

```

/** ----- Set 1 Logistic Regression Model-----
/** FILE NAME:    LR_200_Set1.txt
/** AUTHOR:      Renee Schicker
/** CREATED:     19 NOVEMBER 2009
/** UPDATED:     07 DECEMBER 2009
/**
/** The scripts may be supplied in a more readily useable format if the work is acknowledged
/** CONTACT:     Renee_Schicker@hotmail.com
/**
/** PURPOSE:      To apply the results of the logistic regression (generalised
/**               linear/nonlinear) analysis from STATISTICA to the spatial data
/**               in GIS.
/**
/** HOW:          The beta values applied to each variable included in the logistic
/**               regression result can be obtained as a PMML script in STATISTICA.
/**               These beta values can be taken and applied here. This script allows
/**               the user to specify the number of variables in the equation, for the
/**               purpose of this study there is a maximum of 10 variables (have excluded
/**               roads as these might be related to the reporting of landslides, otherwise
/**               there would be 11).
/**               The user can also specify which variables are included in the equation,
/**               but these must be encased with % signs and spelt exactly as the instructions
/**               say. This is because these relate to the variables which are set in the
/**               CONTROLVARS routine which basically give the filepath for each of these
/**               parameters.
/**               Using the user's specifications the script firstly applies the beta values
/**               to each of the specified parameter spatial (grid/raster) dataset then it
/**               calculates  $Z = B_0 + B_1X_1 + B_2X_2 + \dots + B_nX_n$ . Following this the probability
/**               is calculated. This can either be for the Logit link function or LogLog link
/**               function. May need to comment out the &CALL for the one not required.
/**
/** ----- HISTORY-----
/**
/** 19 NOVEMBER 2009
/** 26 NOVEMBER 2009    Added the Variables at the bottom for files and file paths, and add
/**                     the top lot of variables to link up to them. A series of linked &IF
/**                     statements were added which carry out processes according to the
/**                     number set as TotVars (Total variables in the model). All this
/**                     combined allows more freedom in the script as the user can specify
/**                     the number of predictor variables and which ones and based on this
/**                     input the script will figure out the model.
/**
/** 07 DECEMBER 2009    Changed the SETWINDOW and SETMASK to MyBnds, which should
/**                     clip Lake Taupo out. Added some &TYPE comments to explain the
/**                     processes and equations as the script runs..
/**
/** *****
/** IMPORTANT - RELATES TO FILES AND FILEPATHS - DO NOT CHANGE FOR THIS STUDY
/** *****
/** Set folder to work in and save output to:
Workspace D:\Renee_GIS\Output_data\Organised\15_Multi_Reg\200m_set1

```

```

/* IMPORTANT: DO NOT COMMENT OUT, Set filepath variables (located at end of script):
&CALL CONTROLVARS
/*****
/*****
/**
/**      VARIABLES TO ALTER (INPUT THE OUTPUT FROM EXTERNAL STATISTICAL ANALYSIS):
/**
/*****
/*****

/* SET Number of Variables to use (Must be between 1 and 10)

&SETVAR .TotVars = 4

/* SET Variables (Chose from: Slope, Aspect, Elevation, AvRain, MaxRain, LandUse, Geology, Soil,
Faults, Rivers)
/* Enclose variable name in % signs!! e.g. %Slope%
/* Can comment out those not needed using /*

&SETVAR .Var1 = %Slope%
&SETVAR .Var2 = %AvRain%
&SETVAR .Var3 = %LandUse%
&SETVAR .Var4 = %Geology%
/*&SETVAR .Var5 =
/*&SETVAR .Var6 =
/*&SETVAR .Var7 =
/*&SETVAR .Var8 =
/*&SETVAR .Var9 =
/*&SETVAR .Var10 =

/* SET Beta Values

&SETVAR .B0 = -4.06515399156639          /* (p1) Intercept
&SETVAR .B1 = -0.00684837015901455 /* (p2) Var1 beta
&SETVAR .B2 = 0.0318256285236851 /* (p3) Var2 beta
&SETVAR .B3 = -0.120829226861181 /* (p4) Var3 beta
&SETVAR .B4 = -0.00285868279153378 /* (p5) Var4 beta
/*&SETVAR .B5 =          /* (p6) Var5 beta
/*&SETVAR .B6 =          /* (p7) var6 beta
/*&SETVAR .B7 =          /* (p8) var7 beta
/*&SETVAR .B8 =          /* (p9) var8 beta
/*&SETVAR .B9 =          /* (p10) var9 beta
/*&SETVAR .B10 =         /* (p11) var10 beta

/*****
/**                                PROCESSED TO CALL
/*****
&CALL Kill_Temp

