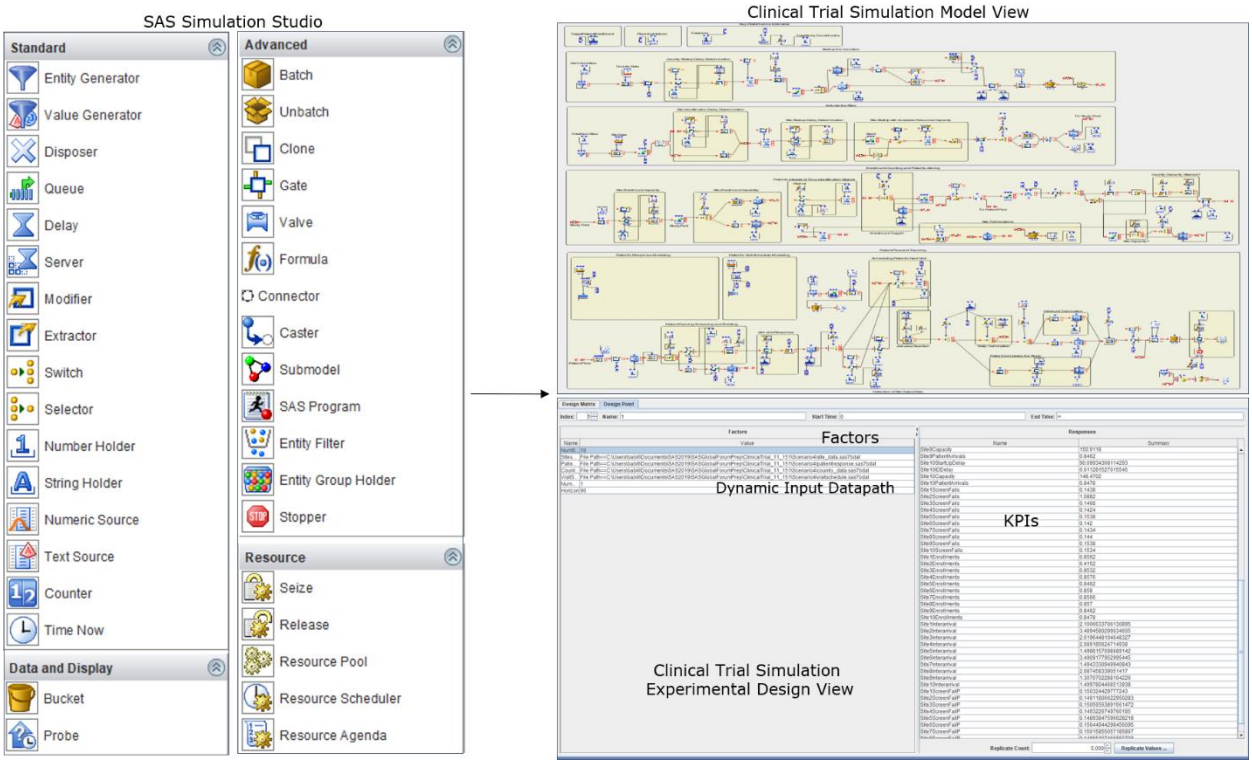


Figure 2. SAS Simulation Studio: Blocks, Model Design, and Experimental Design

- SAS simulation of clinical trial enrollment planning is *scalable* because it gives you the full power to choose any number of countries and any number of sites, each with its own patient enrollment model, without making any changes to the logic of the existing simulation. This is because SAS Simulation Studio reads country and site characterizations from SAS data tables. In the numerical use case, there is a single country with 10 sites. Therefore, the country data set contains a single row, and the site data set contains 10 rows. If you want to use the simulation for 200 different sites, all you have to do is update the country and site SAS data sets without changing the simulation logic (that is, the model view of SAS Simulation Studio in Figure 2). This is how SAS Simulation Studio enables you to build scalable simulations.
- SAS clinical trial enrollment simulation is *data-driven* because it enables you to dynamically create input data paths and store the clinical trial enrollment input and output data in the SAS data tables. As illustrated in Figure 2, you can specify the path to each country, site, and patient data set as a factor in the experimental design window. This enables you to readily change the content of the input data outside the simulation by directly replacing existing sets of input data with different data sets that you might want

to experiment with. Furthermore, you can easily change the location that the input data are read from; this aspect of the simulation design is treated independently from the simulation model and simulation output analysis. Finally, input data are not necessarily read at the beginning of the simulation. The input data can be called into the simulation logic at any time during the simulation run.

- SAS clinical trial enrollment simulation is *flexible* because the modular model development of SAS Simulation Studio makes it possible for you to easily incorporate the changes to the process timeline through drag-and-drop construction, frequently without any impact on a significant portion of the existing model. Often, modifying a portion of the process flow can cause gridlock during a simulation run. But because of the nonblocking queue block—unique to SAS Simulation Studio—this is almost never a concern. Thus, the clinical trial enrollment model, developed in SAS Simulation Studio, is entirely flexible and plays a key role during the validation phase of the underlying discrete-event simulation.

After you use its drag-and-drop functionality to construct the patient enrollment simulation logic, SAS Simulation Studio captures the uncertainty in the inputs that drive the process logic. It propagates input uncertainty through the simulation during execution and quantifies the impact of the input uncertainty in the KPIs through confidence intervals and risk profiles that are obtained from many potential future sample paths of the clinical trial enrollment plans. Because SAS Simulation Studio can perform automated collection of output data for experimental design of any size and can store the resulting simulation outputs in SAS data tables, you can conduct extensive statistical output analyses and learn from the enrollment simulation outputs.

ANALYZING OUTPUTS FOR KPI GENERATION AND RISK QUANTIFICATION

When researchers weigh the benefits of using discrete-event simulations for studying complex and dynamic stochastic systems, such as clinical trial patient enrollment processes, the emphasis is usually on being able to represent the system behavior as it is. However, any discrete-event simulation is foremost a data generation program to answer questions 1 and 2 earlier in the paper. Figure 3 displays the results of using simulation-generated output data to predict the KPI and to analyze the risk in KPI prediction and sensitivity of mean KPI to the means of inputs.

