

```

/* Generated Code (IMPORT) */
/* Source File: nest456.csv */
/* Source Path: /home/greenberg29390/my_courses/richardcutler0/STAT 5650 Stat Learning & Data Mining I */
/* Code generated on: 2/7/20, 12:00 PM */

```

```
%web_drop_table(WORK.Nest);
```

```
FILENAME REFFILE '/home/greenberg29390/sasuser.v94/Nest.csv';
```

```
PROC IMPORT DATAFILE=REFFILE
```

```
    DBMS=CSV
```

```
    OUT=WORK.Nest;
```

```
    GETNAMES=YES;
```

```
RUN;
```

```
ods rtf file='/home/greenberg29390/sasuser.v94/hw4.rtf';
```

```
ods graphics on;
```

```
%web_drop_table(WORK.Glass);
```

```
FILENAME REFFILE '/home/greenberg29390/sasuser.v94/Glass.csv';
```

```
PROC IMPORT DATAFILE=REFFILE
```

```
    DBMS=CSV
```

```
    OUT=WORK.Glass;
```

```
    GETNAMES=YES;
```

```
RUN;
```

```
%web_open_table(WORK.Glass);
```

```

/*
title1 'FITTING A CLASSIFICATION TREE TO GLASS';
title2 'Selection of Tree Size. Random Seed set to 123';
proc hpsplit data=Glass cvmethod=random(10) seed=123
    intervalbins=5000 cvmodelfit plots(only)=cvcc;
class GlassType;
model GlassType =
Aluminum Barium Calcium Iron Magnesium Potassium Refindex Silicon Sodium;
grow gini;
run;
title1 'FITTING A CLASSIFICATION TREE TO GLASS';
title2 'Tree with 12 Terminal Nodes. Random Seed set to 123';
proc hpsplit data=Glass cvmethod=random(10) seed=123
    intervalbins=5000 cvmodelfit plots(only)=cvcc;
class GlassType;
model GlassType =
Aluminum Barium Calcium Iron Magnesium Potassium Refindex Silicon Sodium;
grow gini;
prune costcomplexity (leaves=12);
code file='/home/greenberg29390/sasuser.v94/fournodes3.sas';
rules file='/home/greenberg29390/sasuser.v94FournodesRules2.txt';
run;
*/

```

```
Options pageno=1 nodate; run;
```

```
title1 "FITTING RANDOM FORESTS TO Glass DATA" ;
```

```
title2 "Using Glass as Training Data and pilotI as Validation Data" ;
```

```
title3 ;
```

```
proc hpforest data=Glass maxtrees=200 inbagfraction=0.6 scoreprole=oob
```

```
seed=5461;
```

```
input Aluminum Barium Calcium Iron Magnesium Potassium Refindex Silicon Sodium / level=interval;
```

```
    save file='/home/greenberg29390/rfglass.txt';
```

```
target GlassType / level=nominal;
```

```
score
```

```
out=SNRAPred;
```

```
ods output
```

```
FitStatistics=fitstats(rename=(Ntrees=Trees));
```

```
run
```

```
;
```

```

data
  fitstats;
  set fitstats;
  label
  Trees = 'Number of Trees'
;
  label
  MiscAll = 'Using ALL Observations';
  label
  MiscOob = 'Using Only OOB Observations'
;
run
;
proc
sgplot
  data=fitstats;
  title
  "Misclassification Rate for OOB vs Training Data";
  series x=Trees y=MiscAll;
  series x=Trees y=MiscOob/
lineattrs=(pattern=shortdash thickness=
2
);
  yaxis
  label
  ='Misclassification Rate';
run
;
title1 "RANDOM FORESTS ANALYSIS OF SAWTOOTH NRA DATA";
title2 "Using All Predictor Variables, Partitioned Data";
title3 ;
proc
hpsample
  data=Glass out=SNRApartition seed
  =
5461
  partition sampct=
30
;
  class
  GlassType;
  var Aluminum Barium Calcium Iron Magnesium Potassium Refindex Silicon Sodium;
run
;
data
  SNRAtrain SNRAtest;
  set SNRApartition;
  if _PartInd_ eq '0' then
output
  SNRAtrain;
  else
if _PARTIND_ eq '1'
then output
  SNRAtest;
  drop
  _PARTIND_ ;
run
;
Options pageno=1 nodate; run;
title1 "FITTING RANDOM FORESTS TO Glass DATA" ;
title2 "Using Glass as Training Data and pilotI as Validation Data" ;
title3 ;
proc hpforest data=SNRAtrain maxtrees=200 inbagfraction=0.6 scoreprole=oob
seed=5461;
input Calcium Iron Potassium Refindex Silicon Sodium Barium Magnesium Aluminum/ level=interval;
score out=Glasspred;
save file='/home/greenberg29390/rfglass.txt';
target GlassType / level=nominal;
run;

title3 "Out-of-bag Confusion Matrix";
proc freq data=GlassPred;
tables GlassType*I_GlassType;
run;
Options pageno=1 nodate; run;
title1 "FITTING RANDOM FORESTS TO Glass DATA" ;
title2 "Using Glass as Training Data and pilotI as Validation Data" ;

