A Project-based Lab Report On

FAILURE ANALYSIS OF PARAMETER-INDUCED SIMULATION CRASHES IN CLIMATE MODELS

Submitted in partial fulfillment of the requirements for

the award of the degree of

Bachelor of Technology

In

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

By

D.GUNASRI JYOTHI SAI	**160030312**
G. MANIKANTA	**160030370**
M.LOHITHA	**160030861**

Under the supervision of

HARSHITHA PATEL

ACADEMIC YEAR 2018-2019

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

K L E F

GREEN FIELDS, VADDESWARAM, DIST. GUNTUR

DECLARATION

The Project based Lab Report for Machine Learning(15CS4171) entitled "**FAILURE ANALYSIS OF PARAMETER-INDUCED SIMULATION CRASHES IN CLIMATE MODELS"** is a record of bonafied work of **D.GUNASRI JYOTHI SAI (160030312) G.MANIKATA(160030370)M.LOHITHA(160030861)** submitted in partial fulfilment for the award of B.Tech in **"Computer Science and Engineering**" in K L E F. The results embodied in this report have not been copied from any other departments/University/Institute.

D.GUNASRI JYOTHI SAI	**160030312**
G.MANIKANTA	**160030370**
M.LOHITHA	**160030861**

K L E F

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATEE

This is to certify that the Machine Learning (15CS4171) Project Based Lab Report entitled "**FAILURE ANALYSIS OF PARAMETER-INDUCED SIMULATION CRASHES IN CLIMATEMODELS"** is being submitted by **D.GUNASRI JYOTHI SAI (160030312) G.MANIKATA(160030370)M.LOHITHA(160030861)** in partial fulfilment for the award of B.Tech **"Computer Science and Engineering**" to K L E F is a record of bonafied work carried out under efficient guidance and supervision. The results embodied in this report have not been copied from any other departments/University/Institute

.

Signature of the Guide **Signature of the HOD**

ACKNOWLEDGEMENT

Our sincere thanks **MS.HARSHITHA PATEL** to for their outstanding support throughout the for the successful completion of the work.

We express our gratitude to **Mr.V.Harikiran**, Head of the Department for Computer Science and Engineering, CSE for providing us with adequate facilities, ways and means by which we are able to complete this project based Lab.

We would like to place on record the deep sense of gratitude to the honourable Vice Chancellor, K L University for providing the necessary facilities to carry the concluded project based lab.

Last but not the least, we thank all Teaching and Non-Teaching Staff of our department and especially our classmates and our friends for their support in the completion of our work.

D.GUNASRI JYOTHI SAI 160030312

G.MANIKANTA 160030370

M.LOHITHA 160030861

INDEX

S.NO	TITLE	PAGE NO
1	ABSTARCT	6
2	INTRODUCTION	7
3	LITERATURE SURVEY	8
3.1	PAPER 1	8
3.2	PAPER 2	9
3.3	PAPER 3	10-11
3.4	PAPER 4	12
4	SOURCE CODE	13
5	RESULTS AND DISCUSSION	14-16
6	METHODOLOGY	17
7	CONCLUSION	18
8	FUTURE WORK	19
9	REFERENCES	20

ABSTRACT

Climate model is constructed based on parameter values taken from atmosphere, oceans, land, and other reservoirs of the Earth system. Here we construct classification to predict simulation outcomes (fail or succeed) from input parameter values, and to use sensitivity analysis and feature selection to determine the causes of simulation crashes.

For constructing the classification model, we use cart algorithm caret and CA Tools packages and build the Decision tree. 46 out of the 540 simulations failed for numerical reasons at combinations of parameter values. We have to predict the attributes which are leading the failure in decision tree by analysing the probability value which are more for attributes which lead to failure.