With the KPI chosen as the number of patients enrolled in the clinical trial within the next 12 months, 5,000 independent replications of the clinical trial enrollment simulation are performed; the resulting 5,000 rows and 52 columns of data are stored as the simulation output data in SAS data tables. There are 5,000 rows of simulation outputs because 5,000 replications of the clinical trial enrollment simulation were performed. There are 52 rows of data because one column stores the realizations of the KPI (one per replication); one column stores the realizations of the country startup delays that are generated from the triangular distribution with a minimum of 5 days, a most likely value of 10 days, and a maximum of 30 days; and the remaining 50 columns store the realizations of the site-specific inputs shown in Table 3. This is the simulation-generated output data set that has been statistically analyzed by JMP® Pro software and results in the simulation output analysis presented in Figure 3.

A close look at the KPI risk profile that is displayed in the left-hand side of Figure 3 reveals an expected enrollment of 889 patients. This average prediction of patient enrollment is further estimated to fall between 887 patients and 892 patients with a 95% probability. There is also a 10% chance of enrolling fewer than 780 patients and a predicted 2.5% chance that 1,054 or more patients might be enrolled in the study within the next 12 months. This completes the answer to the first question and presents an example of the risk analysis that you can carry out using SAS Simulation Studio and JMP Pro software for enrollment planning.

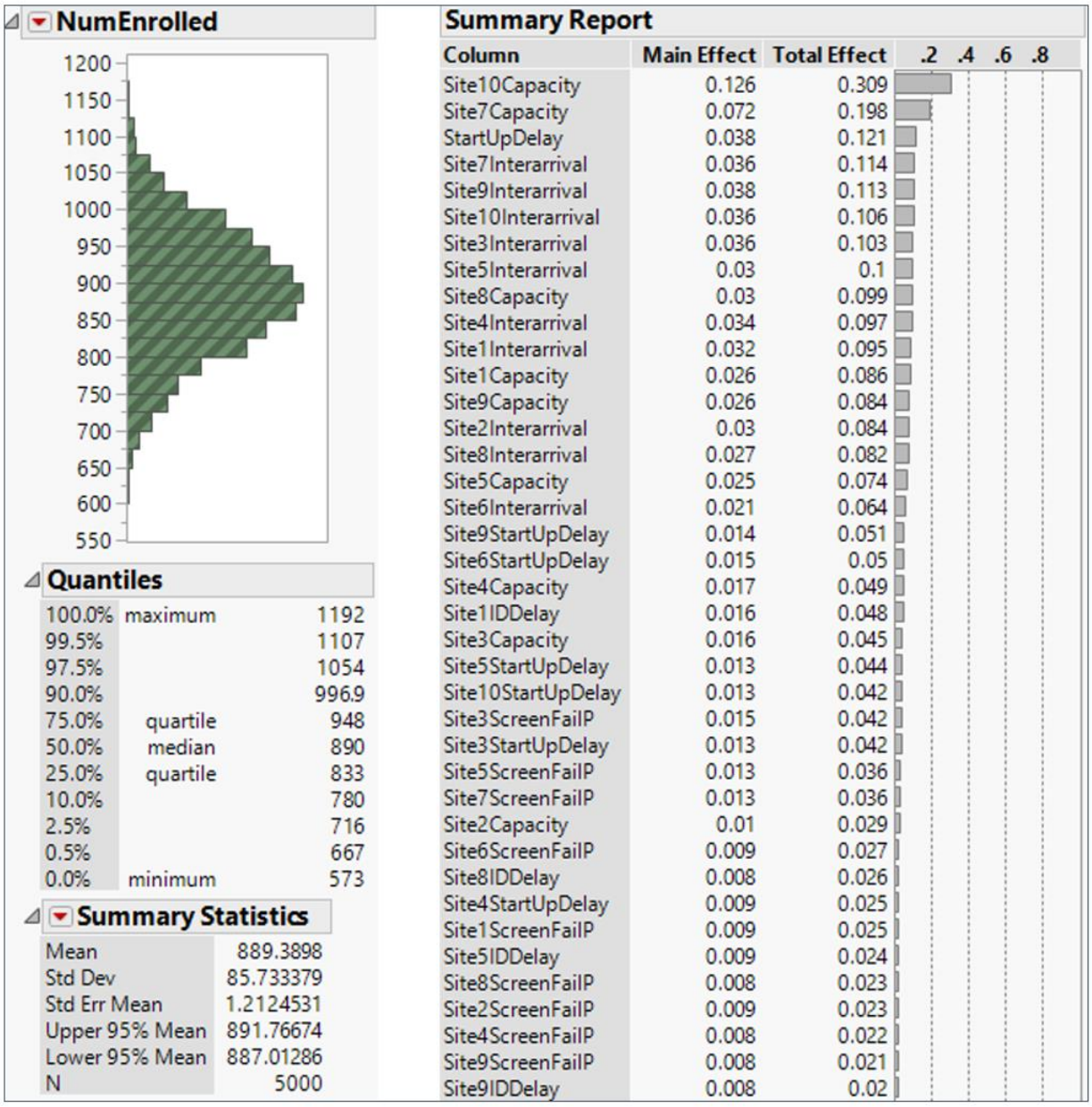


Figure 3. Clinical Trial Enrollment Simulation Output Analysis (Unit of Time: Day)

MEASURING SENSITIVITY OF MEAN KPI TO SIMULATION INPUTS AND TIME

The right-hand side of Figure 3 displays results of the sensitivity analysis and answers the second question: the site enrollment capacity, especially at Site 7 and Site 10, appears to be the input with the highest impact on the mean number of patients who can be enrolled in the clinical trial within the next 12 months. However, what is missing from the output analysis in Figure 3 is how the patient enrollment changes with time. Figure 3 considers only a planning horizon of 12 months. If you start with a planning horizon of 3 months, you can further analyze a simulation output data set obtained from, for example, 34 different scenarios in which each scenario corresponds to a different horizon length. Consequently, you would be analyzing a 53-column simulation output data set with 170,000 (=34*5000) rows and obtain a plot with patient enrollment on the Y axis, time in months on the X axis, and each curve corresponding to a value of the quantile. An example of such a plot is shown in Figure 4.

