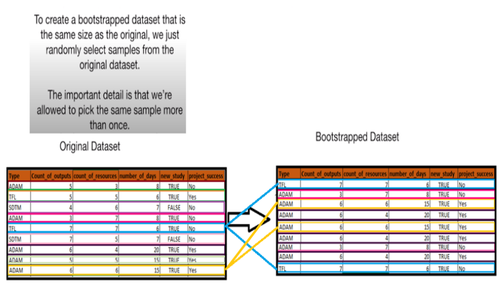
Random Forest Machine Learning Application using R Shiny for Project Timeline Forecasting

Aman Bahl, Associate Director, Syneos Health
Hrideep Antony, Principal Statistical Programmer, Syneos Health

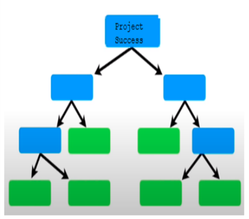
Introduction

- Machine learning models can be used to build powerful tools to facilitate decision-making, especially when the outcome is dependent on numerous and varying factors.
- This poster introduces an R-Shiny based application called "Project Success Forecast' that can predict the possibility of a project's success in a clinical trial environment, using a random forest machine-learning algorithm.
- The rate of success is dependent on the user inputs such as deliverable type, number of resources, number of days, etc. The greater the value of success prediction, the higher is the likelihood of project success.
- Project Success forecast application is built using R-Shiny package.

Step 1: Create a bootstrapped dataset using the training data

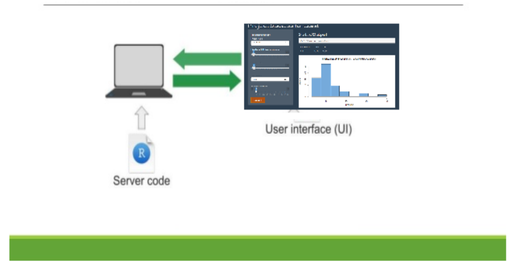


Step 2: Create a decision tree using the bootstrapped dataset from step 1, but only using a random subset of variables at each step.



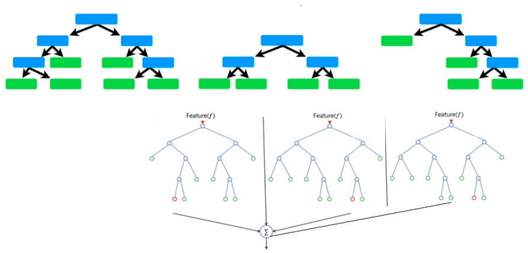
Type	Count_of_outputs	count_of_resources	number_of_days	new_study	project_success
TFL	7	7	6	TRUE	No
ADAM	3	7	8	TRUE	No
ADAM	6	6	15	TRUE	Yes
ADAM	6	4	20	TRUE	Yes
ADAM	6	6	15	TRUE	Yes
ADAM	6	4	20	TRUE	Yes
ADAM	3	7	8	TRUE	No
ADAM	6	4	20	TRUE	Yes
TFL	7	7	6	TRUE	No

R-Shiny application components



- Shiny is an R package that allows application development and comes as a built-in package available through CRAN that interacts well with RStudio. Shiny applications can be deployed as a webpage or a standalone desktop tool to view interactive data or perform statistical analysis.
- UI.R: The function that builds the appearance and assembles the HTML user interface of the application.
- Server Code: The function with instructions on how to execute and rebuild the objects displayed in UI. It also contains the algorithms or models that builds upon data.