INTRODUCTION

Modern global three-dimensional climate models are extraordinarily complex pieces of science and software engineering. They contain over a million lines of code and use hundreds to thousands of files, functions, and subroutines to solve equations of state and conservation laws for the flows of matter, energy, and momentum within and between the atmosphere, oceans, land, and other reservoirs of the Earth system. To compound this complexity, these algorithms operate across many orders of magnitude in space and time, and contain constituents that exist in gas, liquid, solid and mixed phases. Given this enormous range of scientific complexity, climate models are vulnerable to many types of software design and implementation issues. We report here on a series of simulation crashes encountered while running perturbed parameter UQ ensembles of the Community Climate System Model Version 4 (CCSM4).Numerous studies have applied UQ techniques to climate models similar to CCSM4.As climate models and other geo-scientific codes become more complex and UQ studies more commonplace, we fully expect parameter-induced simulation crashes to occur in these models with a greater frequency. Our failure analysis method will be benificial for quantifying and determining the causes of these crashes.

LITERATURE REVIEW

PAPER1

Ensemble yield simulations: crop and climate uncertainties, sensitivity to temperature and genotypic adaptation to climate change

-Andrew Juan Challinor

Firstly, observed and simulatedyields in the baseline climate were compared.

Secondly, the response of yield to changes in mean temperature was examined and compared to that found in the literature. No consistent response to temperature change was found across studies.

Thirdly, the relative contribution of uncertainty in crop and climate simulation to the total uncertainty in projected yield changes was examined. In simulations without genotypic adaptation, most of the uncertainty came from the climate model parameters.

Results:

suggest that the germplasm for complete adaptation of groundnut cultivation in western India to a doubled-CO_2 environment may not exist. In conjunction with analyses of germplasm and local management practices, results such as this can identify thegenetic resources needed to adapt to climate change.

PAPER 2

Detection of Climate Crashes using Fuzzy Neural Networks

-Rahib H.Abiyev,

Fuzzy neural networks (FNN) based on Takagi-Sugeno-Kang (TSK) typefuzzy rule is presented to determine chances of failure of the climate models. For this purpose, the parameters characterising the climate crashes in the simulation are used. For comparative analysis, Support Vector Machine (SVM) is applied for simulation of the same problem.The FNN model was discovered to be having better performance in modelling climate crashes.

The Fuzzy neural networks (FNN) model conducts a fuzzy reasoning process using the neural network structure [19]-[22]. Here, problem is to determine the accurate values of the parameters of the FNN model.

Methodology:

The input layer (block) is used for distributing of the coming xi signals

In next block the membership degrees of input signal for ach linguistic value are calculated. Linguistic values are represented by Gaussian membership functions that are characterized by the width and center parameters. The output signals of the rule layer are computed through the use of t-norm min (AND) operation,where, pi is the min operation. These □ j(x) signals are input signals for the output layer. The consequent layer includes n linear systems. In this layer, at first the values of the rules' output are determined as the output signals of the rule layer are multiplied by the output signals of the consequent layer. After calculating the output signal, the training of the parameters of the network starts.

Results:

S.no	TypesSVM	Crossvalidation	Accuracy
1	Linear	10-fold	92.1%
2	Quadratic	10-fold	93.1%
3	Cubic	10-fold	97.4%
4	Medium guassian	10-fold	93.4%
5	Fine guassian	10-fold	91.7%
6	Coarse guassian	10-fold	91.5%

PAPER 3

Error Reduction and Convergence in Climate Prediction

-CHARLES

S. JACKSON

Although climate models have steadily improved their ability to reproduce the observed climate, over the years there has been little change to the wide range of sensitivities exhibited by different models to a doubling of atmospheric CO2 concentrations.In global climate models (GCMs), unresolved physical processes are included through simplified representations referred to as parameterizations. Parameterizations typically contain one or more adjustable phenomenological parameters. Parameter values can be estimated directly from theory or observations or by "tuning" the models by comparing model simulations to the climate record**.**

Methodology:

Experiment design

Each experiment testing the sensitivity of CAM3.1 to combined changes in select parameters follows an experimental design in which the model is forced by observed sea surface temperatures.

