M.Sc. Geographical Information Science

School of Earth, Environmental, and Geographical Sciences

GIS, ENVIRONMENTAL CHANGE, AND THE LATE QUATERNARY DISTRIBUTION OF THE EURASIAN/AFRICAN WILDCAT (*FELIS SILVESTRIS*)

Dissertation project by Erin E. Rees

September 2002

Copyright

Statement of Originality

I declare that this dissertation represents my own work, and where the work of others has been used it has been duly accredited. I further declare that the length of the components if this dissertation is 5,443 words for the Research Paper and 10,160 for the Technical Report.

Erin E Rees September 11, 2002

ACKNOWLEDGMENTS

This research topic was proposed by Dr. Andrew J. Dugmore. I was immediately fascinated by it, and he figured it would be good to have a Canadian working on it since I should have no preconceptions about Eurasian and African wildcats. This certainly was the case because I pictured wildcats as looking like the large mountain lions and cougars we have back in Canada. As I began looking into this topic I could not understand how the wildcat species was under threat by being bred into extinction through interbreeding with domestic cats - maybe size doesn't matter? This topic definitely seemed worthy of a dissertation project.

Many thanks to Andy for a great topic, his enthusiasm, and for sharing his pool of knowledge about ecology, Quaternary environments, conservation issues, etc…

Dr. Andrew Kitchener from the Museum of Scotland was also extremely valuable to me because of his extensive knowledge about the wildcat species and their habitat preferences. Thank you.

Putting the technical parts of my project together would have seemed impossible without the help of key Geography department staff members: Chris Hill, Steve Dowers, Stewart Jamieson, Chris Place, and especially Dr. Nicholas R. J. Hulton. Thank you Nick, your help was invaluable to me, for getting me through modelling stumbling blocks and another session of JAVA programming.

Finally, I would like to thank my classmates with whom I could bounce around ideas with, get their help for resolving nick picky software issues, and could always count on for comic relief.

GIS, ENVIRONMENTAL CHANGE, AND THE LATE QUATERNARY DISTRIBUTION OF THE EURASIAN/AFRICAN WILDCAT (*FELIS SILVESTRIS*)

PART 1: Research Report

Erin E. Rees

September 2002

ABSTRACT

The benefits of using geographical information systems (GIS) to perform biogeographical analysis for clarifying taxonomic designations of species is demonstrated for the Eurasian and African wildcat (*Felis silvestris*). Wildcat taxonomy is a contentious issue because hybridisation with domestic cats (*Felis catus*) impedes the ability of traditional taxonomic techniques to define taxonomy. The genetic, morphological, or behavioural criteria used by these techniques have difficulty separating out the effects of hybridisation. The biogeographical approach used in this study overcomes this disadvantage because the data used to infer taxonomy (temperature, precipitation, and topography) does not show the effect of hybridisation with domestic cats. Temperature, precipitation, and topographical data are used to reconstruct the changing distribution of the wildcat over the Late Quaternary (past 20,000 years). The evolutionary history of wildcats is inferred from spatial analysis of the predicted distributions. The outcomes are assessed using known wildcat sightings data and by creating two additional biogeographical models that reconstruct wildcat distribution. The main biogeographical model has an 83% prediction success rate when compared to known wildcat sightings data. This model has a high level of correspondence with the outcomes of the secondary models for the modern day, but the level of correspondence is less when past reconstructions are compared. The results suggest areas where more morphologically and genetically distinct wildcat populations may exist and help explain why wildcats are not found in areas that have favourable wildcat habitat. The biogeographical approach is a promising technique for inferring biological taxonomic designations by supporting or disputing current taxonomic designations and by suggesting hypotheses for further morphological and genetic analysis. Additional benefits for biogeographical analysis in aiding conservation efforts are also discussed.

1 INTRODUCTION

1.1 Overall Aim

The overall aim of this study is to use a biogeographical approach to reconstruct the changing distribution of the wildcat over the Late Quaternary (last 20,000 years) in Eurasia and Africa. The Quaternary experienced substantial environmental change through a series of glacial cycles, with the last glacial maximum occurring at about 18k[1] and the last interglacial maximum occurring at approximately 9k (Williams *et al*, 1998). The Late Quaternary time period captures the minimum and maximum potential shifts in population range that could have occurred in response to climate change. This study's intent is to be able to assess the reconstructed distribution of wildcats relative to range extents, population density and fragmentation to improve understanding of the evolutionary development of wildcats and clarify the status of subspecies classifications

1.2 Uncertain Wildcat Taxonomy

The underlying motivation of this study is that the taxonomic status of wildcats is unclear. The wildcat distribution is massive compared to most mammals. Wildcats are found in western Europe, across southern Asia from the Gulf Region to western China, and occur throughout most of Africa. Given the extent of their distribution, it would seem likely that morphological and genetic differences exist between various populations that warrant subspecies classification. Over the years, numerous subspecies classifications have been assigned, however, these are often based on insufficient data (Nowell and Jackson, 1996). The major problem hampering proper subspecies classification is that hybridisation with domestic (and feral) cats, *Felis Catus*, is blurring the definition of what constitutes a wildcat. Wildcats and domestic cats interbreed wherever the two species are within close proximity of each other (Skinner and Smithers, 1990). Many studies have used morphological and genetic analysis to determine wildcat taxonomy (e.g. Balharry *et al*, 1997; Beaumont *et al*, 2001;

Daniels *et al*, 1998, 2001). A problem exists because these techniques have difficulty separating out the effect that hybridisation with domestic cats has on wildcat morphology and genetics. Since there is no known 'pure-bred' wildcat population that can be used as a base point for comparison with domestic cats, taxonomic conclusions based on morphological and genetic criteria are debatable (Balharry *et al*, 1997; Daniels *et al*, 1998; Daniels *et al*, 2001).

1.3 A Species at Risk

It is hoped this study will shed light on the taxonomic status of wildcats. This is important because proper taxonomic classification is crucial to conserving species, and wildcats are a threatened species that warrant conservation. Loss of habitat forces wildcats into more marginal areas where survival is much more difficult (Beaumont *et al*, 2001). Perhaps of greater concern is that habitat loss is pushing wildcats into closer proximity with human settlements. This is increasing the incidences of hybridisation with domestic cats, which leads to dilution of the wildcat gene pool. Unless habitats distant enough from domestic cats are conserved, wildcats are expected to be bred into extinction (Nowell and Jackson, 1996; Skinner and Smithers, 1990).

Proper taxonomic classification of wildcats will help their continued existence because they aid the development of appropriate conservation strategies. Misclassification of species can hamper conservation efforts by unknowingly keeping two genetically similar populations separate, or by mixing distinct populations and reducing biodiversity. Furthermore, conservationists might work to protect a seemingly rare population when it is common elsewhere but unrecognised as being the same species. Proper taxonomic classification is also necessary for being able to create and enforce legislation that affords legal protection for a species. Unfortunately there are too many examples of endangered populations failing to be protected by law because it is has been difficult to classify them under a taxonomic group that can be written into environmental legislation for legal protection (Freeman and Herron, 2001; O'Brien and Mayr, 1991; Pennock and Dimmick, 1997).

[1] k = 1000 years before present

1.4 Current Taxonomic Classification Techniques

Species are traditionally classified by their morphological, behavioural, and genetic characteristics, however, these techniques have their faults. Morphological criteria used to define wildcats include pelage patterns, body measurements, and gut lengths. However, morphometrics has been criticised as a method of grouping biological taxa because of the possibility of *phenotypic plasticity*, that is, the potential for an organism to develop any one of several phenotypes depending on their environment (Balharry *et al*, 1997). Defining species by behavioural traits, such as parenting strategies and social structure, maybe prone to interpretation and may not provide enough information to designate taxonomic status. Genetic techniques, such as immunological distances, electrophoresis, and DNA hybridisation, were initially hailed as the best and only means needed to determine taxonomy. Recent studies investigating wildcat taxonomy use genetics (Balharry *et al*, 1997; Beaumont *et al*, 2001; Daniels *et al*, 2001). However, the genetic structure of organisms and its meaning still needs to fully understood because varying genetic interpretations can result in different taxonomic conclusions. Balharry and colleagues (1997) argue that genetics should only be considered another tool from which hypotheses can be tested. However, using different taxonomic techniques together to define taxonomy can be complicated because the results do not always agree with one another (Freeman and Herron, 2001; Nowell and Jackson, 1996), necessitating decisions for why one technique should be supported over the others.

1.5 Biogeographical Analysis: a Novel Technique for Taxonomic Classification

This study uses a biogeographical approach to reconstruct the wildcat distribution over the Late Quaternary. This is done for the purpose of investigating whether significant shifts in ecological zones caused by global climate change fragment the wildcat population, and whether these patterns of fragmentation have any spatial similarity to the distribution of known modern day putative groups. Environmental variables (temperature, precipitation, and topography) define environmental regions (areas of unique climate and vegetation e.g. rainforest, desert) which are then ranked by current knowledge relative to their suitability as wildcat habitats. Mapping the ranked environmental regions creates the species distribution because it indicates the likelihood of wildcats occupying or avoiding the regions. The underlying theory is that environmental variables determine potential vegetation of a region, which in turn dictates the inhabiting prey, and that defines the likelihood of predator presence such as wildcats. Altering the environmental variables to simulate climate change reconstructs past distributions. Given that sufficient geographic and temporal isolation can lead to genetic differentiation (Freeman and Herron, 2001; Futuyma, 1998) the results of this study can shed light on subspecies classification by (1) hinting at areas where isolated or migratory wildcat populations may have developed and potentially diverged into new subspecies, and by (2) highlighting populations that should show little or no genetic/morphological divergence because they remained connected to the main population.

An important advantage of the proposed biogeographical approach over traditional taxonomic classification techniques is that the analysis is not based on morphological, genetic, or behavioural criteria. Because of this, the effect of hybridisation with domestic cats on clouding wildcat taxonomy can be overcome. The only factors affecting the results are the environmental variables used to drive the model. Furthermore, the wildcats' evolutionary history can be reconstructed even before domestic cats ever evolved[2]. Kitchener and Dugmore (2000) used a similar biogeographical approach to clarify inconsistent subspecies classifications of the tiger, *Panthera tigris*. Their results indicate that a biogeographical approach is a promising technique for resolving taxonomic classification issues.

[2] Domestic cats are thought to have evolved from Egypt about 4000 to 8000 years ago (Balharry *et al*, 1997; Nowell and Jackson, 1996). This study will reconstruct the wildcat distribution as far back as 18,000 years ago.

2 METHODOLOGY

This study uses three models to achieve its aims. The Rule Based Model (RBM) is the primary model for testing the biogeographical approach and for predicting the wildcat distributions. This model predicts wildcat distributions by using current knowledge to determine wildcat habitat preferences. The Biome Model (BM) and the Secondary Biome Model (SBM) are used to evaluate the RBM. These models predict wildcat distributions by using known wildcat sightings data to determine wildcat habitat preferences.

2.1 Model 1: The Rule Based Model

Current knowledge of wildcat habitats is used to define 32 environmental regions that represent areas that wildcats seek to occupy or avoid (Corbet, 1978; Corbet and Hill, 1992; Happold, 1987; Harrison and Bates, 1991; Kitchener, 2002; Mitchell-Jones *et al*, 1999; Nowell and Jackson, 1996; Roberts, 1997). The Koppen Climate System (KCS) climate types form the majority of the different environmental regions. The KCS is a useful classification scheme because its climate types closely follow the boundaries of the major vegetation biomes, and this can easily be equated to wildcat habitats. Map algebra is used to create grids of each environmental region, such that, the environmental region grid cells are derived as a function of the geographically corresponding environmental variable grid cells (Table 1a, b). The environmental variable grid data consist of 50 km resolution rasters for: mean annual surface temperature (ctmp), mean January surface temperature (tjan), mean July surface temperature (tjul), mean annual surface precipitation (cpre), mean January surface precipitation (pjan), mean July surface precipitation (pjul), topography (dem), temperature range (trange), and continentality (cont). The 32 environmental regions were ranked relative to wildcat habitat preference. Merging the ranked environmental region grids into one grid produced the modern day wildcat species distribution where the grid cells of the study area are classified as either being absent (rank 0), low (rank 1), medium (rank 2), or high (rank 3) for wildcat presence. Subjectively ranking the environmental regions affects the outcome of the distribution. However, because this stage is transparent, it is easy to evaluate the effectiveness of the rankings in recreating the known distribution by altering the rankings until the most realistic distribution is produced.