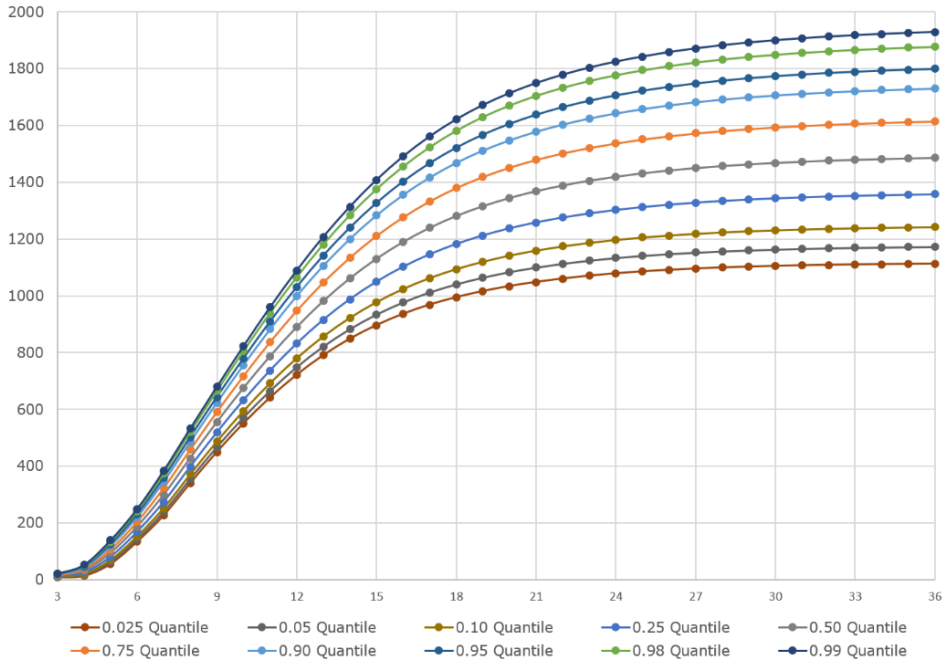


Figure 4. Sensitivity of Patient Enrollment Quantiles (Y Axis) to the Time (X Axis)

USING MACHINE LEARNING FOR FAST SCENARIO ANALYSIS

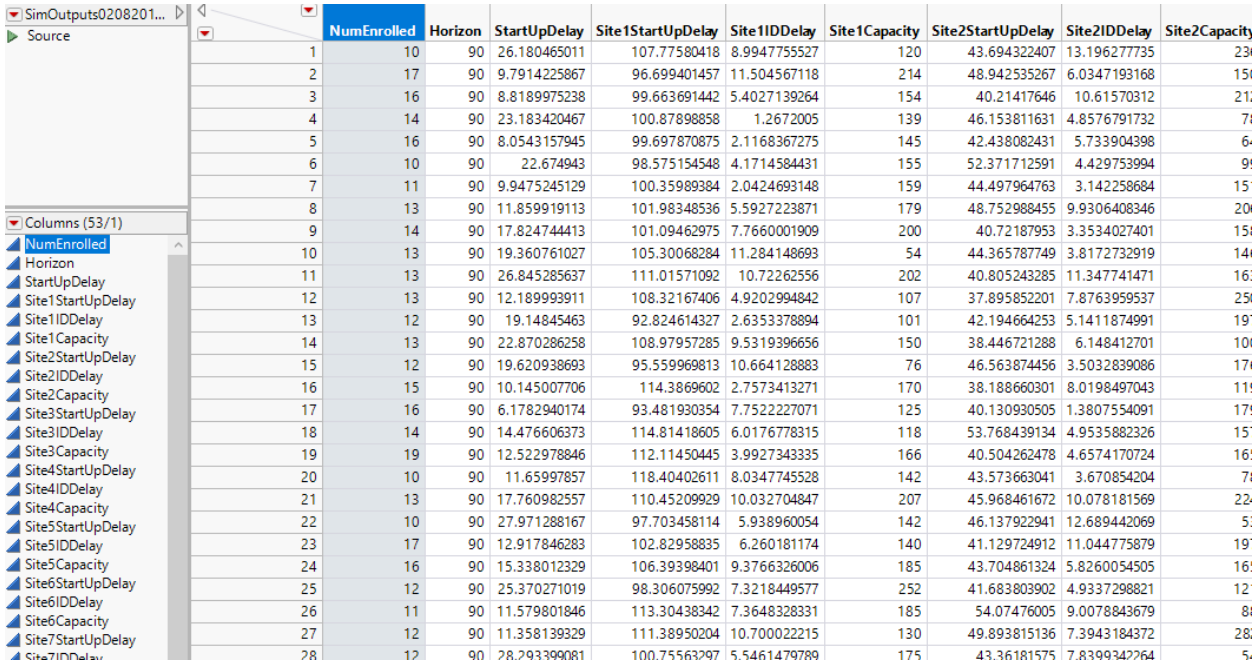
Despite such a comprehensive analysis of the risk in strategic patient enrollment planning, the results displayed in Figures 3 and 4 might not be sufficient to answer what-if questions in real time. For example, your sponsor might define a completely new scenario—say, Scenario 1—which you have not simulated and which you have not used for any of the analyses reported in Figures 3 and 4. Here are the details of Scenario 1:

- All simulation inputs except site enrollment capacity and site-dependent patient screening probability are assumed to be equal to the average values of probability distributions tabulated in Table 3. For example, Scenario 1 sets the country startup delay to 15 days; this is the average of 5 days, 10 days, and 30 days in Table 3, where these three numerical values represent minimum, most likely, and maximum values for country startup delay, as described in Table 2.
- Enrollment capacity and screening probability are 200 patients and 20%, respectively.
- The sponsor wants to know the number of patients who might enroll within 13.5 months.