Observational constraints

Observational constraints include satellite, instrumental, and reanalysis data products. The fields selected were chosen because of the existence of corresponding instrumental or reanalysis data products; they provide good constraints on top of the atmosphere and surface energy budgets, and they are fields that are commonly used to evaluate model performance.

Definition of cost function

The cost function used to evaluate modelfollows the treatmentin which squared differences between model predictions and observations are projected onto a truncated set of empirical orthogonal functions representing larger spatial regions of correlated year-to-year variability

Renormalization factor "S"

To select candidate model configurations that represent the intended uncertainties, one needs the cost function to be normalized with respect to these uncertainties.That is, the MVFSA algorithm is designed to searchthroughcandidateparametersetsthatarewithin a certain cost function distance from the global minimum, passing over places that are notably badly performing and searching more thoroughly where the performance is acceptable.

Results:

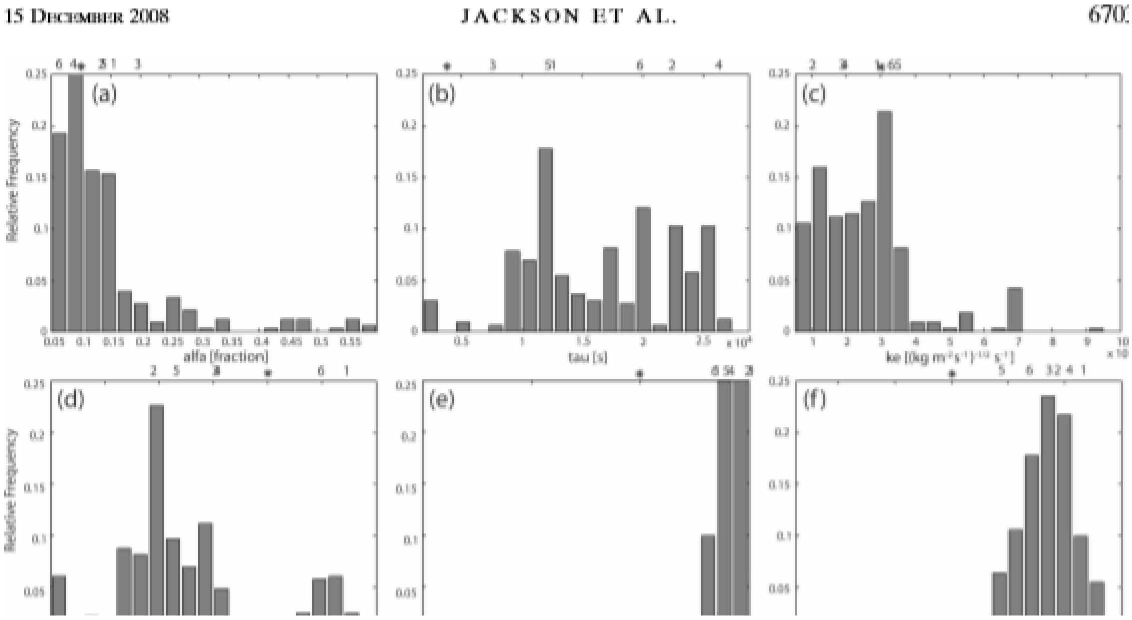


Fig.3.3

PAPER 4

Application of all-relevant feature selection for the failure analysis of parameter-induced simulation crashes in climate models

-Wiesław Paja1

The development of realistic models of climate is one of the most important areas of research due to the dangers posed by global warming.It is by no means a trivial task sinceit involves the parameterisation of many processes that are not directly solved within the model.It has been shown that certain combinations of these parameters lead to failure of a model, despite each individual parameter having a reasonable value.