Table 1a. The environmental variable specifications of the environmental regions based on the Koppen Climate System climate types. The 'latitude' grid is used to separate northern hemisphere requirements from southern hemisphere requirements. Environmental variable grids used are: cpre = mean annual precipitation, ctmp = mean annual surface temperature, cont = continentality, dem = topography, pjan = mean January precipitation, pjul = mean July precipitation, tjan = mean January temperature, tjul = mean July temperature, trange = temperature range.

Environmental Region	Environmental Variable Specifications
Af (tropical rainforest)	ctmp >= 180 & pjan >= 18 & pjul >= 18 & trange <= 7 & pjul > 400 & pjan > 500 & cpre > 1000
Am (monsoon)	ctmp > 180 & cpre < 5840 & pjan < 500 & pjul > 9000
Aw (tropical savanna)	ctmp >= 180 & tjan >= 18 & tjul >= 18 & trange < 10 & pjan > (1.5 * pjul) or pjul > (1.5 * pjan)
BSh (dry-hot steppe)	(latitude >= 0 & tjul >= 22 & tjul < 35 & tjan >= 3 & ctmp > 35 & pjan < 6000 & pjul < 4500 & (pjul + pjan) > 450 & cpre < 5000) ~ or (latitude < 0 & tjul >= 7 & tjan >= 22 & pjul < 1500 & pjan < 4500 & trange <= 17 & (pjul + pjan) > 450)
BSk (dry-cold steppe)	(latitude >= 0 & tjan <= 1 & tjul > 10 & tjul < 24 & pjan <= 2000 & pjul <= 2000) or (latitude < 0 & tjul <= 1 & tjan > 10 & tjan < 24 & pjul <= 2000 & pjul <= 2000)
BWh (dry-hot desert)	tjan > 0 & tjul > 0 & pjan < 1500 & pjul < 1500 & cpre < 4000
BWk (dry-cold desert)	(latitude >= 0 & tjul >= 18 & ctmp > -150 & tjan < 5 & tjan > -20 & cpre < 4000 & pjul < 1000 & pjan < 1000) ~ or (latitude < 0 & tjan >= 18 & ctmp > -150 & tjul < 5 & tjul > -20 & cpre < 4000 & pjan < 1000 & pjan < 1000)
Cfa (mild, humid, no dry season, hot summers)	(latitude >= 0 & tjan >= -3 & tjul >= 22 & trange >= 18 & trange < 30 & cpre > 1000 & cpre < 13000 & pjan > 400 & pjul > 1000) ~ or (latitude <= 0 & tjul >= -3 & tjan >= 22 & trange >= 5 & cpre > 1000 & cpre < 17000 & pjul > 600 & pjan > 1800 & pjan > (1.5 * pjul))
Cfb (mild, humid, no dry season, warm summers)	(latitude >= 0 & tjan >= -3 & tjan < 18 & tjul > 10 & tjul <= 22 & pjan >= 60 & pjul >= 60) ~ or (latitude < 0 & tjul >= -3 & tjul < 18 & tjan > 10 & tjan <= 22 & pjul >= 60 & pjan >= 60)
Cfc (mild, humid, no dry season, short summer)	(latitude >= 0 & tjan >= -3 & tjan < 10 & tjul < 10 & pjan >= 60 & pjul >= 60 & dem < 1000) ~ or (latitude < 0 & tjul >= -3 & tjul < 10 & tjan < 15 & pjul >= 60 & pjan >= 60 & dem < 1000)
Cwa (mild, humid, dry winters, hot summers)	(latitude >= 0 & tjan >= -3 & tjul > 15 & ctmp < 230 & trange > 3 & pjan < 1000 & pjul > 1000 & cpre > 300) ~ or (latitude < 0 & tjul >= -3 & tjul <= 19 & tjan > 15 & trange > 2 & trange < 14 & pjan > 3500 & pjul < 500)
Csa (mild, humid, dry summers, hot summers)	(latitude >= 0 & tjan > -3 & tjan < 18 & tjul > 18 & pjul < 500 & pjan > 1000) or (latitude >= 0 & tjul > -3 & tjul < 18 & tjan > 18 & pjan < 500 & pjul > 1000)
Csb (mild, humid, dry summers, warm summers)	(latitude >= 0 & tjan > 0 & tjan <= 10 & tjul <= 23 & pjan <= 10000 & pjul < 1500 & dem < 1000 & cont < 1 & trange < 20) or (latitude < 0 & tjul > 0 & tjul <= 10 & tjan <= 23 & pjul <= 10000 & pjan < 1500 & dem < 1000 & cont < 1 & trange < 20)
Dfa (snowy-forest, no dry season, hot summer)	(latitude >= 0 & tjan >= -5 & tjan < 0 & tjul >= 22 & pjan >= 2000 & pjul >= 1900 & trange < 30) or (latitude < 0 & tjul >= -5 & tjul < 0 & tjan >= 22 & pjul >= 2000 & pjan >= 1900 & trange < 30)
Dfbc (snow-forest, no dry season, warm summer/ cool short summer)	(latitude >= 0 & tjan > -38 & tjan < -3 & tjul < 22 & pjul > 1500 & pjan > 500) or (latitude < 0 & tjul > -38 & tjul < -3 & tjan < 22 & pjan > 1500 & pjul > 500)
Dfd (snow-forest, no dry season, cold winter)	(latitude >= 0 & tjan <= -38 & tjul > 10 & (4 * pjan) >= pjul)
Dwa (snowy-forest, dry winter, hot summer)	(latitude >= 0 & pjan < 500 & pjul > 2000 & tjan < -3 & tjul > 22) or (latitude < 0 & pjul < 500 & pjan > 2000 & tjul < -3 & tjan > 22)
Dwbc (snow-forest, dry winter, cool short summer)	(latitude >= 0 & tjul > 10 & tjul < 22 & tjan < -3 & (4 * pjan) < pjul) or (latitude < 0 & tjan > 10 & tjan < 22 & tjul < -3 & (4 * pjul) < pjan)
Dwd (snow-forest, dry winter, warm summer)	(latitude >= 0 & (4 * pjan) < pjul & tjan <= -38) or (latitude < 0 & (4 * pjul) < pjan & tjul <= -38)
ET (polar, tundra)	(latitude >= 0 & tjul > 0 & tjul < 10) or (latitude < 0 & tjan > 0 & tjan < 10)
EF (polar, perpetual forest)	tjan < 0 & tjul < 0 & ctmp < 0

The RBM version that produced the best present day outcome was then used to reconstruct the wildcat distribution for 9k, 13k, and 18k. These time periods represent the last interglacial (9k) and glacial (18k) maxima and the approximate midpoint (13k) between the two. Climate change was simulated between the three time periods by modifying the continentality and temperature environmental variables. The RBM was written using Arc Macro Language (AML) and run in ArcInfo 8.0.2 (see Rees (2002) for details).

Table 1b. The environmental variable specifications of the environmental regions not based on the Koppen Climate System climate types. Continental grids 'europe', 'asia', and 'africa' are used to keep the specifications restricted to the continent. Environmental variable grids used are: cpre = mean annual precipitation, dem = topography, pjan = mean January precipitation, pjul = mean July precipitation, tjan = mean January temperature, tjul = mean July temperature. Highland environmental regions are denoted by 'h'.

Environmental Region	Environmental Variable Specifications
african precipitation	africa = 1 & cpre < 365 & pjan < 100 & pjul < 1000
snow	tjan < -5 & tjul <= 20
h0a	europe == 1 & dem > 3000
h0b	africa == 1 & dem > 3500
h0c	asia == 1 & dem > 3000
h2a	africa == 1 & dem < 2000
h2b	asia == 1 & dem < 2000
h3a	africa == 1 & dem >= 2000 & dem < 2500
h3b	asia == 1 & dem >= 2000 & dem < 2500
h4a	africa == 1 & dem >= 2500 & dem < 3500
h4b	asia == 1 & dem >= 2500 & dem < 3000

2.2 <u>Investigating Changes to the Late Quaternary Distribution as Modelled by the Rule Based Model</u>

The transition of the distribution over time is investigated by visually comparing the 18k and 13k, the 13k and 9k, and the 9k and present day outcomes. Furthermore, the percent coverage of the entire area of suitable habitat (i.e. wildcat habitat preference 1, 2, and 3) is calculated for the present day, 9k, 13k, and 18k distributions to determine how much the area of the entire distribution has altered over time. To investigate *where* the distribution has changed through time, the percent coverage of each wildcat preference rank is calculated for Europe, Africa, and Asia for the present day, 9k, 13k, and 18k outcomes. A difference grid is also calculated by subtracting the 18k outcome from the 9k outcome to identify areas were wildcat habitat preference differed between the last glacial and interglacial maxima.

2.3 <u>Assessing the Rule Based Model</u>

Two approaches were used to assess the reconstructed wildcat distributions produced by the RBM.

(1) The modern day RBM outcome was compared with unrelated known wildcat sightings data. The level of correspondence between the two datasets was compared visually by mapping the sightings data over the RBM outcome. Furthermore, the GIS analytical *intersect*

operation was used to calculate the percentage of grid cells for each class of wildcat preference that geographically coincided with areas of known wildcat presence.

(2) Two additional models that can predict the wildcat distribution were created to assess the RBM. These models are constructed in a similar manner to the RBM, in that, environmental data is ranked relative to wildcat habitat preference and then mapped to reconstruct the distribution. However, the rankings are not assigned by current knowledge but by the statistical correlation between known wildcat sightings data and unrelated environmental data[3]. For instance, if wildcats are rarely found in the tundra habitat the resulting wildcat habitat preference for this region as derived by statistical analysis will be extremely low.

2.3.1 Model 2: The Biome Model

The BM is created by intersecting the known wildcat sightings data with present day BIOME 6000 palaeo-data of biome types to identify the biome types wildcats prefer and avoid. Confusion matrices are produced from the intersections and Kappa statistical analysis is performed on this information to determine which biomes are meaningful predictors of wildcat presence and absence. The results of the Kappa analysis are then used to rank the biomes relative to wildcat preference following the same four-point ranking system used in the RBM (Table 2). The biome rankings are assigned to the present day, 9k, and 18k BIOME 6000 biome data to reconstruct the wildcat distribution for these time periods (see Rees (2002) for details).

2.3.2 Model 3: The Secondary Biome Model

The wildcat distribution constructed by the BM does not cover the entire study area because distribution data is only present where data points from the BIOME 6000 datasets exist. The SBM was written in JAVA and run in Kawa 4.10a to derive a modern day wildcat distribution

[3] Some of the environmental data used in one of the models is the same as that used in the RBM. Implications of this are discussed by Rees (2002).

for every grid cell in the study area. The aim was to produce an outcome that would be a more meaningful result with which to evaluate the modern day RBM outcome because every grid cell value in the RBM could be assessed. The SBM uses a nearest neighbour approach to define biome types for each grid cell as a function of the environmental variable values found at that location. The environmental variables used are: mean annual precipitation, mean annual temperature, and temperature range (see Rees (2002) for details). Once biome types are derived they are then ranked relative to wildcat preference in the same manner described for the BM (Table 3). Mapping the ranked habitats reconstructs the present day wildcat distribution.

2.3.3 Assessing the Rule Based Model with the Biome Model and Secondary Biome Model

The present day, 9k, and 18k RBM wildcat distribution reconstructions are compared with corresponding time period outcomes from the BM and SBM outputs. Since all models use a four point wildcat preference ranking system, subtracting the BM or SBM outcome from the RBM outcome can identify the level of correspondence between the reconstructions. Negative values are cells where the RBM has a lower preference. Positive values are cells where the RBM has a higher preference. A value of zero occurs when the wildcat preference between the RBM and BM or SBM are identical.

Table 2. Wildcat preference, probability of wildcat presence, and Kappa statistic of each biome used in the creation of the Biome Model. Probability of wildcat presence is calculated as the division between the number of times sightings data coincides with the biome type and the total number of times that biome type occurs in the dataset. Probability of presence gives additional information as to the importance of each biome type for predicting wildcat presence. Biomes with higher Kappa values are more meaningful predictors of wildcat presence. * Biome types with fewer than 25 records were not used in creating the Biome Model because they do not yield meaniful preference, presence and Kappa results.

