Guidelines for computing MaxEnt model output values from a lambdas file

Peter D. Wilson
Research Fellow
Invasive Plants and Climate Project
Department of Biological Sciences
Macquarie University, New South Wales, Australia 2109

email: pwilson@bio.mq.edu.au

This version produced on 9 June 2009

The following notes represent guidelines on how to use a lambdas file generated when fitting a model using the MaxEnt program (available from www.cs.princeton.edu/~schapire/maxent). I developed them as part of my research into the application of species distribution modelling to understanding the response of invasive plants responding to climate change.

Information for developing these guidelines on "parsing" the lambdas file has been pieced together from the fragments found in the MaxEnt helpfile, a post to the MaxEnt Google Groups email list by Miroslav Dudik on 8 Feb 2008, and the mathematical definitions given in several papers including Phillips *et al*. (2006, *Ecological Modelling* 190:231-259), Dudik *et al.* (2007, *Journal of Machine Learning Research* 8:1217-1260), and Phillips and Dudik (2008, *Ecography* 31: 161-175, 2008).

My aim in producing these guidelines was to provide a non-mathematical outline of how the lambdas files are interpreted to generate an output map. It became apparent from a number of posts to the MaxEnt list, and a question from a new MaxEnt user at a recent workshop, that there was no comprehensive and easily accessible explanation of the way lambdas files worked. For mathematically trained readers, there is a series of rigorous mathematical explanations of the MaxEnt algorithm in the papers cited above, but these guidelines might help you understand the way the mathematical principles have been cleverly implemented by Steven Phillips and colleagues. Developing the guidelines has helped me make that transition.

Feature types:

With feature selection set to the default "auto" value, MaxEnt selects a combination of the following feature types when fitting a model: (a) Raw features = untransformed raw data, (b) quadratic features = squared data values, (c) product features = products of two different variables, (d) forward hinge features = a linear "ramp"-shaped function rising from 0 to 1, starting at the hinge value and clamped to 0 when less than the hinge value, (e) reverse hinge features = ramp-function falling from 1 to reach 0 at the hinge value and 0 when the variable is greater than the hinge value, and (f) threshold features = 0 when the variable is less than the threshold and 1 when greater than or equal to the threshold. The key thing to note is that the range of all functions which transform a raw variable prior to multiplication by a lambda-value is 0 and 1.

Format of the lambdas file:

A lambdas file is a simple text file and consists of two parts. The first contains information on features and the second, which is located at the end of the file, gives values for vital constants needed to compute a value for the fitted model. They are:

linearPredictorNormalizer
densityNormalizer
numBackgroundPoints
entropy

The constant "numBackgroundPoints" is not needed for calculating a model output value, but can be useful when analysing raw model output. Raw output values are sensitive to the number of background points (amongst other things).