Methodology:

Random Forest is an ensemble algorithm based on decision trees. To ensure the low correlation between elementary learners, each tree is grown using a different random subsample of the original data set. Moreover, each split in the tree is built using only a random subset of the predictor variables. A big advantage of the algorithm is that it estimates both the classification error and the importance of variables by internal cross validation. To estimate the latter, it measures how much the accuracy of base learners is decreased when information about the variable in question is removed from the system.

The Boruta algorithm infers features' relevance using the estimate of their importance from Random Forest. To this end it extends the information system by variables that arenon-informative by design – the so-called contrast variables.

Results:

a et al.: Feature selection in failure analysis of climate simulation crashes

. Summary of results. The variables indicated as important by Lucas et al. (2013) are marked with ∗; the va rtant in the first test are highlighted in bold face. Δ(AUC) is given in 0.0001 units. Three values are reporte : was deemed relevant, mean difference in AUC due to adding variable to set of variables and number of ng variable to set of variables. The first value is reported for all variables; the two others are reported only relevant significantly more often than randomised variables. The unit for Δ(AUC) is 0.0001.

Variable	**V1***	**V2***	V3	**V4***	V5*	V6	Referen
No. relevant	**660**	**660**	0	**44**	19	33	25 ± 9
Mean Δ(AUC)	**905 ± 80**	**749 ± 90**	–	20 ± 70	–	–	
No. improved	**30**	**30**	–	16	–	–	
Variable	V7	V8	**V9**	V10	V11	V12	Referen
No. relevant	2	17	**62**	11	3	5	25 ± 9
Mean Δ(AUC)	–	–	60 ± 70	–	–	–	
No. improved	–	–	**22**	–	–	–	
Variable	**V13***	**V14***	V15	**V16***	V17*	V18	Referen

Fig 3.4

Source Code:

```
setwd("C:/Users/Personal/Documents")
d1=read.table("climate.dat",header=TRUE)
options(max.print = 540)
d1
names(d1)
is.na(d1)
table(is.na(d1))
library(caTools)
library(caret)
intrain=createDataPartition(y=d1$outcome,p=0.7,list = FALSE)
training=d1[intrain,]
testing=d1[-intrain,]
dim(testing)
dim(training)
library(rpart.plot)
library(rpart)
trctrl=trainControl(method="repeatedcv",number = 2,repeats = 2)
set.seed(3333)
dtree=train(outcome~., data = d1, method="rpart",parms=list(split="gini"),
trControl=trctrl,tuneLength=10)
names(d1)[21]=paste("outcome")
names(d1)
d1
prp(dtree$finalModel)
```