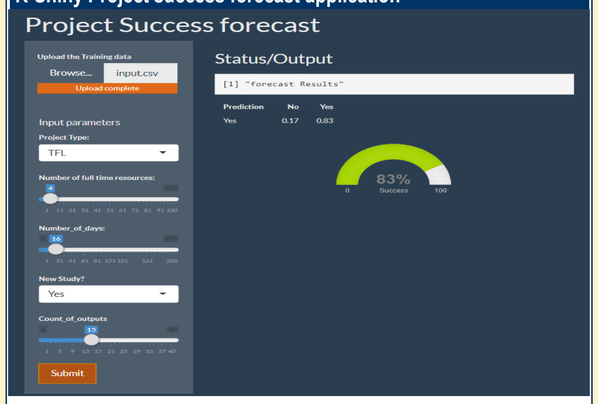
Step3: Repeat step1 and step2 and create more decision trees and use the new input data to run down on all the trees to identify the cumulative predictive outcome



Step4:In this step new sample of data based on which the outcome needs to be predicted is run through the decision trees that has been built using the bootstrapped training data.

Type	Count_of_outputs	count_of_resources	number_of_days	new_study	project_success
ADAM	5	3	8	TRUE	?

R-Shiny Project success forecast application



- Project success forecast application that allows the user to enter real-time information and predict the rate of success of the project. The rate of success is dependent on the user inputs such as deliverable type, number of resources, number of days, etc.
- The greater the value of success prediction, the higher is the likelihood of project success. This application helps the project team to re-assess the project resources or adjust timelines even before the actual initiation of the project, leading to smoother project execution and success.

Random forest package

```
#import the package
library(randomForest)
# Perform training:
rf_classifier = randomForest( data~ ., data=training, ntree=100, mtry=2,
importance=TRUE)
```

Note some important parameters:

–ntree defines the number of trees to be generated.

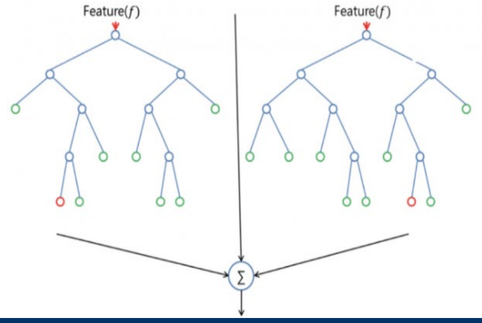
–mtry is the number of features used in the construction of each tree. These features are selected at random, which is where the "random" in "random forests" comes from. The default value for this parameter, when performing classification, is sqrt(number of features).

Training data structure

	A	B	C	D	E	F	G
1	Type	Count_of_outputs	count_of_resources	number_of_days	new_study	project_success	
2	ADAM	5	3	8	TRUE	No	
3	TFL	5	5	6	TRUE	Yes	
4	SDTM	4	6	7	FALSE	No	
5	ADAM	3	7	8	TRUE	No	
6	TFL	7	7	6	TRUE	No	
7	SDTM	7	5	7	FALSE	No	
8	ADAM	6	4	20	TRUE	Yes	
9	ADAM	5	5	15	TRUE	Yes	
10	ADAM	6	6	15	TRUE	Yes	
11	TFL	4	7	15	TRUE	No	
12	SDTM	5	5	20	FALSE	No	
13	SDTM	6	2	4	FALSE	No	
14	ADAM	3	6	20	TRUE	Yes	
15	TFL	9	2	30	TRUE	Yes	
16	TFL	4	5	5	FALSE	No	
17	SDTM	3	5	10	TRUE	Yes	
18	ADAM	5	3	8	TRUE	No	
19	ADAM	3	6	8	TRUE	No	
20	ADAM	10	6	12	TRUE	Yes	

Random Forest Machine Learning Model

- Random forests are based on a simple idea: 'the wisdom of the crowd'.
- Random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction.
- Random forest adds additional randomness to the model, while growing the trees. Instead of searching for the most important feature while splitting a node, it searches for the cumulative best feature among a random subset of features.



References

Reimagining Statistical Reports with R Shiny
https://www.lexjansen.com/pharmasug/2019/AD/PharmaSUG-2019-AD-048.pdf

Contact

Your comments and questions are valued and encouraged. Contact the authors at:

Aman Bahl
+1-905 399 6715
Aman.Bahl@syneoshealth.com

Hrideep Antony
+1-984 459 4785
Hrideep.Antony@syneoshealth.com

POSTER TEMPLATE BY:
www.PosterPresentations.com