The objective is twofold:

- to predict the enrollment in real time for Scenario 1 that is specified by the sponsor
- to answer three additional questions, each representing a different scenario, in real time:
  - ✓ Scenario 2: What would enrollment predictions be six months later?
  - ✓ Scenario 3: What if the site enrollment capacities were reduced by 10%?
  - ✓ Scenario 4: What could push enrollment predictions to the lower quantile curves?

What is important to notice is that none of the four scenarios have been simulated in the previous section. SAS machine learning learns from the simulation outputs, which are a 170,000-row, 53-column data set shown in Figure 5, to approximate the cumulative patient enrollment by a specified time. Specifically, patient enrollment is approximated by a neural network whose response variable corresponds to 170,000 different observations of *Y(t)*—the total number of patients enrolled in the clinical trial by time *t*—recorded during the clinical trial enrollment simulation runs. This is the first column of the simulation output data set in Figure 5. The second column represents the time (in days) by which the patient enrollment is predicted. The remaining 51 columns—startup delay for one country and startup delay, identification delay, enrollment capacity, enrollment rate, and screen failure probability for each of the 10 sites—correspond to inputs of the neural network, each of which is sampled during 170,000 replications of the simulation model. The probability distributions from which these inputs are sampled in each replication of the simulation are given in Table 3.


SimOutputs0208201...
Source
Columns (53/1)
NumEnrolled
Horizon
StartUpDelay
Site1StartUpDelay
Site1IDDelay
Site1Capacity
Site2StartUpDelay
Site2IDDelay
Site2Capacity
Site3StartUpDelay
Site3IDDelay
Site3Capacity
Site4StartUpDelay
Site4IDDelay
Site4Capacity
Site5StartUpDelay
Site5IDDelay
Site5Capacity
Site6StartUpDelay
Site6IDDelay
Site6Capacity
Site7StartUpDelay

	NumEnrolled	Horizon	StartUpDelay	Site1StartUpDelay	Site1IDDelay	Site1Capacity	Site2StartUpDelay	Site2IDDelay	Site2Capacity
1	10	90	26.180465011	107.77580418	8.9947755527	120	43.694322407	13.196277735	23
2	17	90	9.7914225867	96.699401457	11.504567118	214	48.942535267	6.0347193168	15
3	16	90	8.8189975238	99.663691442	5.4027139264	154	40.21417646	10.61570312	21
4	14	90	23.183420467	100.87898858	1.2672005	139	46.153811631	4.8576791732	7
5	16	90	8.0543157945	99.697870875	2.1168367275	145	42.438082431	5.733904398	6
6	10	90	22.674943	98.575154548	4.1714584431	155	52.371712591	4.429753994	9
7	11	90	9.9475245129	100.35989384	2.0424693148	159	44.497964763	3.142258684	15
8	13	90	11.859919113	101.98348536	5.5927223871	179	48.752988455	9.9306408346	20
9	14	90	17.824744413	101.09462975	7.7660001909	200	40.72187953	3.3534027401	15
10	13	90	19.360761027	105.30068284	11.284148693	54	44.365787749	3.8172732919	14
11	13	90	26.845285637	111.01571092	10.72262556	202	40.805243285	11.347741471	16
12	13	90	12.189993911	108.32167406	4.9202994842	107	37.895852201	7.8763959537	25
13	12	90	19.14845463	92.824614327	2.6353378894	101	42.194664253	5.1411874991	19
14	13	90	22.870286258	108.97957285	9.5319396656	150	38.446721288	6.148412701	10
15	12	90	19.620938693	95.559969813	10.664128883	76	46.563874456	3.5032839086	17
16	15	90	10.145007706	114.3869602	2.7573413271	170	38.188660301	8.0198497043	11
17	16	90	6.1782940174	93.481930354	7.7522227071	125	40.130930505	1.3807554091	17
18	14	90	14.476606373	114.81418605	6.0176778315	118	53.768439134	4.9535882326	15
19	19	90	12.522978846	112.11450445	3.9927343335	166	40.504262478	4.6574170724	16
20	10	90	11.65997857	118.40402611	8.0347745528	142	43.573663041	3.670854204	7
21	13	90	17.760982557	110.45209929	10.032704847	207	45.968461672	10.078181569	22
22	10	90	27.971288167	97.703458114	5.938960054	142	46.137922941	12.689442069	5
23	17	90	12.917846283	102.82958835	6.260181174	140	41.129724912	11.044775879	19
24	16	90	15.338012329	106.39398401	9.3766326006	185	43.704861324	5.8260054505	16
25	12	90	25.370271019	98.306075992	7.3218449577	252	41.683803902	4.9337298821	12
26	11	90	11.579801846	113.30438342	7.3648328331	185	54.07476005	9.0078843679	8
27	12	90	11.358139329	111.38950204	10.700022215	130	49.893815136	7.3943184372	28
28	12	90	28.293399081	100.75563297	5.5461479789	175	43.36181575	7.8399342264	5

Figure 5. Illustration of the Simulation Output Data in JMP Pro Software

The 170,000-row simulation output data set is divided into training, validation, and test data sets (80% training, 10% validation, 10% test). The three tasks of training, validation, and testing are performed using SAS Visual Data Mining and Machine Learning through the SWAT package—a Python interface to SAS® Cloud Analytic Services (CAS). You can use the following code to carry out these steps:

```
import swat
import os
import pandas as pd
conn = swat.CAS(os.environ['CASHOST'],os.environ['CASPORT'])

df=pd.read_csv('SimulationOutputDataSet.csv')
conn.loadactionset('sampling')
conn.upload(df, casout=dict(name='data', replace=True))

res=conn.sampling.stratified(table='data', partind=True,output=dict(
casout=dict(name='data_partitioned',replace=True),copyvars='ALL',
partindname='partition'),samppct=10,samppct2=10,seed=388264836)
```

```
train=
conn.CASTable(name='data_partitioned', where='partition=0', casout='train')
valid=
conn.CASTable(name='data_partitioned', where='partition=1', casout='valid')
test=
conn.CASTable(name='data_partitioned', where='partition=2', casout='test')

inputs = [x for x in df.columns if x != 'NumEnrolled']

from dlpy.applications import *
from dlpy.model import *
from dlpy.layers import *
from swat import CAS, CASTable

model = Sequential(conn, model_table=CASTable('model', replace=True))
model.add(InputLayer(name='input'))
model.add(Dense(20, act='relu', name='dense'))
model.add(OutputLayer(act='AUTO', name='output'))

optimizer = Optimizer(algorithm=AdamSolver(learning_rate=0.001,
learning_rate_policy='step', gamma=0.9, step_size=5),
mini_batch_size=1, max_epochs=200, log_level=1)

model.fit(train, inputs=inputs, target='NumEnrolled', optimizer=optimizer,
gpu=Gpu(devices=[0]))

res=model.predict(test)

out =
pd.DataFrame(conn.CASTable(res['OutputCasTables'].Name[0]).to_frame())
test_set = pd.DataFrame(test.to_frame())
test_set['out']=out['P_NumEnrolled']

from sklearn.metrics import r2_score
r2_score(test_set['NumEnrolled'], test_set['out'])

model.deploy(path='ENTER YOUR CHOICE', output_format='table')
model.get_model_info()
```