```

```

&CALL Betas
&CALL Z
&CALL Logit          /** If Logit Link Function was Used (comment out /* if not)
/*&CALL LogLog       /** If Log-LOg Link Function was Used (comment out /* if not)
/*&CALL Kill_Temp

&RETURN

/*****
/**
PROCESSSES THAT WILL RUN
*****/
&ROUTINE Betas
&TYPE Calculating Betas.
&TYPE The number of Beta values to be calculated is determined by the Total Variables.
&CALL Setup_Grid

BOGRID = (%Intercept% * %.B0%)
&TYPE INTERCEPT GRID CREATED

&IF %.TotVars% = 1 &THEN
  &CALL Calc_1_Beta
&ELSE

&IF %.TotVars% = 2 &THEN
  &CALL Calc_2_Betas
&ELSE

&IF %.TotVars% = 3 &THEN
  &CALL Calc_3_Betas
&ELSE

&IF %.TotVars% = 4 &THEN
  &CALL Calc_4_Betas
&ELSE

&IF %.TotVars% = 5 &THEN
  &CALL Calc_5_Betas
&ELSE

&IF %.TotVars% = 6 &THEN
  &CALL Calc_6_Betas
&ELSE

&IF %.TotVars% = 7 &THEN
  &CALL Calc_7_Betas
&ELSE

&IF %.TotVars% = 8 &THEN
  &CALL Calc_8_Betas
&ELSE

```

```
&IF %.TotVars% = 9 &THEN  
  &CALL Calc_9_Betas  
&ELSE
```

```
&IF %.TotVars% = 10 &THEN  
  &CALL Calc_10_Betas  
&ELSE  
  &Type number of variables not between 1 and 10
```

```
&TYPE Finished Calculating Betas  
&CALL Exit_Grid  
&RETURN
```

```
/*****
```

```
&ROUTINE Z  
&TYPE Calculating Z By adding the beta adjusted parameter layers  
&TYPE Following the  $Z = B_0 + B_1X_1 + B_2X_2 + \dots + B_nX_n$  Calculation  
&TYPE Where  $B_0$  = Model Intercept,  $B_i$  ( $i = 1, 2, \dots, n$ ) = beta value,  
&TYPE and  $X_i$  ( $i = 1, 2, \dots, n$ ) = a predictor parameter.  
&TYPE The calculation relies on knowing the total variables in order  
&TYPE to determine the number of layers (e.g.  $B_1X_1, B_2X_2, \dots, B_nX_n$ ) to add to  $B_0$ .  
&CALL Setup_Grid
```

```
&IF %.TotVars% = 1 &THEN  
  &CALL 1_Variable  
&ELSE
```

```
&IF %.TotVars% = 2 &THEN  
  &CALL 2_variables  
&ELSE
```

```
&IF %.TotVars% = 3 &THEN  
  &CALL 3_variables  
&ELSE
```

```
&IF %.TotVars% = 4 &THEN  
  &CALL 4_variables  
&ELSE
```

```
&IF %.TotVars% = 5 &THEN  
  &CALL 5_variables  
&ELSE
```

```
&IF %.TotVars% = 6 &THEN  
  &CALL 6_variables  
&ELSE
```

```
&IF %.TotVars% = 7 &THEN  
  &CALL 7_variables
```

&ELSE

&IF %.TotVars% = 8 &THEN

&CALL 8_variables

&ELSE

&IF %.TotVars% = 9 &THEN

&CALL 9_variables

&ELSE

&IF %.TotVars% = 10 &THEN

&CALL 10_variables

&ELSE

&TYPE Error!