RESULTS AND DISCUSSION

```
Console   Terminal
C:/Users/javhu/Desktop/ML Project/
> library(e1071)
> mydata=read.csv(file.choose())
> str(mydata)
'data.frame':	540 obs. of  21 variables:
 $ study                : int  1 1 1 1 1 1 1 1 1 1 ...
 $ Run                  : int  1 2 3 4 5 6 7 8 9 10 ...
 $ vconst_corr          : num  0.859 0.606 0.998 0.783 0.406 ...
 $ vconst_2             : num  0.928 0.458 0.373 0.104 0.513 ...
 $ vconst_3             : num  0.2529 0.3594 0.5174 0.1975 0.0618 ...
 $ vconst_4             : num  0.299 0.307 0.505 0.422 0.636 ...
 $ vconst_5             : num  0.171 0.843 0.619 0.742 0.845 ...
 $ vconst_7             : num  0.736 0.935 0.606 0.491 0.442 ...
 $ ah_corr              : num  0.42833 0.44457 0.74623 0.00553 0.19193 ...
 $ ah_bolus             : num  0.568 0.828 0.196 0.392 0.488 ...
 $ slm_corr             : num  0.474 0.297 0.816 0.01 0.359 ...
 $ efficiency_factor    : num  0.246 0.617 0.679 0.471 0.552 ...
 $ tidal_mix_max        : num  0.104 0.976 0.803 0.598 0.744 ...
 $ vertical_decay_scale: Factor w/ 540 levels "0.001281531",..: 313 329 232 275 113 222 345 252 142 240 ...
 $ convect_corr         : Factor w/ 540 levels "0.00433221","0.004825306",..: 360 305 259 131 235 8 252 320 92 136 ...
 $ bckgrnd_vdc1         : num  0.449 0.864 0.925 0.913 0.522 ...
 $ bckgrnd_vdc_ban      : num  0.3073 0.3467 0.3154 0.978 0.0435 ...
 $ bckgrnd_vdc_eq       : num  0.858 0.357 0.251 0.846 0.377 ...
 $ bckgrnd_vdc_psim     : num  0.797 0.438 0.286 0.699 0.28 ...
 $ Prandtl              : num  0.87 0.512 0.366 0.476 0.132 ...
 $ outcome              : int  0 1 1 1 1 1 1 1 1 1 ...
> dim(mydata)
[1] 540  21
> mydata$outcome=factor(mydata$outcome,levels = c(0,1),labels = c('Fail','Succeed'))
> mydata=mydata[3:21]
> str(mydata)
'data.frame':	540 obs. of  19 variables:
 $ vconst_corr          : num  0.859 0.606 0.998 0.783 0.406 ...
 $ vconst_2             : num  0.928 0.458 0.373 0.104 0.513 ...
 $ vconst_3             : num  0.2529 0.3594 0.5174 0.1975 0.0618 ...
 $ vconst_4             : num  0.299 0.307 0.505 0.422 0.636 ...
 $ vconst_5             : num  0.171 0.843 0.619 0.742 0.845 ...
 $ vconst_7             : num  0.736 0.935 0.606 0.491 0.442 ...
 $ ah_corr              : num  0.42833 0.44457 0.74623 0.00553 0.19193 ...
```

```
Console   Terminal
C:/Users/javhu/Desktop/ML Project/
> dim(mydata)
[1] 540  19
> tindex = sort(sample(nrow(mydata), nrow(mydata)*.7))
> mtrain=mydata[tindex,]
> mtrain
   vconst_corr    vconst_2    vconst_3    vconst_4    vconst_5    vconst_7
2  0.606041025 0.457728363 0.359448423 0.306957377 0.843330767 0.934850661
3  0.997599777 0.373238487 0.517399356 0.504992546 0.618903336 0.605570823
4  0.783407859 0.104055314 0.197532695 0.421837159 0.742055668 0.490827882
5  0.406249531 0.513199297 0.061811577 0.635836719 0.844797517 0.441502227
7  0.161050068 0.548838498 0.153583456 0.654415397 0.140346213 0.796645887
8  0.415298986 0.898731363 0.931821763 0.916647939 0.399105638 0.009445115
10 0.655626079 0.413930717 0.805288478 0.163485864 0.861901695 0.947594532
12 0.881929079 0.424916157 0.903222367 0.173305914 0.791005372 0.476183524
13 0.961010262 0.976906806 0.857928773 0.614971227 0.615515929 0.352769204
14 0.172529525 0.013556374 0.623446508 0.519135859 0.254456363 0.366947346
15 0.396513559 0.201241250 0.968107640 0.836439982 0.609976638 0.244681982
17 0.814356436 0.696773866 0.702882553 0.975635677 0.983484815 0.031136722
19 0.453287468 0.227814194 0.868963932 0.839546770 0.133349425 0.951339573
20 0.358677493 0.539763975 0.408204473 0.391341766 0.660984427 0.730276729
21 0.795198342 0.129786545 0.881824612 0.531266301 0.546996702 0.626595940
22 0.500682859 0.360446702 0.939109798 0.149994564 0.579039918 0.297490247
24 0.294863722 0.817869572 0.548480255 0.983250152 0.475169340 0.672254588
26 0.773996627 0.029994884 0.500944728 0.181440099 0.501907079 0.017164768
28 0.531351045 0.148787195 0.345090002 0.085156677 0.737006982 0.345312469
29 0.305038056 0.240670239 0.830738987 0.021990788 0.998549574 0.294418373
33 0.678596222 0.115738127 0.529137065 0.188375472 0.418222120 0.333327638
34 0.742689086 0.420645437 0.431974905 0.686290600 0.043077611 0.041824483
35 0.014698195 0.944565988 0.245755275 0.680600516 0.967000433 0.449789209
36 0.198483945 0.436048952 0.668261273 0.995954859 0.109088587 0.034916367
38 0.942360705 0.758718331 0.475841498 0.833072592 0.093249780 0.161020770
40 0.122877340 0.604438373 0.491998566 0.117148366 0.403813848 0.591458200
41 0.809336176 0.252251145 0.554051210 0.056498849 0.569248970 0.053863752
43 0.319991405 0.500185766 0.542541513 0.578932290 0.939126686 0.187563712
45 0.132895597 0.961589490 0.514782509 0.819531342 0.288644342 0.528931266
46 0.366113848 0.162144608 0.559230181 0.888008265 0.334936440 0.089915759
47 0.115795248 0.761214168 0.344103065 0.777035822 0.158458300 0.664698656
48 0.862134440 0.390890688 0.599260395 0.854088527 0.771460458 0.060717765
51 0.347202039 0.023073613 0.665625899 0.967062064 0.378127830 0.336371063
56 0.896376788 0.195539942 0.461462157 0.709033515 0.880215547 0.323155789
```

