Paper JM-04 JMP[®] Pro Bootstrap Forest George J. Hurley, The Hershey Company, Hershey, PA

Abstract

JMP Pro includes a number of analytical features that are very powerful, including a technique called "Bootstrap Forest". The Bootstrap Forest uses many decision tree type classification models, based on data and variable subsets to determine an optimal model. Through this bootstrapping methodology, a superior model can typically be generated relative to typical decision tree partitioning methods. Generally speaking, most applications of classification that would use a typical decision tree can use the bootstrap forest method; hence there is wide applicability of this method across industries. This paper will focus on how to use JMP Pro to perform this analysis, as well as some potential applications of it. The paper is not intended to be a theoretical explanation of this method.

Sample Data

JMP Pro installs with a number of built in datasets. These should be available to anyone who is using JMP. To access these files, you go to the "Help" menu and select "Sample Data". Throughout this paper, various datasets from the sample data will be referenced. All examples in this paper will make use of JMP Pro.

Bootstrap Forest

The bootstrap forest method is available in JMP Pro. Bootstrap Forest is a method that creates many decision trees and in effect averages them to get a final predicted value. Each tree is created from its own random sample, with replacement. The method also limits the splitting criteria to a randomly selected sample of columns.¹

Due to the nature of this methodology, in most instances where a decision tree is applicable, the bootstrap forest method is also an option. This author finds the method particularly useful for data mining and predictive modeling and will leverage these methods for examples.

Titanic Example: Part 1

Here we look at using the bootstrap forest method to investigate drivers of survival for the passengers of the Titanic.

The Titanic data is accessed under "Help", "Sample Data", "Examples for Teaching", "Titanic Passengers". The dataset consists of one record per passenger who traveled on the ill-fated voyage of the Titanic, their demographic information, information on their home and destination, their passenger class, fare, cabin, and other information. A new nominal variable is also created here, called "On A Boat", it represents if the passenger got on any lifeboat at all. If the variable "Lifeboat" is blank, then "On A Boat" is assigned 1; otherwise it is 0.

First, it is typically useful to create a validation dataset prior to running any of the partition methods. To do so, choose "Cols", "New Column". Typically, this column is named "Validate", but does not have to be. Data can be initialized by selection "Random" under "Initialize Data". A random indicator is used for this type of analysis. Hence it should be chosen. In JMP Pro, rows with 0 are used for model training, rows with 1 for validation, and rows with 2 for testing. Choosing the right percentage of each can be somewhat tricky and is dependent on dataset size; this author often begins with an allocation of 0.7 to training and 0.15 to each other classification.

To run a bootstrap forest on this data, select "Analyze", "Model", "Partition" to bring up the partition menu (See Figure 1).

-Select Columns	ng			Columns into Roles	Action	
Passenger C	lass	^	Y, Response	L Survived	ОК	
ASume Name Sex Age Siblings and	Spouses		X, Factor	L Sex Age Siblings and Spouses Parents and Children	Cancel	
Parents and			Weight	optional numeric	Recall	
Ticket #			Freq	optional numeric	Help	
Cabin			Validation	Validate		
Lifeboat Body Home / Dest			By	optional		
Missing value cat Method Validation Portion	egories Bootstrap	Forest 💌 0				

Figure 1: Screenshot of Partition Window Selecting Bootstrap Forest

Here, the variables "Sex", "Age", "Siblings and Spouses", "Parents and Children", "Fare", "Port", and "On A Boat" are chosen as the independent variables, with the response variable being "Survived". For Validation, "Validate" is selected. Validation Portion is left at 0, because a validation column, "Validate" was chosen. Once this is selected, the bootstrap forest window will appear (See Figure 2).

Figure 2: Screenshot of Bootstrap Forest Menu

🛱 Bootstrap Forest		×
Bootstrap Forest Speci	fication	
Number of rows:	1309	
Number of terms:	7	
Number of trees in the	forest:	100
Number of terms sam	pled per split:	1
Bootstrap sample rate	· [1
Minimum Splits Per Tr	ee:	10
Minimum Size Split:		5
Early Stopping Multiple Fits over nu Max Number of] Cancel

The bootstrap forest menu states the number of rows and terms (dependent/factor variables) chosen. Several options are then given. Per the JMP Pro 10.0.0 help file¹, they are:

Number of trees in the forest is the number of trees to grow, and then average together.

Number of terms sampled per split is the number of columns to consider as splitting candidates at each split. For each split, a new random sample of columns is taken as the candidate set.

Bootstrap sample rate is the proportion of observations to sample (with replacement) for growing each tree. A new random sample is generated for each tree.

Minimum Splits Per Tree is the minimum number of splits for each tree.

Minimum Size Split is the minimum number of observations needed on a candidate split.