&TYPE Either %.TotVars% outside of 1-10 range => Check .TotVars variable

&TYPE Or Script has some other problem

&CALL Exit_Grid

&RETURN

&ROUTINE Calc_1_Beta

B1Grid = (%.Var1% * %.B1%)

&TYPE B1Grid = (%.Var1% * %.B1%) Done

&RETURN

&ROUTINE Calc_2_Betas

&CALL Calc_1_Beta

B2Grid = (%.Var2% * %.B2%)

&TYPE B2Grid = (%.Var2% * %.B2%) Done

&RETURN

&ROUTINE Calc_3_Betas

&CALL Calc_2_Betas

B3Grid = (%.Var3% * %.B3%)

&TYPE B3Grid = (%.Var3% * %.B3%) Done

&RETURN

&ROUTINE Calc_4_Betas

&CALL Calc_3_Betas

B4Grid = (%.Var4% * %.B4%)

&TYPE B4Grid = (%.Var4% * %.B4%) Done

&RETURN

&ROUTINE Calc_5_Betas

&CALL Calc_4_Betas

B5Grid = (%.Var5% * %.B5%)

&TYPE B5Grid = (%.Var5% * %.B5%) Done

&RETURN

&ROUTINE Calc_6_Betas

```

&CALL Calc_5_Betas
B6Grid = (%.Var6% * %.B6%)
&TYPE B6Grid = (%.Var6% * %.B6%)   Done
&RETURN
*****

&ROUTINE Calc_7_Betas
&CALL Calc_6_Betas
B7Grid = (%.Var7% * %.B7%)
&TYPE B7Grid = (%.Var7% * %.B7%)   Done
&RETURN
*****

&ROUTINE Calc_8_Betas
&CALL Calc_7_Betas
B8Grid = (%.Var8% * %.B8%)
&TYPE B8Grid = (%.Var8% * %.B8%)   Done
&RETURN
*****

&ROUTINE Calc_9_Betas
&CALL Calc_8_Betas
B9Grid = (%.Var9% * %.B9%)
&TYPE B9Grid = (%.Var9% * %.B9%)   Done
&RETURN
*****

&ROUTINE Calc_10_Betas
&CALL Calc_9_Betas
B10Grid = (%.Var10% * %.B10%)
&TYPE B10Grid = (%.Var10% * %.B10%)   Done
&RETURN
*****

*****

&ROUTINE 1_variable
Z = (B0GRID + B1Grid)
&TYPE Z = B0 + B1X1   Done
&TYPE Technically this may not be possible in the Statistical package
&RETURN
*****

&ROUTINE 2_variables
Z = (B0GRID + B1Grid + B2Grid)
&TYPE Z = B0 + B1X1 + B2X2   Done
&RETURN
*****

&ROUTINE 3_variables
Z = (B0GRID + B1Grid + B2Grid + B3Grid)
&TYPE Z = B0 + B1X1 + B2X2 + B3X3   Done
&RETURN
*****

&ROUTINE 4_variables
Z = (B0GRID + B1Grid + B2Grid + B3Grid + B4Grid)
&TYPE Z = B0 + B1X1 + B2X2 + B3X3 + B4X4   Done
&RETURN
*****