```

```

title3 ;
proc hpforest data=SNRAtrain maxtrees=200 inbagfraction=0.6 scoreprole=oob
seed=5461;
input Aluminum Barium Magnesium Calcium Refindex/ level=interval;
score out=Glasspred;
save file='/home/greenberg29390/rfglass.txt';
target GlassType / level=nominal;
run;

```

```

title3 "Out-of-bag Confusion Matrix";
proc freq data=GlassPred;
tables GlassType*I_GlassType;
run;

```

```

data Nest1;
set nest;
NumTreelt1in = log(1+NumTreelt1in);
NumTree1to3in = log(1+NumTree1to3in);
NumTree3to6in = log(1+NumTree3to6in);
NumTree6to9in = log(1+NumTree6to9in);
NumTree9to15in = log(1+NumTree9to15in);
NumTreegt15in = log(1+NumTreegt15in);
NumSnags = log(1+NumSnags);
NumDownSnags = log(1+NumDownSnags);
NumConifer = log(1+NumConifer);
PctShrubCover = log(1+PctShrubCover);

```

```

run;
data Nest2;
set nest1;
lNumTreelt1in = (NumTreelt1in);
lNumTree1to3in = (NumTree1to3in);
lNumTree3to6in = (NumTree3to6in);
lNumTree6to9in = (NumTree6to9in);
lNumTree9to15in = (NumTree9to15in);
lNumTreegt15in = (NumTreegt15in);
lNumSnags = (NumSnags);
lNumDownSnags = (NumDownSnags);
lNumConifer = (NumConifer);
lPctShrubCover = (PctShrubCover);

```

```

run;
proc sort data=Nest2 out=Nest2;
by nest descending species;
run;
title1 "Summary Statistics and Plots for TRANSFORMED Nest Data" ;

```

```

proc univariate normal plots data=Nest2 noprint;
var l: ;
by Nest;

```

```

run;
/*
title1 'FITTING A CLASSIFICATION TREE TO Nest DATA';
title2 'Selection of Tree Size. Random Seed set to 123';
title3 ;
proc hpsplit data=Nest2 cvmethod=random(10) seed=123 cvmodel=fit plots=all;
class nest Species;
model Species =
NumTreelt1in
NumTree1to3in NumTree3to6in NumTree6to9in
NumTree9to15in NumTreegt15in NumSnags
NumDownSnags PctShrubCover NumConifer
StandType ;
grow gini;
prune costcomplexity;
run;
*/

```

```

Options pageno=1 nodate; run;
title1 "FITTING RANDOM FORESTS TO Nest DATA" ;
title2 "Using Nest as Training Data and pilotI as Validation Data" ;
title3 ;
proc hpforest data=Nest maxtrees=200 inbagfraction=0.6 scoreprole=oob
seed=5461;
input NumTreelt1in
NumTree1to3in NumTree3to6in NumTree6to9in
NumTree9to15in NumTreegt15in NumSnags
NumDownSnags PctShrubCover NumConifer / level=interval;
score out=Speciespred;
save file='/home/greenberg29390/rfspecies.txt';

```

```

input StandType / level=binary;
target Species / level=nominal;
run;

title3 "Out-of-bag Confusion Matrix";
proc freq data=SpeciesPred;
tables Species*I_Species;
run;

/*
title1 'FITTING A CLASSIFICATION TREE TO Nest DATA';
title2 'Selection of Tree Size. Random Seed set to 123';
title3 ;
proc hpsplit data=Nest2 cvmethod=random(10) seed=123 cvmodel=fit plots=all;
class nest;
model Nest (event='1') =
    NumTreelt1in
    NumTree1to3in NumTree3to6in NumTree6to9in
    NumTree9to15in NumTreegt15in NumSnags
    NumDownSnags PctShrubCover NumConifer
    StandType ;
grow gini;
prune costcomplexity;
run;
*/
Options pageno=1 nodate; run;
title1 "FITTING RANDOM FORESTS TO Nest DATA" ;
title2 "Using Nest as Training Data and pilotI as Validation Data" ;
title3 ;
proc hpforest data=Nest maxtrees=200 inbagfraction=0.6 scoreprole=oob
seed=5461;
input NumTreelt1in
NumTree1to3in NumTree3to6in NumTree6to9in
NumTree9to15in NumTreegt15in NumSnags
NumDownSnags NumConifer PctShrubCover / level=interval;
input StandType / level=binary;
score out=Nestpred;
save file='/home/greenberg29390/rfnest.txt';
target Nest / level=nominal;
run;

title3 "Out-of-bag Confusion Matrix";
proc freq data=Nestpred;
tables Nest*I_Nest;
run;

ods graphics off;
ods rtf close;

```