Early Stopping is checked to perform early stopping. If checked, the process stops growing additional trees if adding more trees doesn't improve the validation statistic. If not checked, the process continues until the specified number of trees is reached. This option appears only if validation is used.

Multiple Fits over number of terms is checked to create a bootstrap forest for several values of Number of terms sampled per split. The lower value is specified above by the Number of terms samples per split option. The upper value is specified by the following option:

Max Number of terms is the maximum number of terms to consider for a split.

The Titanic data, as seen above, defaults to 100 trees in the forest, 1 term sampled per split, with a sample size equal to the entire data set (note that it is not likely that the dataset sampled will match the underlying dataset because the sampling is done with replacement), a minimum of 10 splits per tree, and a minimum of 5 observations needed for the candidate split. Early Stopping and Multiple Fits are not selected. To run with the default settings, click "OK". Figure 3 shows results as presented by the bootstrap forest.

Bootstrap Forest	for Surviv	ved					
Specifications							
Target Column:		Survived	Trainin	g rows:		91	16
Validation Column:		Validate	Validati	on rows:		19	97
			Test Ro	WS		19	36
Number of trees in the	forest::	100	Numbe	r of terms	в:		7
Number of terms sam	oled per split	: 1	Bootstr	ap samp	les:	91	16
			Minimu	m Splits	Per Tre	e: 1	10
			Minimu	m Size S	plit:		5
Overall Statistics Measure Entropy RSquare Generalized RSquare		0.4205 0.5816	0.4302 1 0.5958 (1	-(L(0)/L(model))		
Measure Entropy RSquare	0.4376 0.5991 0.3722 0.3222 0.2998 0.0622 916	0.4205 0.5816 0.3836 0.3297 0.3078	0.4302 1	Loglike() -(L(0)/L() -Log(ρ()) Σ(V())-ρ() V())-ρ()]/	model)))/n)³/n n		
Measure Entropy RSquare Generalized RSquare Mean -Log p RMSE Mean Abs Dev Misclassification Rate N	0.4376 0.5991 0.3722 0.3222 0.2998 0.0622 916	0.4205 0.5816 0.3836 0.3297 0.3078 0.0711 197	0.4302 1 0.5958 (1 0.3873 Σ 0.3324 √ 0.3097 Σ 0.0918 Σ	Loglike() -(L(0)/L() -Log(ρ()) Σ(V())-ρ() V())-ρ()]/	model)))/n)₹/n n ax)/n		

Figure 3: Screenshot of Bootstrap Forest Results

The results give a variety of statistics describing the model fit, such as Generalized R-Square, RMSE, and the Misclassification Rate.

What is immediately apparent from these results is that, based on the confusion matrix, the model does an extremely well at predicting survivors (as both the validation and test sets had every predicted survivor live). Furthermore, it does relatively well at predicting who died in the shipwreck.

If we go to the red arrow and click Column Contributions, a Column Contribution Summary will be presented. From Figure 4, it is clear that "On A Boat" had a large contribution. This, of course makes sense, and it can be hypothesized that those securing a seat on a lifeboat would have a higher propensity to survive than those left in the frigid waters.

Actual I	Predicted	Actu	al	Pre	dicted	Actual	Pre	edicted
Training N	o Yes	Va	lidation	No	Yes	Test	No	Yes
No 56	7 5	No		123	0	No	114	0
Yes 5	2 292	Yes	3	14	60	Yes	18	64
Cumulative Va	lidation	1						
Per-Tree Summ	naries							
Column Contri	butions							
	Num	ber						
Term	of S	olits	(G^2				
Sex		56	5839.	131		1	ų.	
Age		68	373.	183	4		1	
Siblings and Spous	ses	59	352.	794	1	1 1	4	
Parents and Childr	en	45	498.	339	1	4 4	- 3	
Fare		63	1457.	955		4	- ÷	
Port		55	522.	232	1	4 4	1	
On A Boat		50	17304.	903				

Figure 4: Screenshot of Column Contributions

Based on the confusion matrix, we have built a useful predictive model using the bootstrap forest methodology. Prediction formulas and values can be saved by clicking the red triangle and choosing "Save Columns" and then "Save Predicteds" or "Save Prediction Formula". This is especially useful if there are rows for which the condition is unknown and needs to be predicted based on information available. While in this case, we know with certainty the fates of each passenger, the analogous use would be to consider, for example, another boat, similar to the Titanic, launching. An individual may want to understand predict their propensity to survive if it should sink, based on their demographics, fare type, and life boat access. Further, before purchasing the ticket, they may want to consider what factors contribute to survival. Clearly, this is a contrived example (using non-confidential data), but it should be apparent that this methodology would be particular useful in business.