Here is an example of a rather large lambdas file which includes at least one representative of each feature type:

```
OZ_AnnMeanTemp, 0.0, 65.0, 292.0
OZ_AnnPrecip, 0.0, 130.0, 6458.0
OZ_AnnTempRange, 0.0, 69.0, 343.0
OZ_DryestMonPrecip, 0.0, 0.0, 457.0
OZ_IsoTherm, 0.0, 40.0, 94.0
OZ_MaxTemp, 0.0, 160.0, 418.0
OZ_MeanMonTempRange, 0.0, 52.0, 165.0
OZ_MeanTempColdQtr, 0.41006255001892694, 56.0, 329.0
OZ_MeanTempDryQtr, 0.0, 77.0, 323.0
OZ_MeanTempWarmQtr, -0.6314814577730692, 101.0, 329.0
OZ_MeanTempWetQtr, -0.2416226080583578, 11.0, 329.0
OZ_MinTemp, 0.0, -37.0, 239.0
OZ_PrecipColdQtr, 0.0, 11.0, 1491.0
OZ_PrecipDryQtr, 0.0, 37.0, 2666.0
OZ_PrecipSeason, 0.5331341842923523, 4.0, 140.0
OZ_PrecipWarmQtr, 0.0, 27.0, 1865.0
OZ_PrecipWetQtr, 0.0, 37.0, 2666.0
OZ_TempSeason, 0.0, 1.0, 29.0
OZ_WettestMonPrecip, 0.0, 16.0, 1038.0
OZ_IsoTherm^2, -6.630126631087359, 1600.0, 8836.0
OZ_MeanTempColdQtr^2, 0.35462827441592243, 3136.0, 108241.0
OZ_MeanTempDryQtr^2, -0.552732961475643, 5929.0, 104329.0
OZ_MeanTempWetQtr^2, -0.9665375829580902, 121.0, 108241.0
OZ_AnnMeanTemp*OZ_AnnTempRange, -0.18571309718670556, 11591.0, 81900.0
OZ_AnnMeanTemp*OZ_MaxTemp, -1.1169464841709513, 11375.0, 113588.0
OZ_AnnMeanTemp*OZ_MeanTempColdQtr, -0.312166261384468, 5070.0, 90396.0
OZ_AnnMeanTemp*OZ_MeanTempDryQtr, -0.8150830566984788, 5070.0, 94316.0
OZ_AnnMeanTemp*OZ_MeanTempWarmQtr, -1.1499797974880512, 6955.0, 94316.0
OZ_AnnMeanTemp*OZ_MeanTempWetQtr, -0.7826277317631258, 737.0, 90520.0
OZ_AnnTempRange*OZ_MinTemp, -0.6165748289631581, -8695.0, 37324.0
OZ_AnnTempRange*OZ_PrecipSeason, 1.165382592371033, 360.0, 35712.0
OZ_AnnTempRange*OZ_PrecipWarmQtr, 1.664034482018447, 5549.0, 286560.0
OZ_MeanMonTempRange*OZ_MeanTempWarmQtr, -0.6510099759920648, 8640.0, 50666.0
OZ_MeanMonTempRange*OZ_PrecipSeason, 0.4689479277373668, 328.0, 19096.0
OZ_MeanMonTempRange*OZ_TempSeason, -1.447249616301249, 82.0, 2943.0
OZ_MeanTempColdQtr*OZ_MeanTempWarmQtr, -0.7395940137070127, 8239.0, 108241.0
OZ_MeanTempColdQtr*OZ_MinTemp, -0.88012242760618, -3630.0, 65964.0
OZ_MeanTempColdQtr*OZ_PrecipSeason, 3.854936301783768, 576.0, 37812.0
OZ_MeanTempWarmQtr*OZ_PrecipSeason, 0.0791959280079874, 606.0, 42228.0
```

```
OZ_MeanTempWetQtr*OZ_MinTemp, -0.8121852580972619, -3335.0, 65486.0
OZ_PrecipColdQtr*OZ_PrecipWarmQtr, -6.60890886900382, 729.0, 2273775.0
OZ_PrecipColdQtr*OZ_TempSeason, 2.4409971042947856, 78.0, 16440.0
OZ_PrecipSeason*OZ_PrecipWarmQtr, 2.025563914217865, 572.0, 147159.0
OZ_PrecipSeason*OZ_WettestMonPrecip, -1.0672304718670045, 252.0, 70584.0
(215.5<OZ_MeanTempDryQtr), -0.07180028724433321, 0.0, 1.0
(211.5<OZ_AnnMeanTemp), -0.9965095294482207, 0.0, 1.0
(30.5<OZ_WettestMonPrecip), 1.5763178766430996, 0.0, 1.0
(143.5<OZ_AnnTempRange), 0.04014999184152795, 0.0, 1.0
(100.5<OZ_MinTemp), -0.41746462344659063, 0.0, 1.0
(252.5<OZ_AnnTempRange), -0.3348598825308529, 0.0, 1.0
(403.5<OZ_PrecipColdQtr), -1.019691227730407, 0.0, 1.0
(47.5<OZ_PrecipColdQtr), 0.956435448927318, 0.0, 1.0
(158.5<OZ_AnnTempRange), 0.8851880477847027, 0.0, 1.0
(285.5<OZ_MeanTempWarmQtr), -0.40116489471837397, 0.0, 1.0
(113.5<OZ_MeanTempColdQtr), 5.062961101998734E-4, 0.0, 1.0
(325.5<OZ_AnnPrecip), -0.5749882640449281, 0.0, 1.0
(357.5<OZ_PrecipWarmQtr), 0.34395370708194317, 0.0, 1.0
(259.5<OZ_AnnPrecip), 0.6170740879495593, 0.0, 1.0
(5.5<OZ_MinTemp), 0.6659820081013705, 0.0, 1.0
(49.5<OZ_IsoTherm), -0.007490796989380762, 0.0, 1.0
(219.5<OZ_MaxTemp), 0.5119906196018428, 0.0, 1.0
(229.5<OZ_MeanTempWetQtr), -0.023344223032753124, 0.0, 1.0
`OZ_PrecipWarmQtr, 0.9853404287722853, 27.0, 31.5
(91.5<OZ_PrecipSeason), -1.0825976826537356, 0.0, 1.0
`OZ_PrecipWarmQtr, 1.0853876268425875, 27.0, 33.5
(48.5<OZ_IsoTherm), -0.11842567675949421, 0.0, 1.0
(113.5<OZ_MeanTempDryQtr), 0.23741394412892444, 0.0, 1.0
'OZ_TempSeason, 0.48730037660010167, 28.0, 29.0
(8.5<OZ_DryestMonPrecip), -0.2825552684856313, 0.0, 1.0
`OZ_DryestMonPrecip, -1.1809735919925275, 0.0, 5.5
`OZ_TempSeason, 0.27024722786753674, 1.0, 5.5
(224.5<OZ_MeanTempDryQtr), -0.3338177740835147, 0.0, 1.0
(101.5<OZ_AnnMeanTemp), 0.259757910759768, 0.0, 1.0
(1258.5<OZ_AnnPrecip), 0.024769519510349308, 0.0, 1.0
'OZ_MeanMonTempRange, 0.39118644073873, 153.5, 165.0
`OZ_IsoTherm, -0.15698415026967427, 40.0, 46.5
'OZ_TempSeason, 0.30474953337264094, 22.5, 29.0
(278.5<OZ_MeanTempColdQtr), -0.5980705552815578, 0.0, 1.0
'OZ_AnnTempRange, 0.41789959647201735, 315.5, 343.0
(25.5<OZ_MinTemp), -7.535170548599764E-4, 0.0, 1.0
(141.5<OZ_MeanTempWarmQtr), 0.21248960575987447, 0.0, 1.0
'OZ_AnnTempRange, 1.0590642462330486, 312.5, 343.0
(12.5<OZ_PrecipSeason), -0.1601442671386823, 0.0, 1.0
(82.5<OZ_PrecipSeason), -0.2758448473185918, 0.0, 1.0
`OZ_PrecipWarmQtr, 1.2583973591550064, 27.0, 34.5
`OZ_PrecipColdQtr, -2.0507114707094916, 11.0, 49.5
(44.0<OZ_MeanTempWetQtr), 0.14018432208683881, 0.0, 1.0
'OZ_TempSeason, 0.21356227691610233, 25.5, 29.0
(68.5<OZ_MeanTempWetQtr), 0.22530982256126708, 0.0, 1.0
(149.5<OZ_MeanTempDryQtr), 0.2155770483305779, 0.0, 1.0
'OZ_TempSeason, 0.18621068799975768, 19.5, 29.0
`OZ_IsoTherm, -1.6037183216430737, 40.0, 47.5
(199.5<OZ_MaxTemp), 0.20337972002747934, 0.0, 1.0
(1078.5<OZ_AnnPrecip), 0.012377022984124547, 0.0, 1.0
(51.5<OZ_IsoTherm), 0.06628635227731448, 0.0, 1.0
(263.5<OZ_AnnPrecip), 0.027969395068903, 0.0, 1.0
(11.5<OZ_PrecipSeason), -0.06315644136243079, 0.0, 1.0
`OZ_DryestMonPrecip, -0.2456443177330928, 0.0, 2.5
(45.5<OZ_PrecipWarmQtr), 0.02805366451886604, 0.0, 1.0
linearPredictorNormalizer, 4.846340368471091
```

```
densityNormalizer, 225.55340683425715
numBackgroundPoints, 10000
entropy, 7.692884701802406
```