```

```

&ROUTINE 5_variables
Z = (B0GRID + B1Grid + B2Grid + B3Grid + B4Grid + B5Grid)
&TYPE Z = B0 + B1X1 + B2X2 + ... + B5X5   Done
&RETURN
*****

&ROUTINE 6_variables
Z = (B0GRID + B1Grid + B2Grid + B3Grid + B4Grid + B5Grid + B6Grid)
&TYPE Z = B0 + B1X1 + B2X2 + ... + B6X6   Done
&RETURN
*****

&ROUTINE 7_variables
Z = (B0GRID + B1Grid + B2Grid + B3Grid + B4Grid + B5Grid + B6Grid + B7Grid)
&TYPE Z = B0 + B1X1 + B2X2 + ... + B7X7   Done
&RETURN
*****

&ROUTINE 8_variables
Z = (B0GRID + B1Grid + B2Grid + B3Grid + B4Grid + B5Grid + B6Grid + B7Grid + B8Grid)
&TYPE Z = B0 + B1X1 + B2X2 + ... + B8X8   Done
&RETURN
/*****

&ROUTINE 9_variables
Z = (B0GRID + B1Grid + B2Grid + B3Grid + B4Grid + B5Grid + B6Grid + B7Grid + B8Grid + B9Grid)
&TYPE Z = B0 + B1X1 + B2X2 + ... + B9X9   Done
&RETURN
/*****

&ROUTINE 10_variables
Z = (B0GRID + B1Grid + B2Grid + B3Grid + B4Grid + B5Grid + B6Grid + B7Grid + B8Grid + B9Grid +
B10Grid)
&TYPE Z = B0 + B1X1 + B2X2 + ... + B10X10   Done
&RETURN
/*****

&ROUTINE Logit

&IF [EXIST NegZ -GRID] &THEN KILL NegZ ALL
&IF [EXIST ExpZ -GRID] &THEN KILL ExpZ ALL
&IF [EXIST LogitEq1P -GRID] &THEN KILL LogitEq1P ALL
&IF [EXIST LogitEq2P -GRID] &THEN KILL LogitEq2P ALL
&IF [EXIST LogitEq3P -GRID] &THEN KILL LogitEq3P ALL
&IF [EXIST LogitEq2Int -GRID] &THEN KILL LogitEq2Int ALL
&IF [EXIST LogitEq3Int -GRID] &THEN KILL LogitEq3Int ALL

&CALL Setup_Grid

NegZ = (-1 * Z)

&TYPE Calculating Equation 1:  $\text{EXP}(Z) / (1 + \text{EXP}(Z))$       Idea from Lee et al. (2006) Menard (1995)
ExpZ = Exp(Z)
LogitEq1P = (ExpZ / (1 + ExpZ))

```

&TYPE Calculating Equation 2: $1 / (1 + \text{EXP}(-Z))$ Idea from Can et al. (2005), Suzen & Doyuren (2004),
 &TYPE Lee (2004), Lee & Sambath (2006), Dai et al. (2001), Dai & Lee (2002), Ohlmacher & Davis (2003).

LogitEq2P = $(1 / (1 + \text{EXP}(\text{NegZ})))$
 LogitEq2Int = int((LogitEq2P * 100) + .5)

&TYPE Calculating Equation 3 (Inverse of Equation 2): $1 / (1 + \text{EXP}(Z))$

LogitEq3P = $(1 / (1 + \text{EXP}(Z)))$

LogitEq3Int = int((LogitEq3P * 100) + .5)

&CALL Exit_Grid
 &RETURN

/*****
 &ROUTINE LogLog

&IF [EXIST NegZ -GRID] &THEN KILL NegZ ALL
 &IF [EXIST LogLogEq1P -GRID] &THEN KILL LogLogEq1P ALL
 &IF [EXIST LogLogEq2P -GRID] &THEN KILL LogLogEq2P ALL

&CALL Setup_Grid

NegZ = $(-1 * Z)$

&TYPE Calculating Equation 1: $\text{EXP}(-\text{EXP}(Z))$ Idea from White and Burnham (1999)

LogLogEq1P = $\text{EXP}(-\text{EXP}(Z))$

&TYPE Calculating Equation 2: $\text{EXP}(-\text{EXP}(-Z))$ Idea from Chen and Shao (2000)