Titanic Example: Part 2

Since the lifeboat was such a large contributor to survival, yet many people did not have lifeboats, it is reasonable to consider if a reasonable model can be made to predict survival of those who waited for

rescue in the water. However, a quick look at graph builder proves this will be a daunting task due to the small number of non-lifeboat survivors.

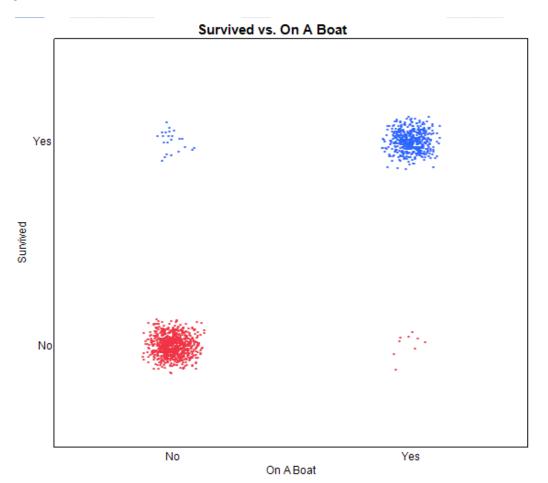


Figure 5: Survivors and Lifeboats

To look into this, rows for people not on a life boat can be shown and included using the data filter. The process for creating the bootstrap forest model is repeated, but without "On A Boat" as a predictive variable.

The results, however, predicted no survivors and hence, did not predict survival. A scaled up model with more trees and terms was created with the selections seen in Figure 6. Figure 7 shows that it also failed to classify the survivals.

This illustrates that this method, like most methods, has a difficult time predicting rare events.

Figure 6: Scaled Up Model Selection

Bootstrap Forest	X
Bootstrap Forest Specification	
Number of rows: 1309	
Number of terms: 7	
Number of trees in the forest:	1000
Number of terms sampled per spli	t 1
Bootstrap sample rate:	1
Minimum Splits Per Tree:	10
Minimum Size Split:	1
Early Stopping	
Multiple Fits over number of term	ns:
Max Number of terms:	5
ОК	Cancel