Note the complexity of the fitted model in terms of the appearance of a variable name in many different feature-types, and/or the appearance of the same variable in several versions of the same feature type. Also, the fitted lambda for a raw feature might be 0 (i.e. the raw values of the variable do not enter into the fitted model directly) but the variable might be part of one of more quadratic or product features so that it makes an indirect contributions to the final value. As Steven Phillips and collaborators have noted (and see explanations in the MaxEnt output and the tutorial), all this makes calculating the contribution of a variable to the fitted model less than straight-forward particularly when there are strong correlations between variables. The bar charts fo variable contributions produced using jackknifed model runs that can be generated by MaxEnt to guide variable importance assessment are the best tool for this task, and you should follow the advice given by Steven Phillips in the MaxEnt output and the tutorial to interpret these bar charts.

The reason that the same variable might appear in several versions of the same feature type (each time associated with a different lambda, max and min value) is that this allows the response to a variable to be modelled in a piece-wise fashion. This mechanism allows very complex and close-fiting relationships to be produced, which can be seen in the response plots produced in the MaxEnt output when the appropriate option is selected. To illustrate how the multiple versions of a feature type work together, consider the following collection of threshold features for the variable OZ_AnnPrecip:

```
(325.5<OZ_AnnPrecip), -0.5749882640449281, 0.0, 1.0
(259.5<OZ_AnnPrecip), 0.6170740879495593, 0.0, 1.0
(1258.5<OZ_AnnPrecip), 0.024769519510349308, 0.0, 1.0
(1078.5<OZ_AnnPrecip), 0.012377022984124547, 0.0, 1.0
(263.5<OZ_AnnPrecip), 0.027969395068903, 0.0, 1.0
```

When OZ_AnnPrecip is less than 259.5, that feature contributes 0 to the total value but 0.6170740879495593 when greater than or equal to this threshold. This is added to (or subtracted from, depending on the sign of each lambda) by each other threshold feature in this group as its threshold is crossed so that the total impact is a stepped function with the width of the steps determined by the threshold values and the height of each step set by the sum of lambda values in play for a given value of the variable. When this ensemble of threshold features is combined with other feature types involving OZ_AnnPrecip, the influence of OZ_AnnPrecip on the output can be modelled as a very complicated, close-fitting function.