Biome Type	Wildcat Preference	Probability of Wildcat Presence	Kappa Value
xerophytic woods/scrub	3	0.77	0.160
savanna	3	0.88	0.155
steppe	3	0.38	0.102
warm mixed forest	3	0.54	0.079
desert	2	0.33	0.027
broadleaved evergreen/warm mixed forest	2	0.34	0.018
tropical dry forest/savannah *	2	0.78	0.018
temperate xerophytic woods/shrubs *	2	1.00	0.017
cold mixed forest	2	0.40	0.012
temperate microphyllous shrubland *	2	0.80	0.010
tropical rain forest *	1	0.67	0.005
warm grass/shrub *	1	1.00	0.003
tropical seasonal forest	1	0.30	0.002
tropical xerophytic bush/savannah *	1	0.50	0.002
hot desert *	0	0.13	-0.005
tundra	0	0.07	-0.051
cool conifer forest	0	0.07	-0.058
cold deciduous forest	0	0.02	-0.064
temperate deciduous forest	0	0.17	-0.080
cool mixed forest	0	0.13	-0.147
taiga	0	0.02	-0.174

Table 3. Wildcat preference, probability of wildcat presence, and Kappa statistic of each biome used in the creation of the Secondary Biome Model. Probability of wildcat presence is calculated as the division between the number of times sightings data coincides with the biome type and the total number of times that biome type occurs in the dataset. Probability of presence gives additional information as to the importance of each biome type for predicting wildcat presence. Biomes with higher Kappa values are more meaningful predictors of wildcat presence.

Biome Type	Wildcat Preference	Probability of Wildcat Presence	Kappa Value
savanna	3	0.75	0.1735
tropical xerophytic bush/savannah	3	0.74	0.0888
tropical dry forest/savannah	3	0.62	0.0726
temperate xerophytic woods/shrubs	3	0.79	0.0624
temperate microphyllous shrubland	3	0.59	0.0553
cold mixed forest	3	0.48	0.0481
warm grass/shrub	2	0.54	0.0168
tropical seasonal forest	2	0.46	0.0103
temperate deciduous forest	1	0.35	0.0032
warm mixed forest	1	0.48	0.0026
xerophytic woods/scrub	1	0.39	0.0021
cool mixed forest	0	0.32	-0.0003
tropical rain forest	0	0.29	-0.0023
hot desert	0	0.16	-0.0109
desert	0	0.29	-0.0111
broadleaved evergreen/warm mixed forest	0	0.25	-0.0185
cool conifer forest	0	0.08	-0.0357
taiga	0	0.01	-0.0442
steppe	0	0.28	-0.0583
cold deciduous forest	0	0.00	-0.1514
tundra	0	0.07	-0.1717

3 RESULTS

3.1 The Rule Based Model

The wildcat habitat preference rankings used to create the optimal RBM are based on a combination of sources of current wildcat knowledge (Nowell and Jackson, 1996; Kitchener, 2002; Skinner and Smithers, 1990); (Table 4). The modern day distribution produced by the RBM covers a much larger area than the known wildcat sightings data (Figure 1). Areas of interest numerically flagged in Figure 1 are:

- The distribution closely follows the known leopard cat (*Prionailurus bengalensis*) distribution on mainland Asia (1).
- The distribution corresponds well with areas of known wildcat absence for the Tibetan Plateau (2a), the central African rainforests (2b), and the South East Asian Archipelago (2c).
- The distribution is exaggerated relative to the known sightings data north of the Caspian Sea (3a), Eastern Europe (3b), the southern and northern fringes of the Sahara Desert (3c, 3d), and the Arabian Penisula (3e).
- Wildcat presence is predicted in areas in which they have never been known to have existed - Ireland (4a), Western Madagascar (4b), Japan (4c), and Australia (4d).

The present day, 9k, 13k, and 18k outcomes from the RBM wildcat distribution show how wildcat distribution may have changed over time (Figure 2). The modelled habitat preferences remain relatively unchanged in Africa, whereas, a high degree of change is suggested to have occurred in Europe and Asia. Areas of interest numerically flagged in Figure 2 are:

- Possible splits in the 18k wildcat distribution occur between the European and Asian populations (1a), between the northern and central African populations (1b), and between the central and southern African populations (1c).
- Following the interglacial maximum, the 13k distribution shows a range expansion of wildcats into northern China (2a) and into Eastern Europe (2b).
- Throughout the Late Quaternary two major wildcat refugia appear to occur on the Iberian Peninsula (R1) and in the Gulf Region (R2).

- The refugia R1 and R2 are directly isolated from each other during the 18k reconstruction (1a). However, both are connected to Africa populations, for most of the Late Quaternary for R1 (3a), and for all of the Late Quaternary for R2 (3b).

Table 4. A comparison between the wildcat preference rankings for the environmental regions used in the Rule Based Model as derived by Kitchener (2002), (K), Nowell & Jackson (1996), (NJ), and Skinner and Smithers (1990), (SS), and the values that finally defined the optimal model (RBM).

Environment and its Description			**Habitat Preference**		
Koppen Climates:			**K**	**NJ**	**RBM**
A tropical rainy climates	**Af tropical rainforest**		0	0	0
	Am seasonal monsoon		1	0	1
	Aw tropical savannah		3	3	3
B dry climates	**BS steppe climate**	**BSh dry-hot**	2	3	3
		BSk dry-cold	2	1	2
	BW desert climate	**BWh dry-hot**	1	2	2
		BWk dry-cold	1	1	2
C warm temperature climates	**Cf no dry season**	**Cfa hot summer**	2	3	2
		Cfb warm summer	2	3	3
		Cfc cool, short summer	2	3	2
	Cw dry winter	**Cwa hot summer**	2	2	2
	Cs dry summer	**Csa hot summer**	2	1	2
		Csb warm summer	2	1	2
D snowy climates	**Df moist winter**	**Dfa hot summer**	1	1	1
		Dfbc warm/cool summer	1	0	1
		Dfd cold winter	0	0	0
	Dw dry winter	**Dwa hot summer**	1	1	1
		Dwbc warm/cool summer	1	0	1
		Dwd cold winter	0	0	0
E polar climates	**ET tundra**		0	0	0
	EF perpetual frost		0	0	0
Non Koppen Climate Environments:					
Europe	elevation > 3000 m		0	0	0
Africa	elevation > 3500		0	0	0
Africa	elevation < 2000		3	3	3
Africa	elevation >= 2000 and elevation < 2500		-	2	2
Africa	elevation >= 2500 and elevation < 3500		-	1	1
Asia	elevation > 3000		0	0	0
Asia	elevation < 2000		3	3	3
Asia	elevation >= 2000 and elevation < 2500		-	2	2
Asia	elevation >= 2500 and elevation < 3000		-	1	1
Snow	snow cover > 20 cm and lasting > 2 consecutive weeks a year		0	0	0
			SS		**RBM**
African precipitation	< 100 mm per year		0		0

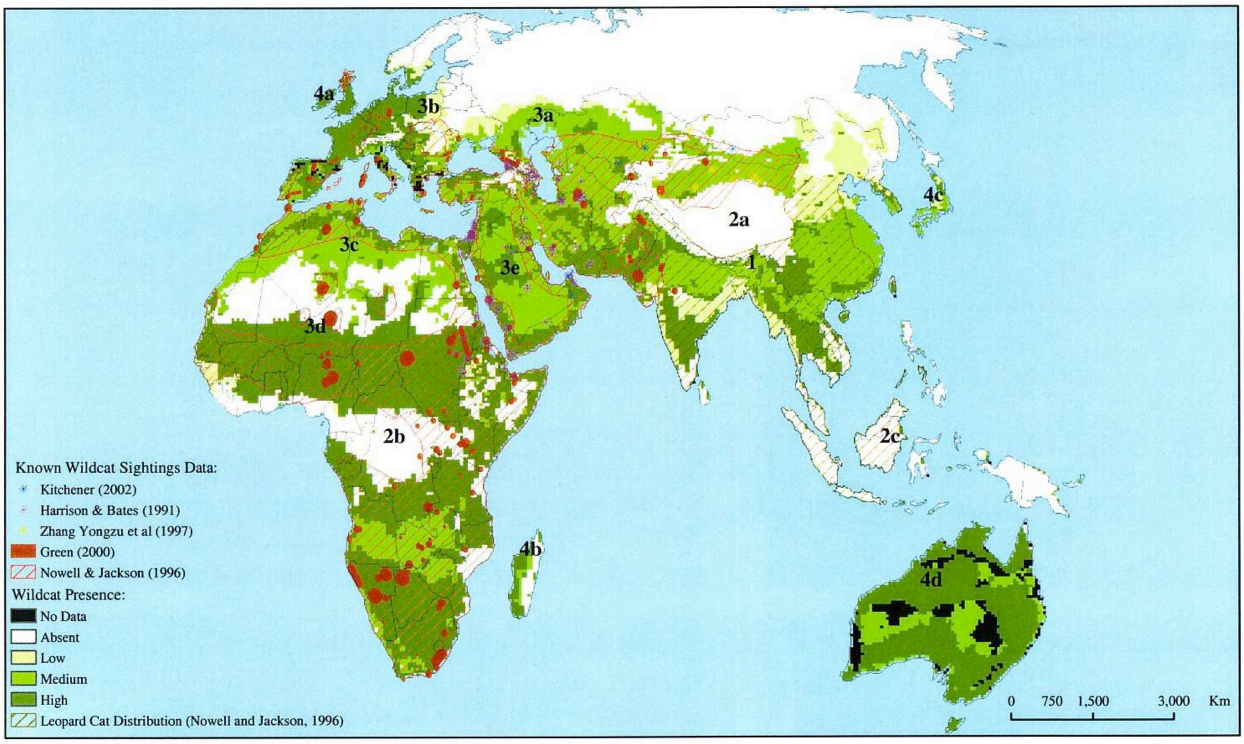


Figure 1. Modern-day RBM distribution.

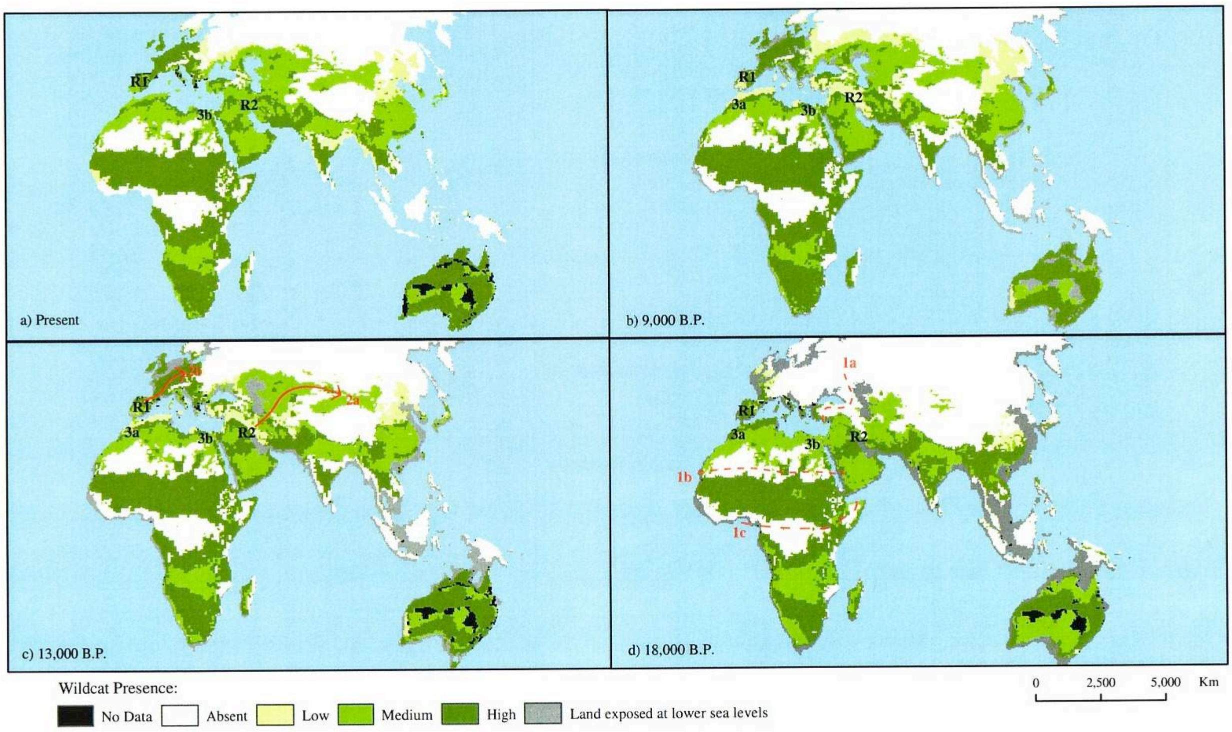


Figure 2. Modern-day and paleo RBM distributions.