Testing exhibits a mean square error of 951.0284 and an R-square of 99.4740%. The entire process of loading the simulation output data, training a neural network, and testing the trained model in the cloud takes a total of 3 minutes, 2.86 seconds. By using the resulting machine learning model of the patient enrollment, you can determine what the four scenarios correspond to in Figure 6 for a 20% screening probability, and you can observe and measure sensitivity as you change focus from one scenario to another. Specifically, Scenario 4 represents a reduction of 40% in enrollment capacity and pushes enrollment prediction to a lower quantile curve in Figure 6. Furthermore, you can find the neural network prediction accuracy—in comparison to enrollment simulation that estimates mean patient enrollment within ± 1 patient—in Table 4. Although each neural network prediction in Table 4 and Figure 6 is a point estimate with no quantification of risk around the prediction, you can readily address this issue by performing a Monte Carlo simulation of the neural network fit to propagate any specified level of variation in the inputs through the patient enrollment process.

Description	Scenario 1	Scenario 2	Scenario 3	Scenario 4

ML Prediction	1,102 patients	1,676 patients	1,593 patients	1,166 patients
Sim Prediction	1,117 patients	1,720 patients	1,626 patients	1,185 patients
RMSE	14.75	43.42	32.22	18.54
ML Run Time	0.0274s	0.0291s	0.0289s	0.0273s
Sim Run Time 1 Replication	6.86s	7.86s	7.62s	7.39s

Table 4. Real-Time Enrollment Predictions Obtained from Trained Neural Network and Comparison to Simulation Mean Enrollment Predictions with ± 1 Patient Error

Why not use the simulation model directly to predict the enrollments presented in Table 4? The reason is that the neural network allows for nearly instantaneous calculation of the desired metrics. This is in comparison to the 6.86 seconds that it takes to run a single replication of clinical trial enrollment simulation lasting 405 days in Scenario 1. Furthermore, Scenario 1 has been simulated for 1,000 replications to predict mean enrollment within ± 1 patient, resulting in a total run time of 5 minutes, 9.67 seconds. When you are on the phone with a client trying to win the business, you might need to quickly explore the implications of a variety of scenarios in real time. Therefore, the solution that is built on the integration of simulation and machine learning might be better suited to your need to make patient enrollment predictions in real time. Furthermore, you are now able to deploy your enrollment simulation in a real-time analytical portal. Therefore, you have an opportunity to turn this solution into a widely used analytics within your organization for enrollment planning.