Features:

A feature entry is always composed of four parts separated by commas. The interpretation of the four parts varies between feature types. The first part gives information on the feature type, the variable or variables involved in the feature and, in the case of threshold features, the threshold value. The second part is *always* the fitted lambda-value. The third and fourth parts are always numeric values but their meaning varies between feature types.

(a) Raw features:

```
OZ_MeanTempColdQtr, 0.41006255001892694, 56.0, 329.0
```

The four parts are simply the variable name, fitted lambda-value, and the minimum and maximum values of the variable encountered during training the model.

The formula for the contribution of a raw feature with a value of, say x, to the output is: fx = lambda*(x - min)/(max - min)

(b) Quadratic features:

```
OZ_IsoTherm^2, -6.630126631087359, 1600.0, 8836.0
```

Quadratic features are recognised by the fact that the first part ends in '^2' representing 'raised to the power 2'. That is, the four parts are interpreted as: Variable name squared, lambda-value, min and max of the squared variable value found during training.

fx = lambda*(x*x - min)/(max - min)

(c) Product features:

```
OZ_AnnTempRange*OZ_PrecipSeason, 1.165382592371033, 360.0, 35712.0
```

Recognised by the fact that the first part represents a multiplication formula between two variable names. Min and max values are the min and max of the product of the two variables found during training.

fx1x2 = lambda*(x1*x2 - min)/(max - min)

(d) Forward hinge features:

```
'OZ_TempSeason, 0.21356227691610233, 25.5, 29.0
```

Recognised by the fact that the first part begins with the character ' followed by the variable name. Second part is the fitted lambda-value, and the min and max values represent the hinge-value and the maximum value encountered in training.

if x < hinge then fx = 0 otherwise fx = lambda*(x - hinge)/(max - hinge)

(e) Reverse hinge features:

```
`OZ_DryestMonPrecip, -0.2456443177330928, 0.0, 2.5
```

Recognised by the fact that the first part begins with the character ` followed by the variable name. Second part is the lambda-value, and the min and max values represent the minimum value encountered in training and the hinge-value.

if x < hinge then fx = lambda*(hinge - x)/(hinge - min) otherwise fx = 0

(f) Threshold features:

```
(263.5<OZ_AnnPrecip), 0.027969395068903, 0.0, 1.0
```

Recognised by the fact that the first part begins and ends in round brackets and reads like an inequality expression (which is exactly what it represents). The value to the left of the less-than sign is the threshold value. The second part is, as always, the fitted lambda-value, and the min and max values represent the outcome of applying the threshold test. That is, 0 if true and 1 if false.

if x < threshold then fx = 0 otherwise fx = lambda

Computing output values:

Computing the value associated with a given grid-cell requires the value of each predictor variable in that cell to be available so that each feature function (fx) can be evaluated. For readers not familiar with computer language short-hand, the exp() function represents "*e* raised to the power of the stuff in brackets". The steps involved in computing an output value are as follows:

- Calculate S = (Sum all the fx-values computed for a set of variables associated with a given grid cell) – linearPredictorNormalizer.
- Calculate qx = exp(S)/densityNormalizer. This is the raw model output.
- For the logistic-scaled output, compute (qx*exp(entropy)/(1 + qx*exp(entropy)). This is the default output from MaxEnt and the logit transformation re-scales the raw output in each grid-cell to fall between 0 and 1.

To compute a complete output map, a program or script needs to be written to systematically evaluate the model at each grid cell. In pseudo-code outline this would be something like:

```
for r = 1 to number of grid rows do
  for c = 1 to number of grid columns do
    read an array of predictor variables for grid-cell [r, c] assembled from each variable's grid
    compute the model for the array of predictor variables at grid-cell [r, c] as shown above
    store the output value for grid-cell [r, c] in an output grid
```

However, MaxEnt has a utility function that will do this for you when you supply a lambdas file and a set of predictor grids. This has been posted to the list by Steven Phillips. The command is:

```
java -mx1024m -cp maxent.jar density.Project "path to the lambdas file"  "path
to the environemntal predictors folder" "path to the output folder" standard
MaxEnt option flags
```

ooOoo