The total area covered by the distribution during the Late Quaternary follows a similar trend to the increases and decreases in global temperature, such that, the area is larger for warmer temperatures and smaller for cooler temperatures (Figure 3). The most expansive distribution occurs during the warmest period of the Late Quaternary, at about 9k, and the smallest distribution occurs during the coolest period of the Late Quaternary at approximately 18k (Figure 4).

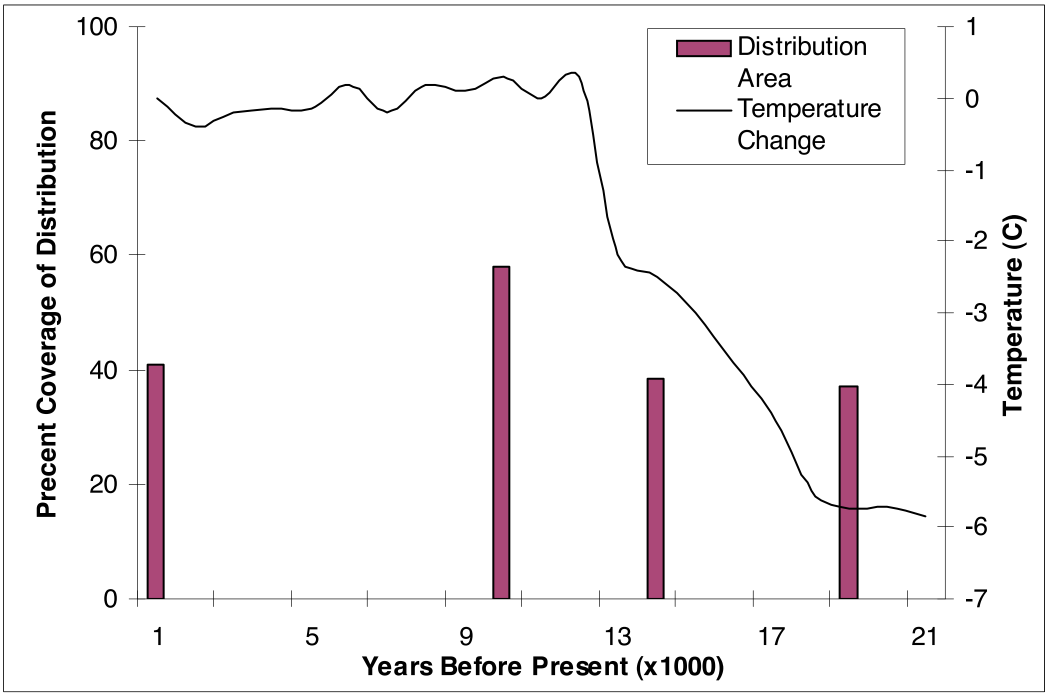


Figure 3. Comparing the global temperature over the last 20,000 years to the total area of preferred wildcat habitat for Eurasia and Africa, as predicted by the Rule Based Model. Total area of habitat predicted for present day is 41%, for 9k is 58%, for 13k is 38%, and for 18k is 37%. The present day distribution is only slightly larger than the 18k distribution, despite a large difference in global temperature.

The effect of Late Quaternary global climate change on the wildcat habitat preferences is predicted to differ between Africa, Asia, and Europe (Figure 5). Africa shows little change in the area of preferred habitat during the Late Quaternary. Asia is also relatively stable, however, there is a slight increase in area of low and medium preferred habitats and a slight decrease in area of avoided habitats at warmer temperatures. In contrast to Africa and Asia, Europe is predicted to display a high degree of change in the area of preferred habitat over the Late Quaternary. Most notable is a large increase in the area of the highest habitat preference, mirrored inversely by a large decrease in the area of avoided habitat in response to increasing global temperatures.

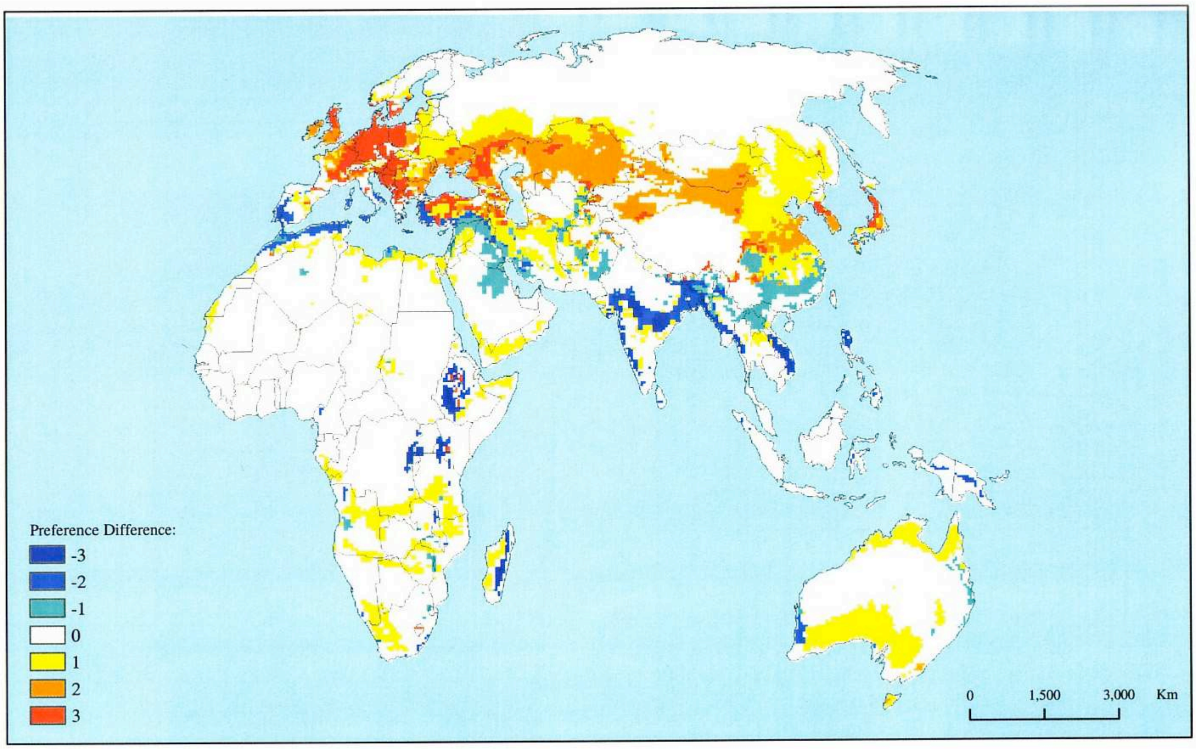


Figure 4. Difference in wildcat habitat preference between the RBM 9,000 b.p. and 18,000 b.p. distributions.

(a) Europe

(b) Africa

(c) Asia

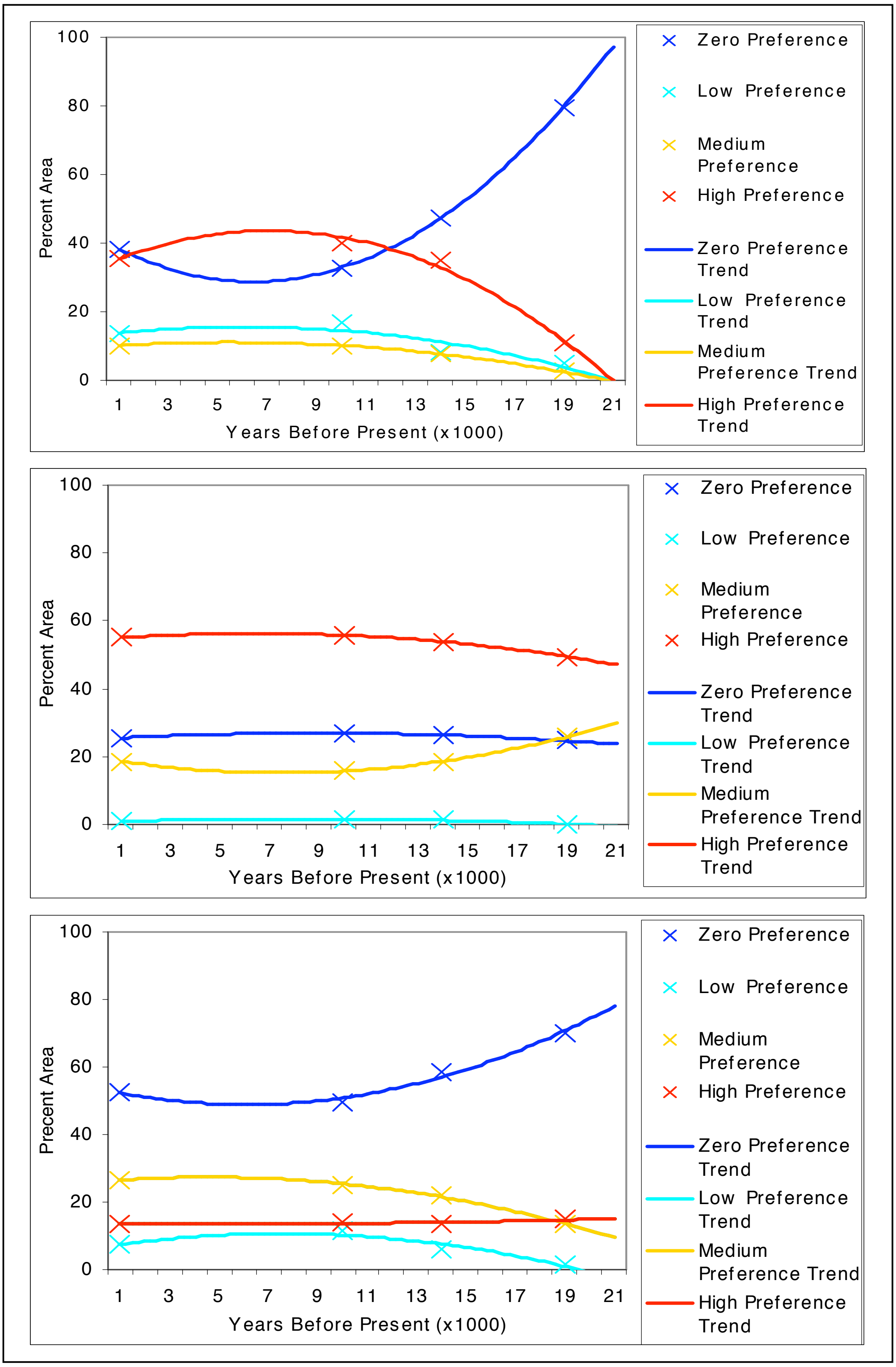


Figure 5. Estimated percent land cover of the four measures of wildcat habitat preference for (a) Europe, (b) Africa, and (c) Asia, over the last 20,000 years.

3.2 Prediction Success

To quantify a level of confidence in the results produced by the RBM, the outcomes were compared to the known wildcat sightings data and to the outcomes produced by the HB and the SHB.

3.2.1 The Rule Based Model versus the known wildcat distribution

When compared to five sources of known wildcat sightings data, the present day RBM outcome shows an overall agreement of 83% for grid cell values predicting wildcat presence in areas of known wildcat presence (Table 5). The highest level of correspondence was with the Nowell and Jackson (1996) data, while the lowest level was with data from Zhang Yongzu *et al* (1997). The highest wildcat preference type (value 3) was the most commonly occurring preference type coinciding with areas of known presence.

Table 5. The percentage of grid cells in the Rule Based Model present day outcome that correspond with known wildcat presence as calculated for each habitat preference ranking: avoid (0), low (1), medium (2), high (3), unknown (no data).

Wildcat Sightings Data:	Habitat Preference Correspondence:					Sightings Data Correspondence:
	No Data	0	1	2	3	
Nowell and Jackson (1996)	0.04	11	1.7	30	57	89
Green (2000)	0.2	13	1.3	25	60	87
Kitchener (2002)	0.0	14	0.0	14	71	86
Zhang Yongzu *et al* (1997)	0.0	33	0.0	67	0.0	67
Harrison and Bates (1991)	7.0	3.5	3.5	41	45	89
				Overall Correspondence:		**83**

3.2.2 The Rule Based Model versus the Biome Model and the Secondary Biome Model

The wildcat distributions created by the RBM were compared to the distributions created by the BM (Figure 6) and the SBM (Figure 7). If the models agree *perfectly* there would be no difference in wildcat preference for all grid cells compared. If the models agree for *most* values, the a cell by cell comparison for habitat preferences would show a normal distribution, that is, the majority of grid cells would show no difference in preference, and as the difference in preference increases the number of incidences of this occurring decreases.