LogLogEq2P = $\text{EXP}(-\text{EXP}(\text{NegZ}))$

&CALL Exit_Grid
 &RETURN

/*****
 &ROUTINE Kill_Temp

&IF [EXIST Z -GRID] &THEN KILL Z ALL
 &IF [EXIST B0Grid -GRID] &THEN KILL B0Grid ALL
 &IF [EXIST B1Grid -GRID] &THEN KILL B1Grid ALL
 &IF [EXIST B2Grid -GRID] &THEN KILL B2Grid ALL
 &IF [EXIST B3Grid -GRID] &THEN KILL B3Grid ALL
 &IF [EXIST B4Grid -GRID] &THEN KILL B4Grid ALL
 &IF [EXIST B5Grid -GRID] &THEN KILL B5Grid ALL


```

&IF [EXIST B6Grid -GRID] &THEN KILL B6Grid ALL
&IF [EXIST B7Grid -GRID] &THEN KILL B7Grid ALL
&IF [EXIST B8Grid -GRID] &THEN KILL B8Grid ALL
&IF [EXIST B9Grid -GRID] &THEN KILL B9Grid ALL
&IF [EXIST B10Grid -GRID] &THEN KILL B10Grid ALL

&RETURN
/*****
&ROUTINE Setup_Grid
&RUN D:\renee_gis\scripts\checkproggrid.txt
&TYPE set window to MyBnds...
SETWINDOW D:\Renee_gis\output_data\Organised\03_DEM\MyBnds

&TYPE Set mask to MyBnds...
SETMASK D:\Renee_gis\output_data\Organised\03_DEM\MyBnds
&TYPE Mask Set
&RETURN
/*****
&ROUTINE Exit_Grid
SETMASK OFF
Q
&RETURN
/*****
/*
/*
/*****
/**      Variables to remain constant (Determines Filepaths for variables)
/*****
&ROUTINE CONTROLVARS
&SETVAR .Drive = D:
&SETVAR .F1 = \Renee_GIS
&SETVAR .F2 = \output_data
&SETVAR .F3 = \Organised
&SETVAR .FPMain = %.Drive%%.F1%%.F2%%.F3%
&SETVAR .F4a = \03_DEM
&SETVAR .F4b = \05_Geology
&SETVAR .F4c = \06_Soil
&SETVAR .F4d = \07_Landcover
&SETVAR .F4e = \10_Faults
&SETVAR .F4f = \12_Rivers
&SETVAR .F4g = \13_Rain
&SETVAR .v0 = \dembounds
&SETVAR .v1 = \slopeint
&SETVAR .v2 = \aspectint
&SETVAR .v3 = \dem25int
&SETVAR .v4 = \rain98av
&SETVAR .v5 = \rain98max
&SETVAR .v6 = \lcdb2grid
&SETVAR .v7 = \geolgrid
&SETVAR .v8 = \soilsgrid
&SETVAR .v9 = \Faultgrid

```

&SETVAR .v10 = \Rivergrid

&SETVAR Intercept = %.FPMAIN%.F4a%.v0%

&SETVAR Slope = %.FPMAIN%.F4a%.v1%

&SETVAR Aspect = %.FPMAIN%.F4a%.v2%

&SETVAR Elevation = %.FPMAIN%.F4a%.v3%

&SETVAR AvRain = %.FPMAIN%.F4g%.v4%

&SETVAR MaxRain = %.FPMAIN%.F4g%.v5%

&SETVAR LandUse = %.FPMAIN%.F4d%.v6%

&SETVAR Geology = %.FPMAIN%.F4b%.v7%

&SETVAR Soil = %.FPMAIN%.F4c%.v8%

&SETVAR Faults = %.FPMAIN%.F4e%.v9%

&SETVAR Rivers = %.FPMAIN%.F4f%.v10%

&TYPE Permanent Variables set

&RETURN