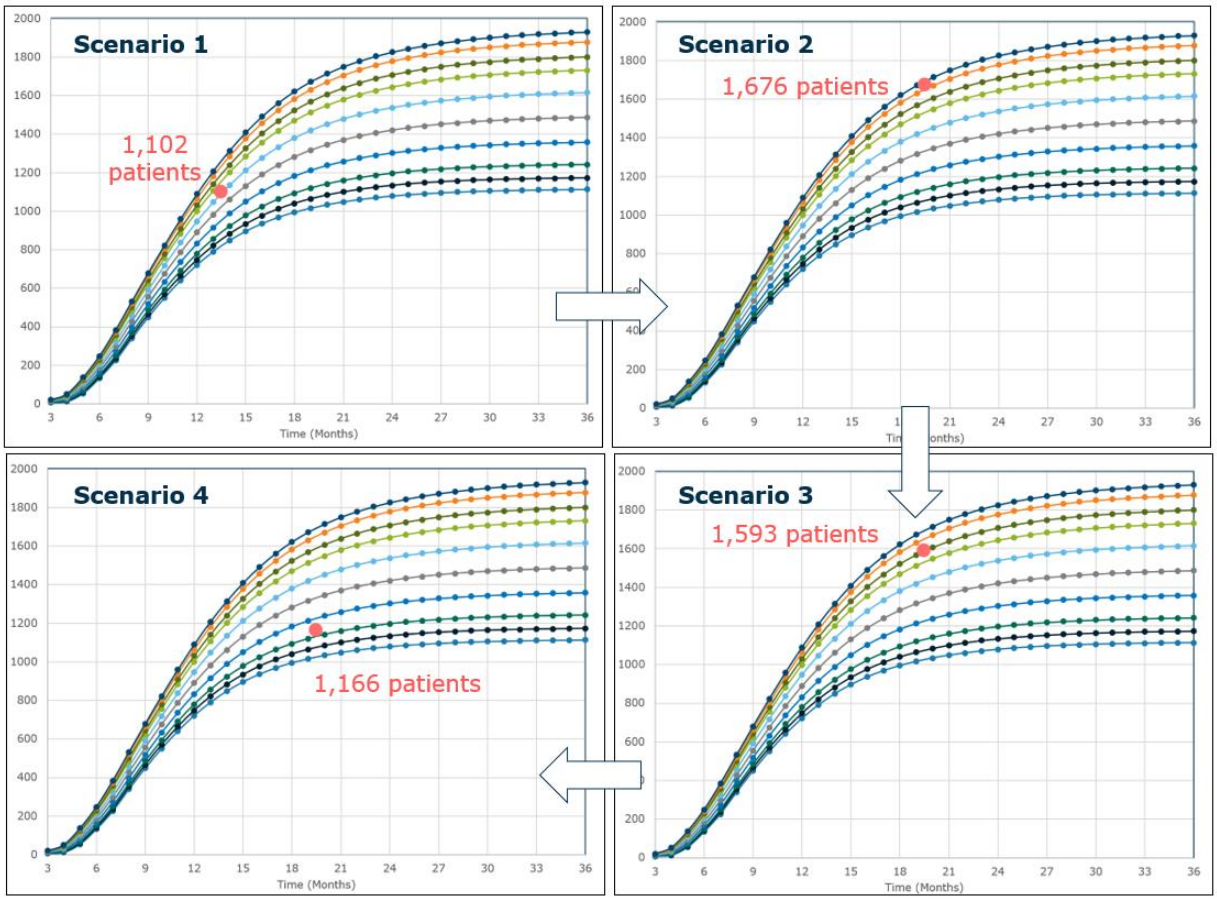


Figure 6. Visualizing Fast Scenario Analysis for 20% Screening Failure Probability

WHAT-IF ANALYSES FOR SITE SELECTION IN REAL TIME

This section considers the situation in which you might want to identify the optimal combination of the sites to activate in order to ensure that the 0.95 quantile of the total patient enrollment exceeds 800 patients in the shortest amount of time. This is a stochastic optimization problem that is formulated as follows:

1. Playing the role of a synthetic data program, the clinical trial enrollment simulation generates predictions of site-specific enrollments in each of the 5,000 replications at 34 different points in time. This results in a 170,000-row simulation output data set. A subset of this data set is used in the previous section to perform fast sensitivity analysis. This section uses another subset of the simulation-generated data set to make site-selection recommendations in real time. For each site i, the data subset contains 170,000 rows and two columns, where one column is associated with site enrollment $Y_i(t)$ and the other column is associated with time t. In this section, these data sets are used to characterize the number of patients enrolled at site i by time t, $Y_i(t)$ through its mean $E[Y_i(t)]$ and standard deviation $V^{1/2}[Y_i(t)]$.
2. Notice that the total patient enrollment $Y(t)$ is the sum of the number of patients enrolled at all activated sites. Therefore, $Y(t)$ can be alternatively written as the sum of $Y_i(t)*Z_i$, $i = 1,2,\ldots,10$, where Z_i is a decision variable that takes a value of 1 if site i is activated and 0 otherwise. The numerical example considers only 10 sites, but the number of sites involved in clinical trial enrollment planning can be significantly higher. In that case, $Y(t)$ can be represented by a normal distribution with a mean that is the sum of $E[Y_i(t)]*Z_i$, $i = 1,2,\ldots,10$, and a variance that is the sum of $V[Y_i(t)]*Z_i$, $i = 1,2,\ldots,10$. Denoting the 0.95 quantile of this characterization of $Y(t)$ as $Q(t;0.95, Z_i, i = 1,2,\ldots,10)$, the site selection problem can be formulated as the minimization of T subject to $Q(T;0.95, Z_i, i = 1,2,\ldots,10) \geq 800$ as a function of the continuous decision variable $T \geq 0$ and the binary decision variables Z_i, $i = 1,2,\ldots,10$. Thus, the identification of the optimal combination of the sites to exceed a given patient enrollment target with confidence is complicated by the nonconvex function $Q(T;0.95, Z_i, i = 1,2,\ldots,10)$ with discrete elements Z_i, $i = 1,2,\ldots,10$.

You can solve this stochastic optimization problem by integrating simulation and optimization with SAS machine learning capability. The functions are identified that best represent the mean $E[Y_i(t)]$ and the standard deviation $V^{1/2}[Y_i(t)]$ of the patient enrollment $Y_i(t)$ at site i for $i=1,2,\ldots,10$. This corresponds to the identification of 20 different function approximations for the one-country, 10-site numerical example, reducing the stochastic site selection problem to a deterministic optimization problem that you can solve using SAS Optimization.

In particular, the local search optimization algorithm enables you to solve problems that have user-defined black-box constraints such as $Q(T;0.95, Z_i, i = 1,2,\ldots,10) \geq 800$. Therefore, by using the local search optimization algorithm of SAS Optimization, you would identify the optimal action as the activation of all sites and enroll 800 patients in 306.89 days. You would also construct an efficient frontier for a range of values for the target patient enrollment (e.g., [20,1,800], including the target of 800 patients for the numerical example) on the Y axis and the corresponding optimal objective function values for the enrollment time on the X axis. You can use the following code to achieve these objectives:

```
proc optmodel printlevel=0;
  set <num> SITES;
  set <num> WEIGHTS;
  read data weightData into WEIGHTS=[r];

  number A{SITES, WEIGHTS};
  read data meanW into SITES=[i=_N_] {j in WEIGHTS} <A[i,j]=col('A'||j)>;
  print A;
```

```
  number B{SITES, WEIGHTS};
  read data stdDevW into SITES=[i=_N_] {j in WEIGHTS} <B[i,j]=col('B'||j)>;
  print B;

  var Z {SITES} binary;
  var T >= 90;
  min f = T;

  num alpha = 0.95;
  num target;
  con c1: sum {i in SITES} (Z[i]*(A[i,1]
  + sum {k in 2..8 by 3} A[i,k]*(TanH((0.5*(A[i,k+1] + A[i,k+2]*T))))))
  + probit(alpha)*SQRT(sum {i in SITES} (Z[i]*B[i,1]
  + sum {k in 2..8 by 3} B[i,k]*(TanH((0.5*(B[i,k+1] + B[i,k+2]*T)))))**2))
  >= target;

  set TARGETSET = 20 to 1800 by 5;
  num optimalT {TARGETSET};
  do target = TARGETSET;
   put target=;
   solve with lso / popsize=100 feastol=1e-6 absfconv=0 nabsfconv=100
   maxgen=100;
   solve with nlp relaxint;
   optimalT[target] = T;
  end;
  create data optdata from [target] optimalT;
quit;

proc sgplot data=optdata;
  scatter x=optimalT y=target;
  xaxis label='Time (Days)';
  yaxis label='Target Patient Enrollment';
run;
```

In this code, meanW and stdDevW are the two primary data sets that are read into the formulation of the site-selection optimization problem. Each row of these data sets represents one of the 10 sites. The columns of meanW store the site-specific weights that are used to characterize the approximation to the functions $E[Y_i(t)]$, i=1,2,…,10. The columns of stdDevW, on the other hand, capture the site-specific weights that are used to characterize the approximation to the functions $V^{1/2}[Y_i(t)]$, i=1,2,…,10. Figure 6 presents the contents of these data sets as displayed by the code.