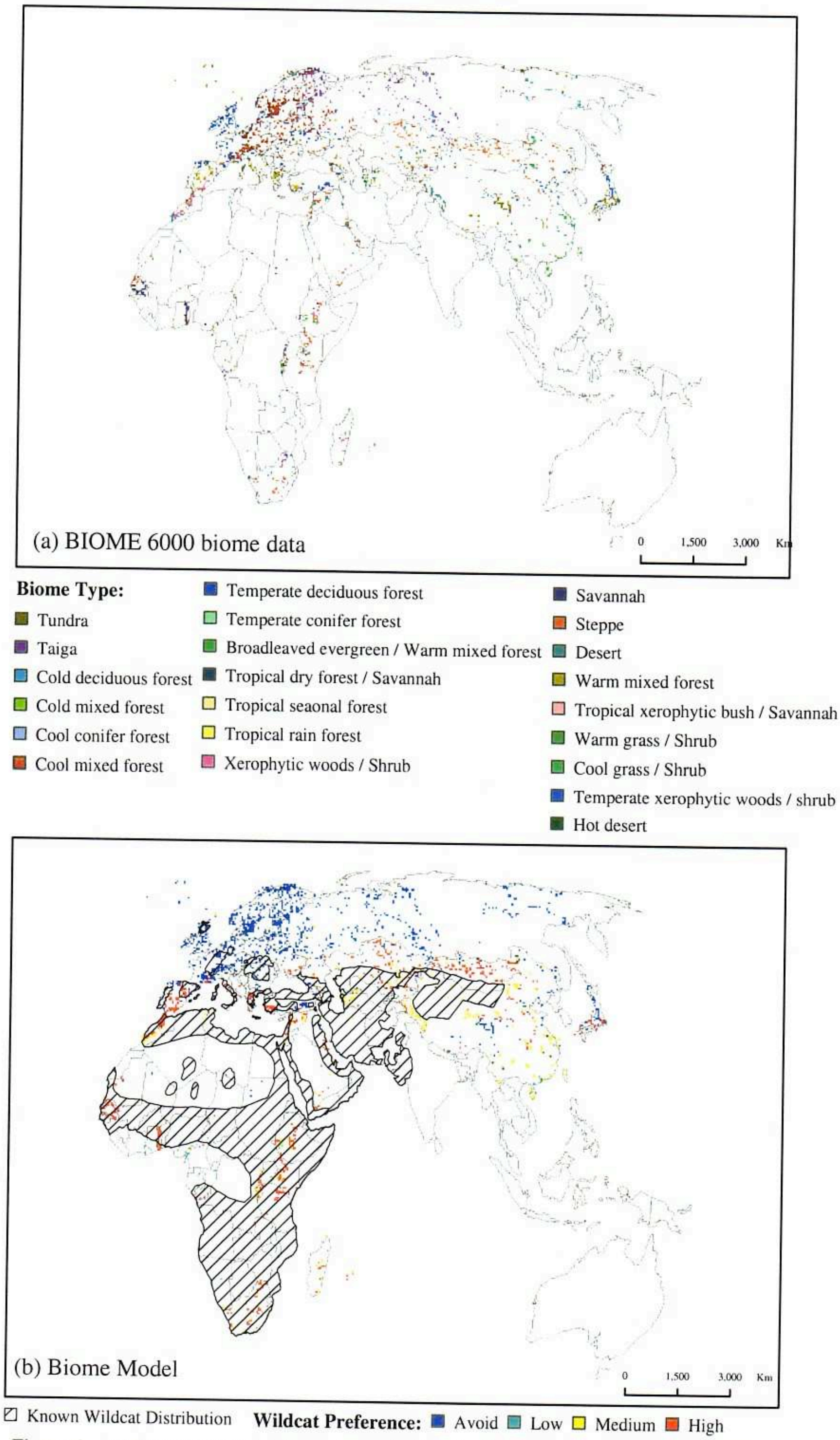


Figure 6. (a) BIOME 6000 data biome types for 0k. (b) The 0k Biome Model outcome that is produced by ranking the BIOME 6000 data with respect to wildcat preferences avoid, low, medium, and high, and the known wildcat distribution as derived from Nowell and Jackson (1996).

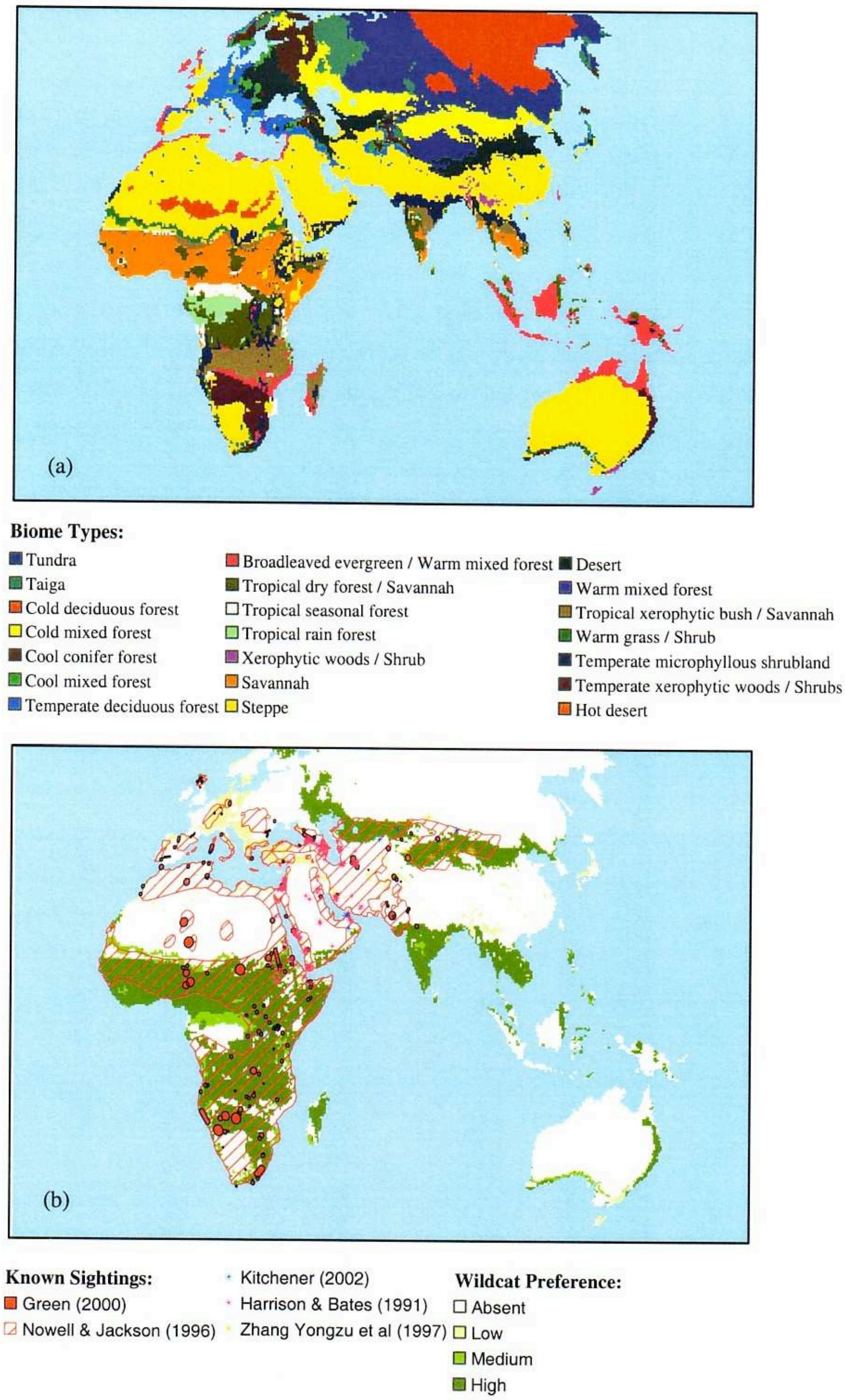
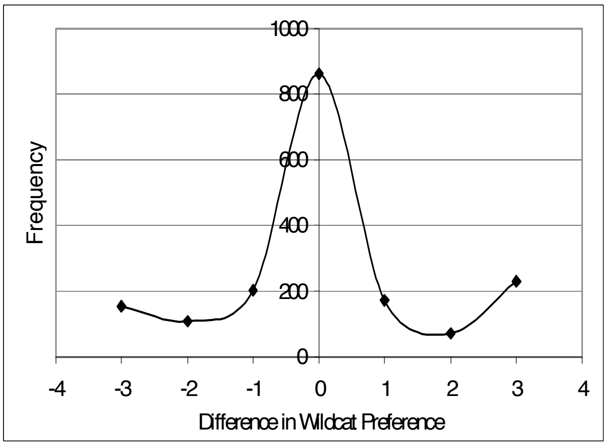


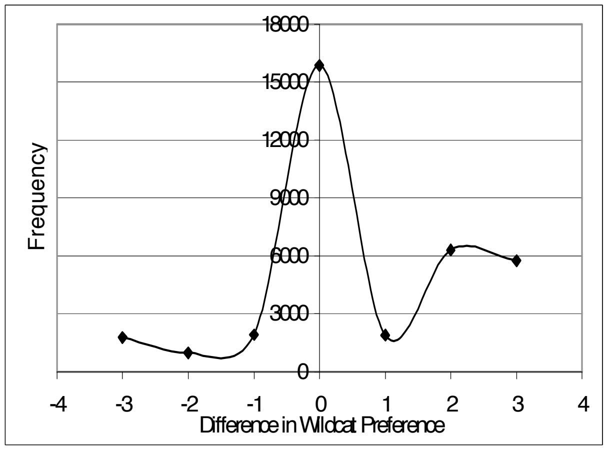
Figure 7. (a) Biome types predicted by the Secondary Biome Model. (b) Wildcat distribution predicted by the Secondary Biome Model for the present day and overlaid with known wildcat sightings data.

The level of correspondence between the RBM and the BM is highest for the modern day outcome and lowest for the 18k outcome (Figure 8). For the present day, 9k and 18k comparisons, the present day evaluation follows a normal distribution most closely. The 9k comparison is slightly bell shaped and is more skewed to the right. This is happening because (1) many grid cells in the RBM are one degree higher in wildcat preference rating than the grid cells assigned by the BM in South Africa and the Rift Valley, and (2) a significant proportion of grid cells in the RBM have lower wildcat preference rankings in central Africa. In this region the RBM assigns zero wildcat preference because *Af tropical rainforest* is the predicted habitat, whereas, the BM assigns higher wildcat preference rankings because *temperate microphyllous shrubland* and *temperate xerophytic woods/shrubs* are the predicted habitats. The 18k comparison would resemble a normal distribution except there are a high number of cells in central Asia where the RBM has assigned a lower habitat preference ranking than the BM. In this region, the RBM assigns a habitat preference of zero because of high snow cover, whereas, the BM assigns habitat preference of three and two because *steppe* and *desert* biomes are found there (see Tables 2, 3, 4 for habitat preference rankings).

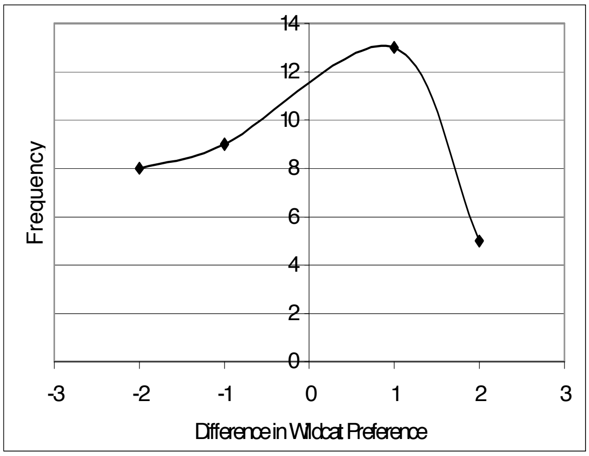
The level of correspondence between RBM and the SBM present day outcomes is good through central and southern Africa and central Asia (Figure 7), as is exemplified by the bell shaped curve illustrating the difference in preference values (Figure 8b). The high frequency of cells where there are habitat preference differences occur because the RBM overpredicts habitat preference in the Gulf Region, Europe, and south-western Asia relative to the SBM.