Figure 7: Scaled Up Model Results

Doording	p Fore	st for S	urvive	d						
Model Val	lidatio	n-Set S	umma	ries						
The fit below	was the	best of t	hese m	odels f	it.					
			y Misc	lassific	cation				A	vg Abs
N Terms N		- 4 R T. 175			Rate	19102		RMS Erro		Error
1	1000	0.120		1.1	.0333		1286	0.176		0.0597
2	1000	0.266			.0333	22	1072	0.170	1.11	0.0563
3	1000	0.327		55	.0333		0983	0.169		0.0522
4	1000	0.314	-		0333		1002	0.172		0.0493
5	1000	0.236	9	0	.0333	0	1115	0.176	51	0.0486
Specificat	lions									
Target Colun	nn:		S	Survive	t	Traini	ng row	S:		578
Validation Co	olumn:		C	olumn	17	Valida	tion ro	ws:		120
						Test F	lows			125
Number of tr	ees in th	ne forest:		10	00	Numb	er of te	rms:		7
Number of te	erms sa	mpled pe	r split:		3	Boots	trap sa	mples:		578
						Minim	um Sp	lits Per Tr	ee:	10
						Minim	um Siz	e Split:		1
Overall St	tatistic	s								
Measure		Trai	n <mark>ing V</mark> a	alidatio	on T	Fest D	efinitio	n		
Measure Entropy RSc	quare		ning Va 1748					n e(model)/l	Loglil	ke(0)
Entropy RS0 Generalized	RSqua	0.4 re 0.5	748 079	0.327	71 0.2 98 0.2	232 1- 376 (1	Loglik -(L(0)/I	e(model)/l _(model))	100 200	
Entropy RSG Generalized Mean -Log p	RSqua	0.4 re 0.5 0.0	748 079 0697	0.327 0.359 0.098	71 0.2 98 0.2 33 0.0	232 1- 376 (1 637 Σ	Loglik -(L(0)/I -Log(p	 e(model)/l _(model)) (j])/n	100 200	
Entropy RSG Generalized Mean -Log (RMSE	d RSqua p	0.4 re 0.5 0.0 0.1	079 0697 417	0.327 0.359 0.098 0.169	71 0.2 98 0.2 33 0.0 90 0.1	232 1- 376 (1 637 Σ 219 √	Loglik -(L(0)/I -Log(p ∑(/[j]-p	e(model)/l _(model)) [j])/n [j])²/n	100 200	
Entropy RSG Generalized Mean -Log (RMSE Mean Abs D	d RSqua p)ev	0.4 re 0.5 0.0 0.1 0.0	1748 1079 1697 1417 1435	0.327 0.359 0.098 0.169 0.052	71 0.2 98 0.2 33 0.0 90 0.1 22 0.0	232 1- 376 (1 637 Σ 219 √ 351 Σ	Loglik -(L(0)/I -Log(ρ Σ(y[i]-ρ Ιy[i]-ρ[i	 e(model)/l _(model)) (j])/n (j])²/n]]/n	100 200	
Entropy RSG Generalized Mean -Log I RMSE Mean Abs D Misclassific	d RSqua p)ev	0.4 re 0.5 0.0 0.1 0.0 ate 0.0	1748 1079 1697 1417 1435 1294	0.327 0.359 0.098 0.169 0.052 0.052	71 0.2 98 0.2 33 0.0 90 0.1 22 0.0	232 1- 376 (1 637 Σ 219 √ 351 Σ 160 Σ	Loglik -(L(0)/I -Log(ρ Σ(y[i]-ρ Ιy[i]-ρ[i	 e(model)/l _(model)) (j])/n (j])²/n]]/n	100 200	
Entropy RSG Generalized Mean -Log (RMSE Mean Abs D	d RSqua p)ev	0.4 re 0.5 0.0 0.1 0.0 ate 0.0	1748 1079 1697 1417 1435	0.327 0.359 0.098 0.169 0.052	71 0.2 98 0.2 33 0.0 90 0.1 22 0.0	232 1- 376 (1 637 Σ 219 √ 351 Σ	Loglik -(L(0)/I -Log(ρ Σ(y[i]-ρ Ιy[i]-ρ[i	 e(model)/l _(model)) (j])/n (j])²/n]]/n	100 200	
Entropy RSG Generalized Mean -Log I RMSE Mean Abs D Misclassific	d RSqua p)ev :ation R:	0.4 nre 0.5 0.0 0.1 0.0 ate 0.0 5	1748 1079 1697 1417 1435 1294	0.327 0.359 0.098 0.169 0.052 0.052	71 0.2 98 0.2 33 0.0 90 0.1 22 0.0	232 1- 376 (1 637 Σ 219 √ 351 Σ 160 Σ	Loglik -(L(0)/I -Log(ρ Σ(y[i]-ρ Ιy[i]-ρ[i	 e(model)/l _(model)) (j])/n (j])²/n]]/n	100 200	