SAS Optimization can solve the site-selection optimization problem from clients other than SAS, e.g., from Python through the SWAT package. In that case, you need to load the data sets weightData, meanW and stdDevW into CAS from the comma-separated-value (CSV) files (weightData.csv, meanW.csv, and stdDevW.csv). You can use the following code to identify the optimal combination of the sites to activate in order to ensure that the 0.95 quantile of the total patient enrollment exceeds 800 patients in the shortest amount of time:

```
import swat
import os
import pandas as pd
conn = swat.CAS(os.environ['CASHOST'],os.environ['CASPORT'])

conn.upload_file('weightData.csv')
```

```
conn.upload_file('meanW.csv')
conn.upload_file('stdDevW.csv')

conn.loadActionSet(actionset="optimization")
conn.runOptmodel(
   code="""
   set <num> SITES;
   set <num> WEIGHTS;
   read data weightdata into WEIGHTS=[r];

   number A{SITES, WEIGHTS};
   read data meanw into SITES=[i=_N_] {j in WEIGHTS} <A[i,j]=col('A'||j)>;

   number B{SITES, WEIGHTS};
   read data stdDevw into SITES=[i=_N_] {j in WEIGHTS}
<B[i,j]=col('B'||j)>;

   var Z {SITES} binary;
   var T >= 90;
   min f = T;

   num alpha = 0.95;
   num target;
   con c1: sum {i in SITES} (Z[i]*(A[i,1]+sum {k in 2..8 by 3}
A[i,k]*(TanH((0.5*(A[i,k+1] + A[i,k+2]*T)))))) + probit(alpha)*SQRT(sum {i
in SITES} (Z[i]*(B[i,1]+ sum {k in 2..8 by 3} B[i,k]*(TanH((0.5*(B[i,k+1] +
B[i,k+2]*T)))))**2)) >= 800;

   solve with lso / popsize=100 feastol=1e-6 absfconv=0 nabsfconv=100
maxgen=100;
   solve with nlp relaxint;
   quit;
   """)
```

Figure 7 presents a graph of the efficient frontier produced by the code. You can gain two primary insights from this illustration:

- It would not be possible to enroll 800 patients in less than 306.89 days.
- If the enrollment time for a combination of sites is predicted to be, for example, 400 days, then you could immediately detect the existence of a better solution to activate the sites that would be 93 days faster. This insight would be increasingly valuable if you have higher numbers of potential site activations for the trial of interest.

The choice of objective function and constraints for site selection is restricted to the numerical example of interest in this paper. You can relax the assumptions of this model to meet the objectives and constraints of your patient enrollment settings, however complicated they might be. You can easily incorporate a budget constraint into this formulation. You can also account for deadlines by which the enrollment target is to be met. Furthermore, you can extend underlying stochastic site-selection problem to jointly solve personnel capacity planning problems, especially when you have an upper bound on the number of countries and sites that you can start up at the same time.

The SAS System

The OPTMODEL Procedure

A

	1	2	3	4	5	6	7	8	9	10
1	-143.6158143	1507.0003482	-1.4014377	0.0013644	1070.3558556	2.7059576	-0.0019630	75.6401172	4.5179851	-0.0065854
2	-1220.197967	-2055.963841	-1.0874751	-0.0005170	-50.9594955	1.5696733	-0.0044860	-381.6923627	-1.2913216	0.0015559
3	46.9599216	7789.4570803	-2.1146813	0.0035098	7902.1325629	2.1386995	-0.0035018	445.4979411	-0.9837982	0.0013828
4	1.3395587	-735.9921886	1.9734897	-0.0037890	-163.9827771	-4.0762237	0.0068157	-508.1686476	-1.7731490	0.0027836
5	220.9545408	-333.5060595	-3.0427405	0.0088611	377.6105198	0.2694441	-0.0059263	-637.8868579	2.4383769	-0.0082907
6	124.9993443	-96.8861600	0.0131121	0.0171431	87.0589671	-1.8921722	0.0047748	35.8155021	-1.6188598	0.0139480
7	1316.2334593	2115014.6692	0.0037455	-0.0000089	-1449.226246	-2.4024355	0.0039474	-16657.10633	0.8265954	-0.0014824
8	331.8833082	576.2599919	8.5876351	-0.1519262	329.9050949	2.5400786	-0.0001881	-123.3129056	1.3354352	-0.0067915
9	-130.0887016	-541.9229874	8.5209890	-0.0249520	-467.3236379	-9.3021366	0.0270231	2125.5273926	0.1901239	0.0000050
10	63.6117840	196.9659702	-3.9861450	0.0154786	-23.1462643	8.9368210	-0.0296191	135.2580443	5.6558060	-0.0201332