Console Terminal

C:/Users/javhu/Desktop/ML Project/

```
> mtest=mydata[-tindex,]
> mtest
    vconst_corr    vconst_2    vconst_3    vconst_4   vconst_5   vconst_7
1   0.859036206 0.927824536 0.252865622 0.298838311 0.17052130 0.73593604
6   0.041379470 0.629025938 0.303380105 0.813407572 0.22281712 0.97120603
9   0.166757878 0.352971856 0.988120611 0.287070317 0.56362560 0.40270787
11  0.590007977 0.293692412 0.423497844 0.329803463 0.45783770 0.82978230
16  0.051620363 0.310912551 0.909292558 0.719117468 0.43316708 0.99714175
18  0.595624121 0.791531254 0.794122938 0.430667284 0.91677814 0.89778542
23  0.029453696 0.675724624 0.024930062 0.113832300 0.83368555 0.26819104
25  0.483920717 0.272825273 0.288013515 0.424075319 0.82586679 0.68672840
27  0.420277396 0.371871862 0.742211770 0.063045151 0.18915988 0.84613910
30  0.250101395 0.734771845 0.937692761 0.892969553 0.23556192 0.98242049
31  0.384082258 0.883148776 0.029280328 0.359466858 0.35738168 0.52626826
32  0.715208537 0.329158344 0.576084515 0.022301076 0.38434227 0.82269380
37  0.492837052 0.236896454 0.636875173 0.541799712 0.06886407 0.86034702
39  0.403572912 0.624181003 0.352266838 0.691971240 0.21477363 0.81458378
42  0.931779857 0.534487388 0.072350192 0.508043944 0.79735064 0.85494172
44  0.675576680 0.904870744 0.965466725 0.763904438 0.63355890 0.75930509
49  0.728289293 0.783255210 0.959955337 0.054977527 0.42230189 0.18955887
50  0.892135999 0.834264948 0.732401138 0.375062161 0.72027511 0.93929678
52  0.466558712 0.925630406 0.604524127 0.304059528 0.58593656 0.54166009
53  0.602310924 0.822251677 0.914548962 0.264409831 0.02709355 0.69112083
54  0.634575971 0.581643589 0.053601508 0.293927283 0.29938183 0.53421345
55  0.684860702 0.468095664 0.709534028 0.964369270 0.67379786 0.13100168
60  0.224471803 0.998814816 0.660950114 0.795793790 0.20166022 0.64586139
61  0.258905911 0.718117960 0.237975838 0.256836856 0.29049147 0.27904098
62  0.522210710 0.560967689 0.048322547 0.809188923 0.93097033 0.78675690
63  0.616469387 0.741595574 0.781655731 0.033420153 0.23974659 0.14974540
65  0.467026279 0.447649263 0.565571023 0.361593517 0.65201609 0.80162969
66  0.663038986 0.266648661 0.105045812 0.922150028 0.23265798 0.12018655
70  0.554228936 0.640316600 0.179199255 0.943815547 0.96543501 0.90797844
72  0.837178396 0.349208930 0.139125013 0.910652475 0.72387344 0.43619814
74  0.023921181 0.788500446 0.392744969 0.862112079 0.81537588 0.63107498
77  0.288344673 0.943531631 0.797573650 0.803857490 0.63270124 0.22055485
80  0.644259173 0.969021720 0.815632448 0.561209651 0.77810576 0.45434660
85  0.977305168 0.597217740 0.309078482 0.983514424 0.27949978 0.90033483
86  0.473098871 0.979195615 0.256343501 0.011955851 0.55319485 0.55476223
87  0.508078429 0.568952284 0.010988286 0.351661826 0.89574463 0.68248819
91  0.693058250 0.058344211 0.696437962 0.474131481 0.86819490 0.80999119
```