Entropy RSG Generalized Mean -Log p RMSE Mean Abs D Misclassific N	d RSqua p Dev cation Ra on Ma	0.4 nre 0.5 0.0 0.1 0.0 ate 0.0 5	1748 1079 1697 1417 1435 1294	0.327 0.359 0.098 0.169 0.052 0.052	71 0.2 98 0.2 93 0.0 90 0.1 22 0.0 33 0.0	232 1- 376 (1 637 Σ 219 √ 351 Σ 160 Σ	Loglikı -(L(0)/I -Log(ρ Σ(/[]-ρ /[]-ρ[] (ρ[]]≠ρ	e(model)/l _(model)) [j]]/n [j])²/n]]/n Max)/n	^(2/n)	
Entropy RSG Generalized Mean -Log p RMSE Mean Abs D Misclassific N	d RSqua p Dev cation R: on Ma	0.4 re 0.5 0.0 0.1 0.0 ate 0.0 5 trix	1748 1079 1697 1417 1435 1294 78	0.327 0.359 0.098 0.169 0.052 0.033 120	71 0.2 98 0.2 93 0.0 90 0.1 22 0.0 33 0.0	232 1- 376 (1 637 Σ 219 √ 351 Σ 160 Σ 125 π	Loglikı -(L(0)/I -Log(ρ Σ(/[]-ρ /[]-ρ[] (ρ[]]≠ρ	e(model)/l _(model)) [j])/n [j]) ² /n]J/n Max)/n	^(2/n)	
Entropy RSd Generalized Mean -Log p RMSE Mean Abs D Misclassific N	d RSqua p Dev cation R: on Ma	0.4 o.2 0.1 0.2 0.1 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2	748 6079 6697 417 1435 1294 78	0.327 0.359 0.098 0.169 0.052 0.033 120	71 0.2 98 0.2 93 0.0 90 0.1 22 0.0 33 0.0 Pre	232 1- 376 (1 637 Σ 219 √ 351 Σ 160 Σ 125 π	Logliki -(L(0)/I -Log(ρ Σ(/U)-ρ [/U)-ρ[i (ρ[j]≠ρ Actua	e(model)/l _(model)) [j])/n [j]) ² /n]J/n Max)/n	^(2/n)	
Entropy RS Generalized Mean -Log p RMSE Mean Abs D Misclassific N Confusi Actual Training	d RSqua p Dev cation Ra on Ma P g No	0.4 0.5 0.1 0.0 0.1 0.0 0.1 0.0 0.1 0.0 0.1 0.0 0.1 0.0 0.1 0.0 0.1 0.0 0.1 0.0 0.0	748 6079 6697 417 4435 9294 78 Actual Valida	0.327 0.359 0.098 0.169 0.052 0.033 120	71 0.2 98 0.2 93 0.0 90 0.1 22 0.0 33 0.0 97 90 0.1 90 0.2 90 0.1 90 0.1	232 1- 376 (1 637 Σ 219 √ 351 Σ 160 Σ 125 n dicted Yes	Loglika -(L(0)/I -Log(ρ Σ(/(i)-ρ[/[i]-ρ[i (ρ[j]≠ρ Actua	e(model)/l _(model)) (j])/n (j])?/n J//n Max)/n I Pred	^(2/n) licted Yes	
Entropy RS Generalized Mean -Log p RMSE Mean Abs D Misclassific N Confusi Actual Training No Yes	d RSqua p Dev cation Ra on Ma P g No 561 17	0.4 re 0.5 0.0 0.1 0.0 trix redicted Yes 0 0	1748 1079 1697 1417 1435 1294 78 Actual Valida No	0.327 0.359 0.098 0.169 0.052 0.033 120	71 0.2 98 0.2 93 0.0 90 0.1 22 0.0 33 0.0 93 0.0 Pre No 116	232 1- 376 (1 637 Σ 219 √ 351 Σ 160 Σ 125 n dicted Yes 0	Loglika -(L(0)// -Log(ρ Σ(/ὑ]-ρἱ (ρ[j]≠ρ Actua Test No	(model)/l (model)/ (())/n (()) ² /n ()//n Max)/n	^(2/n) licted Yes 0	
Entropy RS: Generalized Mean -Log p RMSE Mean Abs D Misclassific N Actual Training No	d RSqua p Dev cation Ra p g No 561 17 ve Vali	0.4 re 0.5 0.0 0.1 0.1 0.0 5 trix redicted 0 Yes 0 0 0 0	1748 1079 1697 1417 1435 1294 78 Actual Valida No	0.327 0.359 0.098 0.169 0.052 0.033 120	71 0.2 98 0.2 93 0.0 90 0.1 22 0.0 33 0.0 93 0.0 Pre No 116	232 1- 376 (1 637 Σ 219 √ 351 Σ 160 Σ 125 n dicted Yes 0	Loglika -(L(0)// -Log(ρ Σ(/ὑ]-ρἱ (ρ[j]≠ρ Actua Test No	(model)/l (model)/ (())/n (()) ² /n ()//n Max)/n	^(2/n) licted Yes 0	