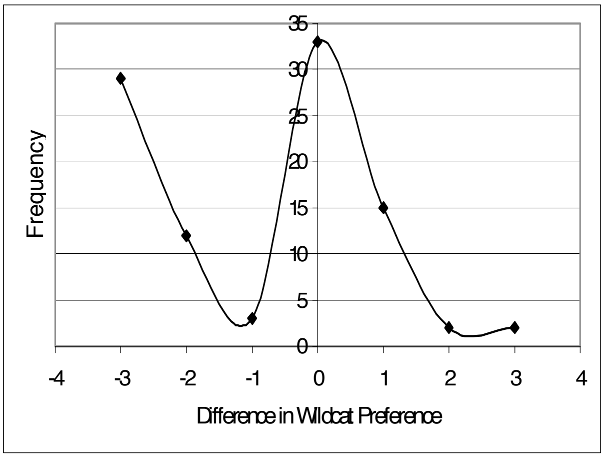
(a) BM 0k Distribution



(b) SBM 0k Distribution



(c) BM 9k Distribution



(d) BM 18k Distribution

Figure 8. Comparing the difference in frequency of wildcat preference between the grid cells of the Rule Based Model (RBM) and the Biome Model (BM) outcomes for (a) 0k, (c) 9k, (d) 18k, and for the RBM and the Secondary Biome Model (SBM) outcomes for (b) 0k. Negative values occur where the RBM cells have a lower wildcat preference, positive values occur where the RBM cells have a higher wildcat preference, and values of zero occur where there is no difference in preference between the models. There are no incidences of cells being three values apart between the 9k RBM and 9k BM comparison because the 9k BM only has biomes that were ranked with wildcat preference values of 1 and 2.

4 DISCUSSION

The wildcat evolutionary history recreated by the biogeographical approach implies the distribution moved in response to Late Quaternary climate change. The response is consistent with major expanses of ice sheets and permafrost (Williams *et al*, 1998). The results suggest a number of interesting implications for ecology and for performing biogeographical analysis.

4.1 Ecological Implications

4.1.1 Subspecies classification

Molecular analysis suggests that African and European wildcats diverged about 20,000 years ago (Nowell and Jackson, 1996). The RBM outcomes support this claim. The modelled split in the distribution along the Carpathian mountains in the 18k reconstruction and the dubious land connection across the Straight of Gibraltar may have isolated European wildcats from Asian and African wildcats long enough for morphological and genetic differentiation to occur (Figure 2d: split 1a, connections 3a).

Nowell and Jackson (1996) hypothesise Asian and African wildcats are conspecific and that European wildcats are a separate species. Once again the RBM outcomes support this hypothesis because the Asian/Africa connection is modelled to be stronger than the European/Asian connection. There is clearly a lot of land along the Nile Delta for Asian and African wildcats to travel between continents, affording the opportunity to exchange genes. However, the land connection over the Straight of Gibraltar is tenuous and would have eventually become a barrier as sea levels rose (Figure 2d: split 1a, connections 3a, 3b).

The RBM outcomes also suggest areas where subspeciation may have occurred that were not found discussed in the wildcat literature. Chinese wildcats are predicted to show the highest degree of morphological and genetic differentiation. As the earth warmed from the glacial maximum, wildcats could have migrated several 1000's of kilometres from their modelled refuge in the Gulf Region, north-easterly around the Tibetan Plateau, and into northern China (Figure 2c: route 2a). It is doubtful they expanded into northern China along the southern side of the Tibetan Plateau because the presence of the leopard cats in that region likely kept them out[4]. Such a long migration over a relatively short period of time (from 18k to 13k) should have thinned out the wildcat population. The potential for morphological and genetic divergence may have existed because small populations are more apt to express recessive traits and preserve mutations than large populations (Freeman and Herron 2001; Futuyma 1998). Furthermore, new areas often have different selective pressures, and these may have favoured different genotypes than ones having success in previous wildcat territories (Freeman and Herron 2001; Futuyma 1998). A range expansion is also predicted to have occurred from Iberia into eastern Europe (Figure 2c: route 2b). However, because the length of migration is suggested to be only a few 100's of kilometres, the degree of morphological and genetic divergence is also estimated to be less.

The RBM outcomes suggest that Africa may contain several separate wildcat subspecies. Three main populations that are weakly connected exist in north, central, and southern Africa as separated by the Sahara and the equatorial rainforests (Figure 2d; splits 1b, 1c). Subspeciation may have occurred in these populations. Genetic analysis has shown animals living along the fringes of one population to show morphologic and genetic distinctions from the same species found living along the fringes of other populations (Freeman and Herron 2001; Futuyma 1998). The fact that the connections between the African populations are modelled to be tenuous would only serve as another factor impeding the mixing of genes between populations. Furthermore, the modelled stability of populations through the Late Quaternary implies that there were no mass migrations from one region to another, thus, also decreasing the probability of genetic exchange between populations. Kitchener (2002) voices support for a more distinctive southern African wildcat population because south African wildcats have a different skull shape than other African wildcats. However, support for a

[4] The presence of leopard cats in south-east Asia are believed to be keeping wildcats out of this area (Kitchener, 2002).

taxonomic distinction between central and northern wildcats was not found from current knowledge of wildcats. And given the palaeo-geological record, the modelled split between these populations is considered unlikely. At the height of the interglacial maximum about 9,000 years ago a well integrated drainage network existed in the Sahara that contained many freshwater lakes and ponds (Williams *et al*, 1998). This would likely have provided prime habitat for wildcats, thus, connecting the distribution between central and northern Africa. Since the RBM does not model changes in precipitation with climate change it is unable to model modifications in vegetation that are caused by shifts in global precipitation. As a consequence, the Sahara is modelled as unfavourable *hot desert* wildcat habitat throughout the Late Quaternary, thus presenting itself as a permanent barrier to northern and central African populations.

4.1.2 Areas where the RBM predicts wildcat presence but wildcats are known to be absent

Western Madagascar, Australia, Japan, and Ireland are potentially prime wildcat habitats because these areas are modelled as having high wildcat habitat preferences and leopard cats are not known to be living these areas. Wildcats are not found in Madagascar and Australia because these landmasses separated from their land connections with Europe millions of years before wildcats first evolved from there[5]. Aside from Madagascar and Australia, the RBM outcomes can help explain why wildcats are absent from Japan and Ireland. Japan was still attached to the Eurasia continent along its northern and southern extents about 18,000 years ago and then only by its northern extents by about 13,000 years ago (Figure 2c, d). Despite having a northerly land connection, the RBM outcomes suggest wildcats were prevented from entering Japan from the north because it was surrounded by habitat wildcats avoid. Infiltration from the south may have been possible because appropriate habitat is predicted to have existed, however, the assumed presence of the leopard cats in south-east Asia should have

[5] The fossil record indicates that wildcats first evolved in Europe from the Martelli's cat about 250,000 years ago (Nowell and Jackson, 1996). This would have occurred 10's of millions of years after Madagascar and Australia became separate landmasses (Williams *et al*, 1998), thus, denying any chance of wildcats migrating into these areas.

impeded an easterly wildcat migration into Japan. Ireland remained connected to the Eurasian continent even longer that Japan but is still thought to never have had wildcats (Kitchener, 2002). The RBM outcomes suggest that Ireland became separated from the mainland before it developed habitat appropriate for wildcat survival (Figure 2b, c, d).

An important aspect of predicting wildcats to thrive in these areas is what it may mean for feral and domestic cats and the ecosystems they affect. Taking the RBM predictions one step further would also imply that feral and domestic cats should have success in these areas, since they are such closely related species with similar survival strategies and instincts. The implication is that unchecked feral and domestic cat populations have the potential to decimate small native animal populations. This is because feral and domestic cats would enjoy high survival rates under favourable habitat conditions. As a consequence, their populations would blossom, and this would lead to too many feral and domestic cats killing prey at levels unsustainable for the native prey populations. This has already been shown to be the consequence of domestic and feral cats having success in many areas around the world (e.g. Hawaii); (Begon *et al*, 1990).

4.2 <u>Implications of the Biogeographical Approach</u>

4.2.1 Prediction Success relative to Known Sightings Data

Comparing the RBM to the known wildcat distribution can only assess how good the model is at predicting the distribution within areas where wildcats have been found. The RBM is considered to be effective at predicting wildcat presence in areas of known sightings data because the overall correspondence of the predicted presence and the known presence is 83 percent. However, the RBM has also predicted wildcat presence outside of the areas where wildcats have been found. This is not surprising because the RBM is a deductive model. It extrapolates from known habitat requirements to identify *all* areas that fulfil these requirements, whereas, the sightings data is only a minimum distribution, because it is unrealistic to record where every wildcat has lived. Since the sightings data is not complete, it is not for certain that RBM distribution predictions outside the known distribution are incorrect (Table 6).

4.2.2 Prediction Success relative to the Biome and Secondary Biome models

The high level of correspondence between the present day RBM and the present day BM and SBM models further suggest the RBM is successful at predicting the modern day wildcat distribution. Differences in the various model outcomes exist because of where the environmental regions are located (e.g. *steppe* in Arabia for the SBM vs. *desert* in Arabia for the RBM), and the rankings assigned to the environmental regions (e.g. 1 for *tropical seasonal forest* for the BM vs. 0 for *tropical seasonal monsoon* for the RBM[6]).

Table 6. A summary of the reasons why the RBM may correctly or incorrectly predict wildcat presence in areas outside of the known wildcat distribution.

Correct Prediction	Incorrect Prediction
Wildcats occur in areas predicted by the RBM, but the sightings data lacks information on their presence.	The model defines an area as being suitable habitat for wildcats when in reality those conditions would be avoided.
The wildcat population is not currently large enough to saturate these habitats.	Under natural conditions these areas would be suitable habitat, but are currently unsuitable because of high human impact (e.g. intensive agriculture, urban areas).
	Other inhabiting species prevent the wildcat from establishing itself. For example, the presence of the leopard cat in south east Asia (Figure 1 region 1) are thought to be prevented wildcats from inhabiting this area (Kitchener, 2002).

The decrease in correspondence between the RBM and BM for past reconstructions implies the RBM has less success predicting past distributions. There are several reasons that compromise the ability of the RBM to predict past distributions. (1) Of the three possible environmental variables used in the RBM to simulate climate change, temperature, precipitation, and continentality, only temperature and continentality are used. By neglecting precipitation, the RBM cannot account for changes to vegetation zones that were highly influenced by increases and decreases in precipitation that occured as the climate shifted from the drier glacial maximum to the wetter glacial minimum (Williams *et al*, 1998). (2) The

[6] BIOME 6000 biome type *tropical seasonal forest* and Koppen Climate System climate type *tropical seasonal monsoon* are assumed to be the same type of environmental region.

RBM environmental variables are not able to simulate all the climate change factors that would have had an effect on global habitats (e.g. wind circulation, dustiness in the atmosphere, seasonal solar radiation, and glaciation). (3) The quality of the continentality and temperature factors driving climate change are not ideal. Other factors influencing continentality over the Late Quaternary are not included (e.g. changes in sea and lake ice coverage), and temperature differences between the poles and the equator would not have occurred as smoothly as modelled by the RBM (see Rees (2002) for details).

5 CONCLUSIONS

The biogeographical approach used in this study provides a relatively simple procedure for reconstructing the recent evolutionary history of wildcats, and in doing so, sheds light on their taxonomic status. From the RBM outcomes a number of hypotheses about wildcat taxonomy and its evolutionary history have arisen that could work with traditional taxonomic techniques by either supporting or rejecting their conclusions, or by providing hypotheses that can be tested by these techniques.

- Asian and European wildcats are more morphologically and genetically distinct from each other than Asian and African or European and African wildcats.
- South African wildcats show morphological and genetic distinctions from other African wildcats.
- Wildcat populations in northern China contain the most genetically and morphologically divergent wildcats of the entire species.
- The Iberian Peninsula and areas along the northern Mediterranean shores, and the Gulf Region were wildcat refugia over the Late Quaternary.
- The distributions of African wildcat populations were relatively unaffected by the Late Quaternary climate change.

Conclusions can also be drawn about the RBM.

- The present and past wildcat distributions can be reconstructed by only using data on temperature, precipitation, and topography.
- The predicted distribution is the maximum distribution that would occur in the absence of factors other than the environmental variables used to run the model (e.g. human impact).
- The reconstructed distribution depends on how and where environmental regions are defined and the wildcat preference ranking assigned to them.