Console Terminal

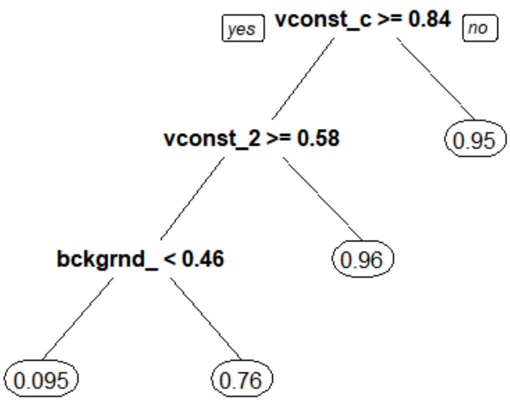
C:/Users/javhu/Desktop/ML Project/

```
> NB=naiveBayes(outcome~.,data=mtrain)
> print(NB)

Naive Bayes Classifier for Discrete Predictors

Call:
naiveBayes.default(x = X, y = Y, laplace = laplace)

A-priori probabilities:
Y
      Fail    Succeed
0.08465608 0.91534392

Conditional probabilities:
         vconst_corr
Y              [,1]      [,2]
  Fail     0.786942 0.1910008
  Succeed 0.478579 0.2893983

         vconst_2
Y               [,1]      [,2]
  Fail     0.7753380 0.1756988
  Succeed 0.4639894 0.2876550

         vconst_3
Y               [,1]      [,2]
  Fail     0.4300499 0.2873949
  Succeed 0.5054232 0.2843132

         vconst_4
Y               [,1]      [,2]
  Fail     0.4288130 0.2811128
  Succeed 0.5039843 0.2840341

         vconst_5
Y               [,1]      [,2]
  Fail     0.4381045 0.2884408
  Succeed 0.5124809 0.2932841

         vconst_7
```