Titanic Example: Part 3

Finally, since lifeboats saved so many lives, it is worthwhile to ask to whom a lifeboat was provided. The bootstrap forest methodology can be used to consider this as well.

In this instance, the response becomes "On A Boat". Using the default JMP settings, it is seen in Figure 8 that sex was a big driver of who got a life boat. Age interestingly didn't have as high of a contribution as anticipated. However, Fare was also a large driver of who got a lifeboat.

Further, it can be seen that this model had a misclassification rate of 0.2451 and performed lower than hoped for in the validation and test conditions under the Confusion Matrix Section.

12 11	Passengers										10	
Boo	otstrap Fo	orest	for C	n A	Boat							
⊿ Spe	cification	s										
Targe	t Column:				On A Boa	t	Trai	ning ra	ws:			914
Valida	ation Colum	n:			Column 1	17	Vali	dation	rows	:		206
							Tes	tRows	5			189
Numb	per of trees i	in the i	forest:	:	10	0	Nur	nber o	fterm	IS:		6
Numb	per of terms	samp	led pe	er spli	t I	1	Boo	tstrap	sam	ples:		914
							Mini	mum	Splits	PerT	ree:	10
							Mini	mum	Size S	Split:		5
⊿ Ove	rall Statis	tics										
Mea	sure		Trai	ning	Validation	-	Test	Defini	tion			
	opy RSquare	e		508						nodel	/Loglike	(0)
Gen	eralized RS	quare	0.2	459	0.2498	0.2	689	(1-(L(0))/L(n	nodel)))^(2/n))/	(1-L(0)^(2/n)
Mea	n -Log p		0.5	585	0.5768	0.5	322	Σ-Log	(p[j])	/n		
RMS	E		0.4	321	0.4412	0.4	175	√Σ(V[j]-p[j]);	²/n		
Mear	n Abs Dev		0.4	067	0.4169	0.3	8888	Σ ly[i]-	p[j] /n			
Misc	lassification	Rate	0.2	451	0.2476	0.1	958	Σ (p[j]	≠pMa	x)/n		
N			9	14	206		189					
⊿ Co	onfusion M	Matri	x									1
A	ctual	Pred	licted	Actu	al	Pre	dicte	d Act	Jal	Pre	dicted	
	Training	0	1	Val	idation	0	1	Te	st	0	1	
() 8	556	22	0		119	1	0		122	3	
1	1 3	202	134	1		50	36	1		34	30	
Cun	ulative V	alida	tion									
	Tree Sum			3								
	umn Cont	ribut	ions									
			Num	ber								
			of Sp	olits	G^	2						
Term				62	4094.193	4						
Term Sex				68	603.480	7						
				71	500.041	9		1				
Sex Age Siblin	gs and Spo					100	1 1	1.1				
Sex Age Siblin	igs and Spo its and Chile			60	548.259	6	1 1	1	- 11 -			
Sex Age Siblin	- 010 - 010 - 010 - 010 - 01			60 79 71	548.259 2144.537 550.198	0		1				