B

	1	2	3	4	5	6	7	8	9	10
1	6826.7374068	8037.2823260	-2.4828807	-0.0000054	274.1585172	-2.9150497	0.0060597	234.2023548	3.0861812	-0.0061525
2	-3.5342432	104.6277790	0.5343807	-0.0000137	59.1610915	-1.7540132	0.0035597	-27.6786876	-1.4551303	0.0022012
3	74.8031660	41.5165104	-2.1989112	0.0051566	155.4294898	0.3002151	-0.0001212	-97.2510418	1.7267496	-0.0000286
4	40.7908199	26.0688638	-2.9552879	0.0034655	-66.5755257	2.2377680	-0.0059833	76.7363282	0.9807244	-0.0025464
5	178.5762424	-1317.083046	2.1246635	-0.0034625	-636.3752726	-3.0625142	0.0046710	-1304.880357	-0.4426227	0.0011830
6	21.9194204	-42.0808925	2.4518768	-0.0042534	16.0569954	0.7696596	-0.0023140	-12.5544173	-0.6473990	-0.0051086
7	94.9825625	37.2588593	-5.7383648	0.0061793	344.1564963	0.4744288	-0.0025662	-239.6555768	1.3215101	-0.0048732
8	-101.4696786	-44.0719183	-3.3256081	0.0058482	78.6004289	-2.6212974	0.0057714	-226.9149321	-1.2324393	-0.0000466
9	12.7314717	-46.7028731	-0.0002064	-0.0033581	-72.4751057	-1.8596144	0.0059769	75.1837158	-3.1766052	0.0089502
10	-739.7296285	-97.7970690	-1.8628756	0.0022303	23.7038807	-3.4481851	0.0112964	1515.0004194	0.9722893	0.0001922

Figure 6. Coefficients Characterizing Site Enrollment Function Approximations

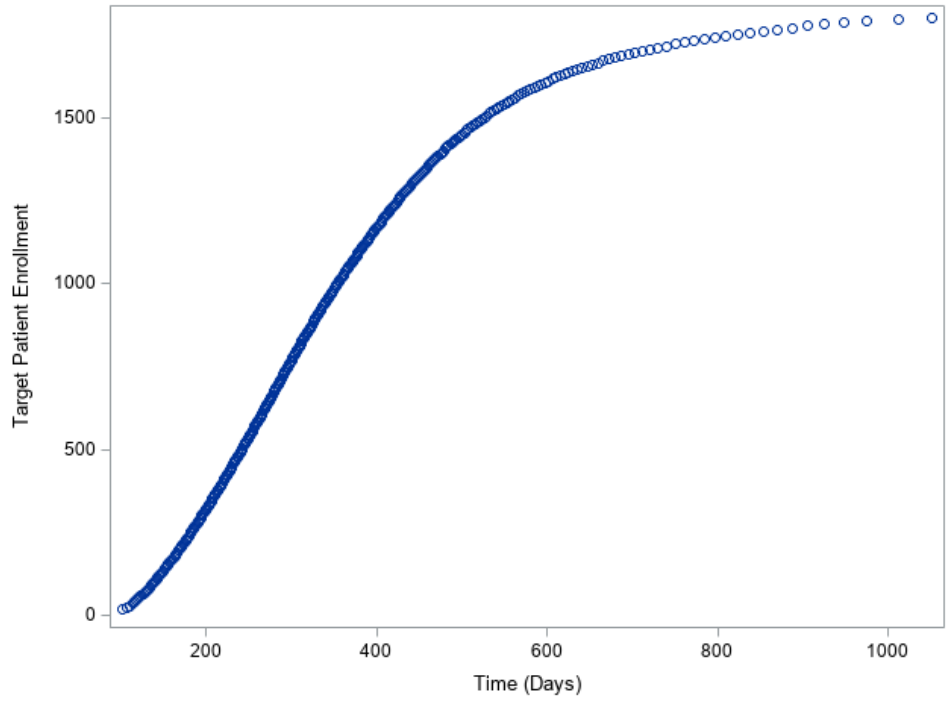


Figure 7. Illustration of Optimal Enrollment Times for Given Targets of Enrollments

CONCLUSION

This paper demonstrates how SAS Clinical Trial Enrollment Simulator, built on SAS Optimization, integrates simulation, machine learning, and optimization to help you design risk-sensitive enrollment plans in real time, accompanied by advanced analytics to determine optimal site activations in order to achieve your target enrollments in the shortest time possible. Throughout the paper, for ease of presentation, the focus is on a numerical example with a single country and 10 sites, each with its own attributes. It is critical to emphasize that SAS Simulation Studio is a tool that has been specifically designed to develop scalable, data-driven, flexible models of dynamic systems that are exposed to high levels of uncertainty. Therefore, the integration of simulation, machine learning, and optimization would readily extend to your need to design clinical trial enrollment plans with large numbers of potential country startups and site activations and with enrollment processes that are significantly more complex than the one illustrated in Figure 1 of the paper. Furthermore, the solution to the problem of strategic clinical trial enrollment planning would readily extend to the study of any complex, dynamic, stochastic system in any domain. Two examples of such applications are data-driven decision support for supply chains in manufacturing and patient flow modeling in health care. The key remains the integration of SAS Simulation Studio with machine learning and artificial intelligence, which learns from large volumes of simulation-generated data, and SAS Optimization, which solves the underlying mathematical programs through its wide range of optimization solvers.

REFERENCES

Cognizant. 2015. "Patient Recruitment Forecast in Clinical Trials." Accessed on February 16, 2019. https://www.cognizant.com/whitepapers/patients-recruitment-forecast-in-clinical-trials-codex1382.pdf.

Elkins, D., LaFleur, C., Foster, E., Tew, J., Biller, B., and Wilson, J.R. 2007. "Clinic: Correlated Inputs in an Automotive Paint Shop Fire Risk Simulation." In Proceedings of the 2007 Winter Simulation Conference. Edited by S.G.Henderson, B.Biller, M.-H. Hsieh, J.Shortle, J.D.Tew, and R.R.Barton, eds. 250–259. Piscataway, NJ: Institute of Electrical and Electronics Engineers.

Handelsman, D. 2012. "Applying Business Analytics to Optimize Clinical Research Operations." In Proceedings of the *SAS Global Forum 2012* Conference. Cary; NC: SAS Institute Inc.

Hughes, E., Pratt, R., and Biller, B. 2018. "Solving Business Problems with SAS Analytics and OPTMODEL." Technology workshop presented at INFORMS Analytics Conference, April 15–17, Baltimore, MD.

RECOMMENDED READING

SAS Simulation Studio 15.1: User's Guide

SAS/OR 15.1 User's Guide: Local Search Optimization

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author:

Bahar Biller
SAS Institute, Analytics Center of Excellence
Bahar.Biller@sas.com

Other brand and product names are trademarks of their respective companies.