Console Terminal

C:/Users/javhu/Desktop/ML Project/

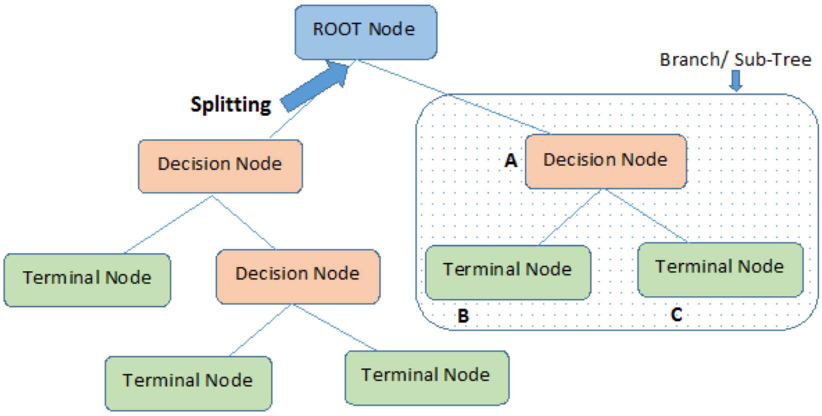
```
> predNB1=predict(NB,mtest)
> table(mtest$outcome,predNB1)
         predNB1
          Fail Succeed
  Fail       1      13
  Succeed    2     146
> plot(predNB1)
> 
```



METHODOLOGY

About the methodology in the project we make use of the dtree algorithm i.e., the decision tree algorithm. Decision tree is one of the most popular machine learning algorithms used all along, This story I wanna talk about it so let's get started!!!

Decision trees are used for both classification and regression problems, this story we talk about classification.



- **Root Node** represents the entire population or sample. It further gets divided into two or more homogeneous sets.
- **Splitting** is a process of dividing a node into two or more sub-nodes.
- When a sub-node splits into further sub-nodes, it is called a **Decision Node**.
- Nodes that do not split is called a **Terminal Node** or a **Leaf**.

Types of Decision trees

- Regression Tree
- Classification Tree

CONCLUSION

Our reanalysis of the results of 540 simulations is in general qualitative agreement with the results of Lucas et al. (2013).The results of the simulation can be predicted with fairly good accuracy using the machine learning approach, and the two different methods give very close results. The crossvalidated AUC reported by Lucas et al. (2013) by ensemble of SVM classifiers was 0.93. In the current study the average of the cross-validated AUC obtained for three strongly important variables was 0.924.

The three most important conclusions for the climate modelling community are the following. Firstly, the efforts on improving the numerical stability of simulations should be concentrated on three parameters of the CCSM4 parallel ocean model, namely vconst_corr, vconst_2 and bckgrnd_vdc1,which were earlier reported as most important by Lucas etal. (2013). The remaining parameters indicated as important in that study are either redundant or not relevant. Secondly,the machine learning methods in general and all-relevant feature selection in particular are useful tools for analysis of influence of simulation parameters on the final outcome. Finally, application of machine learning should involve cross validation, and all important modelling steps should be included in the cross-validation loop.

FUTURE SCOPE

This thesis initiates hybrid classifiers to explore and examine the application of computational intelligent algorithms for solving data mining problems. The optimization techniques are chosen for the classification problems such as data, blur, text and image in this work based on major contributions highlighted as follows:

1. Developing and implementing hybrid optimization algorithm based neural networks for classification problems such as data, text, blur and image classification

2. Testing and verifying the performance measures such as classification accuracy, precision and recall rates of the algorithms on the datasets corresponding to the problem under study

3. Achieving the optimum weight vectors through optimization algorithms for the neural networks hence increasing the computational efficiency and reducing the computational time

REFERENCES

https://docs.google.com/document/d/1aUsRP5lv4YWsQTYC4_YLRdoXFw3TzN547UGAr44OODA/edit

https://docs.google.com/document/d/1Up8dJCO-6JOhJsCIpvmnzm9v-3pwe5q3ncWLaPSjYDQ/edit

https://docs.google.com/document/d/1M56SvV8Shu7iPKmpsTyRKdw3Om5_97lHYXKbHYeTZwo/edit

https://docs.google.com/document/d/1bZmg2RWpi2nyWoM4UCJG92uzq20xQmjmEF_La3ixFhg/edit

https://docs.google.com/document/d/1-P4r46UaYUvhqiTHoZp-zHSBfpScJU8yZEVdcX6zxBI/edit

https://docs.google.com/document/d/1yiiJZ--CHXbtbjT8esMkGWkyIJ_47MYDE33L26g3MPc/edit