Aside from clarifying taxonomy, biogeographical analysis can contribute other valuable information for developing conservation strategies. By reconstructing the maximum potential distribution there is more of a potential to highlight areas where previously overlooked wildcats maybe found, identify areas that would be good habitat for re-introducing wildcat populations, and for identifying the most preferable habitats for conserving the species. Also the biogeographical approach can be used to show how various factors may affect the distribution differently. For instance, the last glacial maximum is predicted to have reduced the wildcat distribution. Human impact has also decreased the distribution as seen by comparing the known sightings data to the present day RBM outcome (Figure 1). However, humans appear to have a much greater effect on the distribution than the continental ice sheets and the cooler and drier global climate of the last glacial maximum ever did.

Biogeographical analysis offers a fresh perspective for unravelling taxonomy. It does not require field samples for analysis or define biological taxa on the basis of morphological, behavioural, or genetic criteria. For inconspicuous species like wildcats, biogeographical analysis is promising approach because it is often difficult to obtain adequate field samples that objectively represent the morphological, behavioural, or genetic nature of the species.

5.1 Future Work

The biogeographical approach used in this study is a promising technique for clarifying subspecies taxonomic classifications. Future work in the following areas should enable increase the ability of the RBM to derive more accurate evolutionary histories of the species needing taxonomic clarification:

- Use precipitation, in addition to temperature and continentality, to simulate climate change.
- Use more complex spatial analytical techniques to assess degree of fragmentation in the distribution.
- Explore how seasonality and the changing monsoon climate over time effects the location and duration of fragmented distributions through the Late Quaternary.
- Include spatial data on human population density and agriculture to investigate how these factors effect the modern day outcome.

6 REFERENCES

Balharry, D., Daniels, M., and Barratt, E.M. 1997. Wildcats: can genetics help their conservation? The role of genetics in conserving small populations. Edited by Tew, T.E., Crawford, T.J., Spencer, J.W., Stevens, D.P., Usher, M.B., and Warren, J. Peterborough: JNCC. 203 pp.

Beamont, M., Barratt, E.M., Gottell, D. Kitchener, A.C., Daniels, M.J., Pritchards, J.K., and Bruford, M.W. 2001. Genetic diversity and introgression in the Scottish wildcat. *Molecular Ecology*. **10**: 319-336.

Begon, M. Harper, J.L., and Townsend, C.R. 1990. Ecology: Individuals, Populations and Communities. 2nd Edition. Boston: Blackwell Scientific Publications. 945pp.

Corbet, G.B. 1978. The Mammals of the Palaearctic Region: a taxonomic review. British Museum (National History), Cornell University Press. 314pp.

Corbet, G.B. and Hill, J.E. 1992. *The Mammals of the Indomalayan Region: A Systematic Review*. Oxford: Oxford University Press. 488pp.

Daniels, M.J., Balharry, D., Hirst D., Kitchener, A.C., and Aspinall, R.J. 1998. Morphological and pelage characteristics of wild living cats in Scotland: implications for defining the 'wildcat'. *Journal of Zoology*. **244**: 231-247.

Daniels, M.J., Beaumont, M.A., Johnson, P.J., Balharry, D., MacDonald, D.W., and Barratt, E. 2001. Ecology and genetics of wild-living cats in north-east of Scotland and the implications for the conservation of the wildcat. *Journal of Applied Ecology*. **38**: 146-161.

Fielding, A.H. and Bell, J.F. 1997. A review of methods for the assessment of prediction errors in conservation presence/absence models. *Environmental Conservation*. **24(1)**: 38-49.

Freeman, S. and Herron, J.C. 2001. *Evolutionary Analysis*. 2nd edition, Prentice Hall, Upper Saddle River, NJ. 704 pp.

Futuyma, D.J. 1998. *Evolutionary Biology*. 3rd edition, Sinauer Association, Inc., Sunderland, Massachusetts. 763 pp.

Green, S.E. 2000. *Modelling the global distribution of the wildcat, Felis silvestris*; Unpublished M.Sc. in GIS Dissertation Project. School of Earth, Environmental and Geographical Sciences, University of Edinburgh.

Happold, D.C.D. 1987. The Mammals of Nigeria. Oxford: Clarendon Press. 402pp.

Harrison, D.L. and Bates, P.J.J. 1991. *The Mammals of Arabia, 2nd Edition*. Kent: Harrison Zoological Museum. 354pp.

Kitchener, A. and Dugmore, A.J. 2000. Biogeographical change in the tiger, *Panthera tigris*. *Animal Conservation*. **3**: 113-124.

Kitchener, A.C. 2002. Department of Zoology, National Museums of Scotland. Personal communication.

Mitchell-Jones, A.J., Amori, G., Bogdanowicz, W., Krystufek, B., Reijnders, P.J.H., Spitzenberger, F., Stubbe, M., Thissen, J.B.M., Vohralik, V., and Zima, J. 1999. *The Atlas of European Mammals*. London: T & AD Poyser Ltd.

Nowell, K. and Jackson, P. 1996. *Status Survey and Conservation Action Plan: Wild Cats*. International Union for Conservation of Nature and Natural Resources, Gland, Switzerland. 382 pp.

O'Brien, S.J. 1996. Subspecies Identification Incorporating Molecular Genetics. In *Status Survey and Conservation Action Plan: Wild Cats*: 210-211. Nowell, K. and Jackson, P. International Union for Conservation of Nature and Natural Resources, Gland, Switzerland. 382 pp.

O'Brien, S.J. and Mayr, E. 1991. Bureaucratic mischief: recognizing endangered species and subspecies. *Science*. **251**: 1187-1188.

Prentice, C. and Webb, T. 1998. BIOME 6000: Reconstructing global mid-Holocene vegetation patterns from palaeoecological records. *Journal of Biogeography*. **25**: 997-1005.

Pennock, D.S. and Dimmick, W.W. 1997. Critique of the Evolutionarily Significant Unit as a Definition for "Distinct Population Segments" under the U.S. Endangered Species Act. *Conservation Biology*. **11(3)**: 611-619.

Rees, E.R. 2002. *GIS, Environmental Change, and the Late Quaternary Distribution of the Eurasian/African Wildcat (Felis silvestris)*. Unpublished M.Sc. in GIS Dissertation Project Technical Report. School of Earth, Environmental and Geographical Sciences, University of Edinburgh.

Roberts, T.J. 1997. *The Mammals of Pakistan*. Oxford: Oxford University Press. 525pp.

Skinner, J.D. and Smithers, R.H.N. 1990. The Mammals of the Southern African Subregion. Pretoria, Transvaal, Republic of South Africa: University of Pretoria. 769pp.

Williams, M., Dunkerley, D., De Decker, P., Kershaw, P., and Chappell, J. 1998. *Quaternary Environments*. 2nd edition, Arnold, London. 329 pp.

7 BIBLIOGRAPHY

Guisan, A. N. and Theurillat, J.P. 2000. Equilibrium modeling of alpine plant distribution: how far can we go? *Phytocoenologia*. **30(3-4)**: 353-384.

Guisan, A. N. and Zimmermann, N.E. 2000. Predictive habitat distribution models in ecology. *Ecological Modelling*. **135**: 147-186.

Jolly, D., Prentice, C. and Bonnefille, R. 1998. Biome reconstruction from pollen and plant macrofossil data for African and the Arabian peninsula at 0 and 6000 years. *Journal of Biography*. **25**: 1007-1027.

Mackey, B.G. and Lindenmayer, D.B. 2001. Towards a hierarchical framework for modelling the spatial distribution of animals. *Journal of Biogeography*. **28**: 1147-1166.

Manel, S., Williams, H.C., and Ormerod, S.J. 2001. Evaluating presence-absence models in ecology: the need to account for prevalence. **38**: 921-931.

Manel, S., Dias, J.M., Buckton, S.T., and Ormerod, S.J. 1999. Alternative methods for predicting species distribution: an illustration with Himalayan river birds. *Journal of Applied Ecology*. **36**: 734-747.

GIS, ENVIRONMENTAL CHANGE, AND THE LATE QUATERNARY DISTRIBUTION OF THE EURASIAN/AFRICAN WILDCAT (*FELIS SILVESTRIS*)

PART 2: Technical Report

Erin E. Rees

September 2002

TABLE OF CONTENTS

APPENDICES

LIST OF FIGURES

LIST OF TABLES

1 SUMMARY

The Technical Report is intended as a supporting document for the Research Report. The first few topics include background information about wildcat ecology, the time period of the study, and the modelling process. This is followed by an in-depth description of the data used in this study and the methodology. Key files used in this study are documented in the Technical Report with their full path and file name as indicated in italics, for example: *dissertation/aml/cat0k*. The remaining files are listed in the Appendices with the relevant commands.

2 GENERAL INFORMATION ABOUT WILDCATS

The entire wildcat distribution covers an extensive area, ranging from southwest Europe, throughout Africa, and across Southern Asian. Given the distribution is much larger than most naturally occurring animal distributions it is possible the total wildcat population is instead a combination of several subspecies or even separate species. Many different subspecies classifications have been assigned to wildcats (Corbet, 1978; Corbet and Hill, 1992; Happold, 1987; Harrison and Bates, 1991; Kitchener, 2002; Mitchell-Jones *et al*, 1999; Nowell and Jackson, 1996; Roberts, 1997). This study recognises the classification designated by Nowell and Jackson (1996) which separates wildcats into three groups: *Felis silvestris silvestris*, *Felis silvestris lybica*, and *Felis silvestris ornata*. *F.s.silvestris* is found predominately in southern Europe extending as far east as the Caspian Sea. *F.s.lybica* ranges throughout Africa and the Middle East, and *F.s.ornata* covers the areas east of the Caspian Sea through northern India and into China. Within the three main groupings of wildcats there is considerable overlap. Where these areas adjoin, it can be difficult, if not impossible, to distinguish one subspecies from another (Roberts, 1997). As geographically separation between the subspecies increases, differences between phenotype, behaviour, and habitat preferences become more noticeable (Roberts, 1997).

2.1 <u>Description</u>

The European wildcat is greyish brown in colour with a well-defined pattern of black stripes, and a blunt-ended tail. In the winter these wildcats develop thick coats. Males tend to weigh about 5 kg and females about 3.5 kg, but their weight has been found to fluctuate with the seasons (Nowell and Jackson, 1996). Asian wildcats are more greyish-yellow than European wildcats or have a stronger reddish background colour. Their coats are marked with small black or red-brown spots which are sometimes joined to form stripes. The males are 3 to 4 kg in size while the females are just under 3 kg (Nowell and Jackson, 1996). Wildcats found in Africa and the Middle East show the largest variation in phenotype. There is a north to south gradient of coat thickness, colour, tabby markings, (Nowell and Jackson, 1996), and length of foot hairs (Harrison and Bates, 1991). The coat colour ranges from reddish to sandy yellow, to tawny brown to grey, and is usually marked with faint tabby stripes or spots. The backs of their ears tend to be reddish or rusty brown in colour. Wildcats found in more arid areas tend to have longer hairs on the pads of their feet (Harrison and Bates, 1991). In contrast to European wildcats, African wildcats are usually lighter in build, have fewer distinct markings, and have thinner tapered tails (Nowell and Jackson, 1996). Male wildcats in southern African were found to average 5 kg in size, while females were approximately 4 kg (Skinner and Smithers, 1990). Of the three main groups, African wildcats are perhaps the most similar in appearance to domestic cats (Harrison and Bates, 1991).

2.2 Behaviour

Wildcats are similar in behaviour across their distribution. The differences that are apparent are a consequence of the characteristics of the area they inhabit. European wildcats are opportunistic feeders, hunting or scavenging for food (Mitchell-Jones *et al*, 1999). They mostly eat rodents, but will also consume rabbits and birds. Scottish wildcats are primarily nocturnal. They will travel around 10 km a night forging for food over the open ground, near coasts, and in farms. During the day they rest in thickets or in young tree plantations (Nowell and Jackson, 1996). Similar to European wildcats, Asian wildcats prefer rodents, but also eat hares, young ungulates, birds, insects, lizards, and snakes. Asian wildcats rest and den in burrows. These cats have frequently been observed during the day (Nowell and Jackson, 1996). African wildcats are also active during the day, especially during the early morning and late afternoon. In exceedingly hot environments these wildcats are primarily nocturnal. African wildcats consume mostly rodents, but will also eat young ungulates, birds, reptiles, amphibians, insects, and arachnids. African wildcats are known to be very solitary and secretive in nature (Happold, 1987; Nowell and Jackson, 1996; Skinner and Smithers, 1990).