Figure 8: Lifeboat Prediction: Default Settings

Wanting to improve predictive power, and considering that it is counterintuitive that age was not contributing more, the process was repeated using the same specifications shown in Figure 6. The result can be seen in Figure 9.

Figure 9: Lifeboat Prediction: Scaled Up

M D	odel Va	lidatio	n-Set S	Summ	aries						
-			Entrop	oy Mis	classific	ation				1	Avg Abs
N	Terms I	V Trees	R Squar			Rate	Avg -	Log p	RMS Err		Error
	3	1000	0.308			.2039		.4700	0.38		0.3095
	4	1000	0.302	23	0	.1990	0	.4741	0.38	87	0.3011
	5	750	0.289	96	0	.2136	0	4827	0.38	97	0.2963
d Sp	pecifica	tions									
Tar	rget Colu	mn:			On A Bo	at	Traini	ng row	s:		914
Val	idation C	olumn:			Column	17	Valida	ation ro	ws:		206
							Test F	Rows			189
Nu	mber of t	rees in th	ne forest	t:	10	00	Numb	per of te	erms:		6
Nu	mber of t	erms sa	mpled p	er split:		3	Boots	trap sa	amples:		914
							Minim	ium Sp	lits Per T	ree:	10
							Minim	ium Siz	ze Split:		1
0	verall S	tatistic	s								
RI	ean -Log MSE		0. 0.	6461 3377 3153	0.470 0.387	00 0.4 79 0.3	1346 Σ 3668 √	-Log(ρ Σ(y[j]-ρ	o[j])/n o[j])²/n	/*(2/II))/(1-L(0)
RI Mi N	ean -Log	p Dev cation Ra	0. 0. ate 0. 9	3377	0.470	00 0.4 79 0.3 95 0.2	1346 Σ	-Log(ρ Σ(y[i]-ρ y[i]-ρ[i	o[j])/n o[j])²/n] /n	r (zm))/(1-L(0)
RI Mi N	ean -Log MSE ean Abs I isclassifi	p Dev cation Ra ion Ma	0. 0. ate 0. 9	3377 3153 2499 1302	0.470 0.387 0.309 0.203 206	00 0.4 79 0.3 95 0.2 89 0.1	1346 Σ 3668 √ 2816 Σ 1746 Σ	-Log(ρ Σ(y[i]-ρ y[i]-ρ[i	o[j])/n o[j])*/n]]/n Max)/n	dicted	
RI Mi N	ean -Log MSE ean Abs I isclassifi Confus	p Dev cation Ra ion Ma Pi	0. 0. ate 0. trix redicted	3377 3153 2499 1302 914 Actua	0.470 0.387 0.309 0.203 206	00 0.4 79 0.3 95 0.2 89 0.1	1346 Σ 3668 √ 2816 Σ 1746 Σ 189 n	-Log(ρ Σ(ν[i]-ρ ν[i]-ρ[i (ρ[i]≠ρ	o[j])/n o[j])*/n]]/n Max)/n		
RI Mi N	ean -Log MSE ean Abs I isclassifi Confus Actual Trainin 0	p Dev cation Ra ion Ma Pi g 0 558	0. 0. 0. 1. 1. 1. 20	3377 3153 2499 1302 914 Actua Valio 0	0.470 0.387 0.309 0.203 206	00 0.4 79 0.3 95 0.2 39 0.1 97 0.2 39 0.1 Pre 0 109	4346 Σ 3668 √ 2816 Σ 1746 Σ 189 n dicted 1 11	-Log(ρ Σ(y[i]-ρ[i (p[i]≠ρ Actua Test	b[j])/n b[j])*/n]]/n Max)/n I Pred I Pred 112	dicted 1 13	
RI Mi N	ean -Log MSE ean Abs I isclassifi Confus Actual Trainin	p Dev cation Ra ion Ma Pi g 0	0. 0. 0. 1. 1. 1. 20	3377 3153 2499 1302 914 Actua Vali	0.470 0.387 0.309 0.203 206	00 0.4 79 0.3 95 0.2 39 0.1 Pre 0	4346 Σ 3668 √ 2816 Σ 1746 Σ 189 n dicted 1	-Log(ρ Σ(Vi)-ρi (Vi)-ρi (ρ[j]≠ρ Actua	o[j])/n o[j])*/n]]/n Max)/n I Prec t 0	dicted 1	
	ean -Log MSE ean Abs I isclassifi Confus Actual Trainin 0	p Dev cation Ra ion Ma Pi ig 0 558 99	0. 0. ate 0. 5 trix redicted 1 20 237	3377 3153 2499 1302 914 Actua Valio 0 1	0.470 0.387 0.309 0.203 206	00 0.4 79 0.3 95 0.2 39 0.1 97 0.2 39 0.1 Pre 0 109	4346 Σ 3668 √ 2816 Σ 1746 Σ 189 n dicted 1 11	-Log(ρ Σ(y[i]-ρ[i (p[i]≠ρ Actua Test	b[j])/n b[j])*/n]]/n Max)/n I Pred I Pred 112	dicted 1 13	
	ean -Log MSE ean Abs I isclassifi Confus Actual Trainin 0 1	p Dev cation Ra ion Ma Pr 0 0 558 99 ve Vali	0. 0. ate 0. trix redicted 1 20 237 dation	3377 3153 2499 1302 914 Actua Valio 0 1	0.470 0.387 0.309 0.203 206	00 0.4 79 0.3 95 0.2 39 0.1 97 0.2 39 0.1 Pre 0 109	4346 Σ 3668 √ 2816 Σ 1746 Σ 189 n dicted 1 11	-Log(ρ Σ(y[i]-ρ[i (p[i]≠ρ Actua Test	b[j])/n b[j])*/n]]/n Max)/n I Pred I Pred 112	dicted 1 13	
RI Mi Mi N	ean -Log MSE ean Abs I isclassifi Confus Actual Trainin 0 1 umulati er-Tree	p Dev cation Ra ion Ma P g 0 558 99 ve Vali Summ	0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0	3377 3153 2499 1302 914 Actua Vali 0 1	0.470 0.387 0.309 0.203 206	00 0.4 79 0.3 95 0.2 39 0.1 97 0.2 39 0.1 Pre 0 109	4346 Σ 3668 √ 2816 Σ 1746 Σ 189 n dicted 1 11	-Log(ρ Σ(y[i]-ρ[i (p[i]≠ρ Actua Test	b[j])/n b[j])*/n]]/n Max)/n I Pred I Pred 112	dicted 1 13	
RI Mi Mi N	ean -Log MSE ean Abs I isclassifi Confus Actual Trainin 0 1 umulati	p Dev cation Ra ion Ma P g 0 558 99 ve Vali Summ	0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0	3377 3153 2499 1302 914 Actua Vali 0 1	0.470 0.387 0.309 0.203 206	00 0.4 79 0.3 95 0.2 39 0.1 97 0.2 39 0.1 Pre 0 109	4346 Σ 3668 √ 2816 Σ 1746 Σ 189 n dicted 1 11	-Log(ρ Σ(y[i]-ρ[i (p[i]≠ρ Actua Test	b[j])/n b[j])*/n]]/n Max)/n I Pred I Pred 112	dicted 1 13	
RI Mi Mi N	ean -Log MSE ean Abs I isclassifi Actual Trainin 0 1 umulati er-Tree olumn (p Dev cation Ra ion Ma P g 0 558 99 ve Vali Summ	0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0	3377 3153 2499 1302 914 Actua Vali 0 1	0.470 0.387 0.309 0.203 206	00 0.4 79 0.3 95 0.2 39 0.1 97 0.2 39 0.1 Pre 0 109	4346 Σ 3668 √ 2816 Σ 1746 Σ 189 n dicted 1 11	-Log(ρ Σ(y[i]-ρ[i (p[i]≠ρ Actua Test	b[j])/n b[j])*/n]]/n Max)/n I Pred I Pred 112	dicted 1 13	
	ean -Log MSE ean Abs I isclassifi Actual Trainin 0 1 umulati er-Tree olumn (p Dev cation Ra ion Ma P g 0 558 99 ve Vali Summ	0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0	3377 3153 2499 1302 914 Actua Valia 0 1	0.470 0.387 0.309 0.203 206	00 0.4 79 0.3 15 0.2 99 0.1 109 31	4346 Σ 3668 √ 2816 Σ 1746 Σ 189 n dicted 1 11	-Log(ρ Σ(y[i]-ρ[i (p[i]≠ρ Actua Test	b[j])/n b[j])*/n]]/n Max)/n I Pred I Pred 112	dicted 1 13	
RI Mi Mi N A (C C C C C C C C C C C C C C C C C C	ean -Log MSE ean Abs I isclassifi Actual Trainin 0 1 umulati er-Tree olumn (rm x e	p Dev cation Ra ion Ma Pr g 0 558 99 ve Vali Summ Contrib	0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0	3377 3153 2499 1302 914 Actua Valia 0 1 1 s nber plits 2817 7318	0.470 0.387 0.309 0.203 206 I dation G 115444. 104751.	00 0.4 79 0.3 95 0.2 39 0.1 109 31 *2 30 	4346 Σ 3668 √ 2816 Σ 1746 Σ 189 n dicted 1 11	-Log(ρ Σ(y[i]-ρ[i (p[i]≠ρ Actua Test	b[j])/n b[j])*/n]]/n Max)/n I Pred I Pred 112	dicted 1 13	
RI Mi N A (Cu Pe Cu Pe Cu Se: Age Sib	ean -Log MSE ean Abs I isclassifi Actual Trainin 0 1 umulati er-Tree olumn (rm x e	p Dev cation Ra ion Ma P g 0 558 99 ve Vali Summ Contrib	0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0	3377 3153 2499 1302 914 Actua Valia 0 1 1 s nber plits 2817 7318 5294	0.470 0.387 0.309 0.203 206 dation dation G 115444. 104751. 45489.	00 0.4 79 0.3 95 0.2 39 0.1 97	4346 Σ 3668 √ 2816 Σ 1746 Σ 189 n dicted 1 11	-Log(ρ Σ(y[i]-ρ[i (p[i]≠ρ Actua Test	b[j])/n b[j])*/n]]/n Max)/n I Pred I Pred 112	dicted 1 13	
RI Mi N A (C C C C C C C C C C C C C C C C C C	ean -Log MSE ean Abs I isclassifi Actual Trainin 0 1 umulati er-Tree olumn (rm x e olings and rents and	p Dev cation Ra ion Ma P g 0 558 99 ve Vali Summ Contrib	0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0	3377 3153 2499 1302 914 Actua Valia 0 1 1 s nber plits 2817 7318 5294 3252	0.470 0.387 0.309 0.203 206 I dation G 115444. 104751. 45489. 37821.	00 0.4 79 0.3 95 0.2 39 0.1 Pre 0 109 31 *2 30 00 87 00	4346 Σ 3668 √ 2816 Σ 1746 Σ 189 n dicted 1 11	-Log(ρ Σ(y[i]-ρ[i (p[i]≠ρ Actua Test	b[j])/n b[j])*/n]]/n Max)/n I Pred I Pred 112	dicted 1 13	
RI Mi N A (Cu Pe Cu Pe Cu Se: Age Sib	ean -Log MSE ean Abs I isclassifi Actual Trainin 0 1 umulati er-Tree olumn C rm x e olings and rents and re	p Dev cation Ra ion Ma P g 0 558 99 ve Vali Summ Contrib	0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0	3377 3153 2499 1302 914 Actua Valia 0 1 1 s nber plits 2817 7318 5294	0.470 0.387 0.309 0.203 206 dation dation G 115444. 104751. 45489.	00 0.4 79 0.3 95 0.2 39 0.1 Pre 0 109 31 *2 30 00 87 00 77	4346 Σ 3668 √ 2816 Σ 1746 Σ 189 n dicted 1 11	-Log(ρ Σ(y[i]-ρ[i (p[i]≠ρ Actua Test	b[j])/n b[j])*/n]]/n Max)/n I Pred I Pred 112	dicted 1 13	

From Figure 9, it is noted that the misclassification rate is now at 0.1302, which is a large improvement over the results using the default method. In the model chosen here, it is apparent that age is also a contributor, validating the expression, "Women and children first!" ... but a thorough look at the results may suggest a revision to "Wealthy people, women, and children first!"

Conclusion

Through the use of bootstrap forest methodology, it has been shown that a dataset can be mined and predictive models built. This methodology is a very useful tool for performing these types of analyses, but can be computationally expensive, especially when performing analyses like the "Scaled Up" versions described in this paper, which were run on a very powerful machine. However, as computing power has become more available, methods like bootstrap forest are becoming more and more accessible to all analytics professionals.

References

¹ JMP Pro 10.0.0 Help

Contact Info

Please contact the author with any questions.

George J. Hurley The Hershey Company 19 E Chocolate Ave. Hershey, PA 17033 ghurley@hersheys.com

Trademark Citation

JMP, SAS, and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. [®] indicates USA registration. Other brand and product names are trademarks of their respective companies.