2.3 Habitat Preferences

2.3.1 European Wildcats

European wildcats avoid areas of high human population density (Daniels *et al*, 2001; Nowell and Jackson, 1996). These wildcats are primarily associated with forested habitat. They show preference to deciduous broad-leaved forests over coniferous forests (Bernhart, 1996; Daniels *et al*, 2001; Nowell and Jackson, 1996), especially forests dominated by oak, then followed by beech and mixed forests (Mitchell-Jones *et al*, 1999). Because of habitat fragmentation and destruction wildcats have been forced to live in more marginal habitat such as montane areas, coniferous forests, swamps, forest plantations, and agricultural areas used for grazing (Corbet, 1978; Nowell and Jackson, 1996). According to Corbet (1978) wildcats are found on all the Mediterranean islands. In this region they live in maquis scrubland, riparian forest, marsh boundaries, and along seacoasts. Wildcats are not found in the high Alps, or areas where snow cover is greater than 50%, more than 20 - 30 cm deep (Bernhart, 1996; Mitchell-Jones *et al*, 1999), and remains for greater than 100 days of the year (Nowell and Jackson, 1996). In mountainous areas wildcats have been found up to 650 m in Scotland's Grampian mountains (Daniels *et al*, 2001), and up to 1200 m in the Swiss Jura mountains (Bernhart, 1996).

2.3.2 Asian Wildcats

Asian wildcats are most commonly found in scrub desert. They usually live close to water, but can live year round in waterless deserts. In mountainous areas with sufficient vegetation wildcats will range up to 2000 - 3000 m. Wildcats are also found near cultivated areas and close to human settlements. Asian wildcats are not found in alpine steppe, or the steppe grasslands of Mongolia. Furthermore, high snow depths limit the northern boundary of their distribution (Nowell and Jackson, 1996). In Pakistan, wildcats are not found along alluvial plains, human populated cultivated tracks, high exposed mountains, and in the northern Himalayan mountains or valleys. Instead, wildcats are found in sparse numbers in the deserts and the dry hilly regions on the west bank of the Indus River (Roberts, 1997). In the Caucasus, the transition zone from Asian to European wildcats, Asian wildcats tend to remain in low-lying desert and semi-desert areas by the Caspian Sea, while nearby European wildcats inhabit the montane forests areas (Nowell and Jackson, 1996).

2.3.3 African Wildcats

African wildcats have a broad habitat tolerance. For such a large distribution they only appear to avoid tropical rain forest (Nowell and Jackson, 1996) and montane forest (Skinner and Smithers, 1990). The wildcats range up to 3000 m in the mountains of Kenya, Ethiopia, and Algeria (Nowell and Jackson, 1996), and in the African subregion they are found between sea level and 1600 m (Skinner and Smithers, 1990). Wildcats are absent from areas with less than 100 mm annual rainfall, except where they can meet their water needs along river courses and mountainous areas[1]. Throughout their range wildcats seek out cover for rest during the day. Appropriate cover includes rocky hillsides, underbrush, reedbeds, stands of tall grass or crops (e.g. maize), vacant fox lairs, and caves (Happold, 1987; Harrison and Bates, 1991; Skinner and Smithers, 1990). In semi-deserts, wildcats seek refuge in stands of scrub bushes, while in plains

[1] Mountains often catch greater rainfall than the surrounding drier areas (Williams *et al*, 1998)

country, wildcats lie in dens made by other species, naturally occurring holes in the ground, or under roots of trees (Skinner and Smithers, 1990).

2.4 Population Status of Wildcats

The wildcat population is relatively strong in Africa (Nowell and Jackson, 1996). In Europe and Asia, human impact has seriously reduced the population because wildcat habitat is being lost through high human population density and intensive agricultural practices (Nowell and Jackson, 1996). Fortunately in Europe, efforts to reclaim native vegetation since the 1900's have resulted in an increase in wildcat numbers (Beaumont *et al*, 2001). Loss of habitat makes wildcats more vulnerable to other pressures threatening their survival. Three main factors affecting their continued existence are: (1) hybridisation with domestic cats, (2) difficulty distinguishing wildcats from feral or domestic cats, and (3) shooting of wildcats in areas where killing feral cats is permitted (Balharry *et al*, 1997; Skinner and Smithers, 1990).

2.5 The Importance of Protecting the Wildcat

It is important to conserve species threatened by human impact because that is key to preventing loss in biodiversity. Biodiversity refers to the number of biologically diverse species. Aside from preserving species for human needs (e.g. food, medicine, aesthetic value), maintaining biodiversity is crucial for the continued existence of the earth's species, including humans. Life on earth has developed as an interconnected web of species. Species rely on each other for survival. For example, soil bacteria breakdown dead plant matter and return nutrients to the soil to be re-used by living plants, wolves depend on caribou and lemmings for food, and apple trees require insects to feed on the nectar produced by their flowers in order for pollination to occur, and so on. As species go extinct, the web begins to breakdown. At a small scale this is not a problem. Extinction is a natural phenomenon - loss of one species makes room for the population of a current species to expand and this can result in the creation of new species (Freeman and Herron, 2001; Futuyma, 1998). However, extinction at a large scale can snowball into a staggering loss of life. Throughout the earth's history of life there have been several major extinction events, that is, periods where over 60% of species become extinct within a million years (Freeman and Herron, 2001). The cause of mass extinctions is not fully understood. It could be that under certain circumstances the earth is more prone to a mass extinction, which can then be triggered by an additional circumstance, such as a meteor impact, or, mass extinctions may simply be a part of the natural cycle of life.

It is possible the earth is experiencing another massive extinction event. However, unlike the previous events, the speed as which this is one is occurring is unprecedented, *and* it is being caused by a biological species, humans, through habitat loss and destruction (Begon *et al*, 1990). If too many species become extinct it is feared the loss of species will once again snowball and decimate life on earth. Therefore, it is necessary to conserve species to help prevent loss of biodiversity and this should decrease the risk of a mass extinction event.

2.6 Uncertainty in the Criteria used to Define Taxonomy

There exist a plethora of concepts specifying criteria for differentiating species from one another (Freeman and Herron, 2001; Futuyma, 1998). No one concept has emerged as the single, definitive definition because it is still unclear what this should be. It may be more appropriate to perceive species variability along a gradient, however, humans tend to categorise the perception of the world and species concepts are no exception. One of the leading concepts is the Biological Species Concept (BSC) developed by Mayr in 1940. The BSC defines species as "groups of actually or potentially interbreeding populations that are reproductively isolated [because of morphological or genetic differences] from other such groups." (O'Brien and Mayr, 1991). The BSC has since been revised to include geographic range and habitat as additional criteria for defining subspecies beyond morphological or genetic criteria (O'Brien and Mayr, 1991). For species such as wildcats, which are difficult to classify by morphological and genetic criteria, species concepts that include other criteria such as habitat may be helpful for defining taxonomy (Beaumont *et al*, 2001).

3 TIME PERIOD FOR THE STUDY

The time period chosen for this study is the Late Quaternary, as defined as the past 20,000 years. There are two main reasons for this. Firstly, within this time period there was a glacial maximum at about 18,000 years ago and an interglacial maximum at approximately 9,000 years ago (Williams *et al*, 1998). Within this climatically variable time period, there exists the potential to test whether climate change influenced the range of the wildcat distribution. For example, the species range may have been pushed to a minimum by the expanding glaciers and conversely increased to a maximum in the absence of continental ice sheets. The second reason for choosing the Late Quaternary is that because it contains the most recent glacial cycle, the data recording this time period will be the best-dated, the best-preserved, and contain the most abundant palaeoclimatic data (Williams *et al*, 1998).

3.1 The Late Quaternary

The global climate during the Late Quaternary experienced a high degree of variation. There are many ways these changes presented themselves, for example, through episodes of increased dryness or wetness, heat or coolness, changes in sea level and consequently continentality, changes in windiness, fluctuations in solar radiation, and increases or decreases of global glacial coverage. In general, global climates during glacial maximums are cooler and drier, and interglacial maximums are warmer and wetter than present day conditions (Williams *et al*, 1998).

3.1.1 The Last Glacial Maximum

During the glacial maximum ice covered northern and western Europe, the Alps, much of North America and Antarctica, and parts of South America. Extensive glaciation has a huge effect on the global climate. Glaciers lock up water from the atmosphere and the oceans. This creates a drier climate that would have increased desertification and deceased the strength of the monsoons. The Gobi and Uzbekistan hot deserts were drier, windier, and colder than today, but their summers may still have been very hot. Sea levels were about 120 to 150 m lower than their current levels. Land bridges would have appeared, for example, connecting Britain to mainland Europe. The expansive coverage of ice would have lowered the earth's albedo. This means a higher degree of incoming short-wave radiation from the sun would have been reflected, thus reducing the amount of the sun's energy absorbed by the earth and reinforcing the cooler climate. Colder climatic zones would have extended further from the poles, and tropical climatic zones would have been reduced in size. On average, the global climate was about 9°C cooler than today (Williams *et al*, 1998).

3.1.2 The Last Interglacial Maximum

In contrast, the last interglacial global environment at about 9k, was warmer and wetter than present day conditions (k = 1000 years before present). Glaciers no longer dominated such expansive areas, but were instead restricted to mountainous areas, and Greenland and Antarctica. A volume of ice that took about 90k years to accumulate melted in only 8k. Sea levels were about 4 to 10 m higher than present day levels. The global climate was also about 2°C warmer than present day (Williams *et al*, 1998). A warmer climate with higher sea levels would have resulted in a higher rate of evaporation from inter-tropical oceans. Consequently, monsoons would have been stronger than present day occurrences and far stronger than during the glacial maximum. Tropical areas that had dried out during the glacial maximum would have been able to replenish their aquifers and raise their groundwater levels. Mobile dunes would have stabilised as they became more vegetated. Semi-arid regions would have become savannah woodland and grassland. There is even evidence of many fresh water lakes and ponds and a well-integrated drainage network that coursed through the Sahara, Arabia, and Rajasthan deserts (Williams *et al*, 1998).

3.1.3 Shifting between Glacial and Interglacial Maximums

It is important to realise with reference to this study that the transition between glacial and interglacial maximums is not uniform gradual process. Evidence from ice cores and sea level change indicators suggest climate change has progressed as periods of sudden jumps and moments of slower advancement. For example, according to the Barbados record detailed by Fairbanks in 1989, the sea level began to rise slowly at about 20k by 5 m/k. At 14k the rate increased rapidly to 37 m/k. Around 12k the level rose 24 m in less than 1000 years, and then the rate then slowed to 8 m/k, to finally jump once more to 25 m/k at approximately 11-10k (Williams *et al*, 1998). Thus, changes in species distribution also may have shown an erratic progression between glacial and interglacial maximums.

4 THE MODELS

The Rule Based Model (RBM) is the primary model for this study, and the Biome Model (BM) and the Secondary Biome Model (SBM) are secondary models used to evaluate the RBM.

4.1 The Rule Based Model

The RBM is being tested as a novel GIS approach towards reconstructing the evolutionary histories of a species. Global environmental regions (i.e. areas of unique climate and vegetation) are reconstructed from nine environmental land surface variables: temperature range (trange), mean annual temperature (ctmp), mean January temperature (tjan), mean July temperature (tjul), mean annual precipitation (cpre), mean January precipitation (pjan), mean July precipitation (pjul), continentality (cont), and topography (dem). The environmental regions are then ranked according to known wildcat habitat preference. Mapping this information reconstructs the distribution because it indicates wildcat presence with respect to having a preference for or against the environmental regions. Altering the environmental variables to simulate climate change enables past distributions to be reconstructed for 9k, 13k, and 18k. There are several advantages of basing the RBM on the environmental variable data:

- Information deriving the environmental variables can be downloaded for free from the Internet.
- The environmental variables are a relatively primary source of data. Therefore, the information should be nearer to the true values than data that has been processed by environmental models.
- Environmental variable data can reveal the habitat characteristics of an area that would occur in the absence of human impact.
- The environmental variables include no data on how other species may affect the wildcat distribution (e.g. species competing for the same food sources). Therefore, all the potential environmental regions where wildcats would thrive or avoid can be identified.

4.1.1 The Koppen Climate System

Twenty-one of the thirty-two environmental regions composing the RBM are based on the Koppen Climate System (KCS) classification scheme. The KCS is a well known and well used climate classification scheme developed by Dr. Vladimir Koppen of the University of Graz, Austria, in 1918. Climate types are defined according to mean temperature and precipitation values (Strahler and Strahler, 1997). The classification scheme does not account for factors such as cloudiness, solar radiation, wind, or extremes in temperature, however, it is still considered by many to be a useful system. (Marsh, 2000). Since its development the classification scheme has been modified. This study follows the scheme devised by R. Geiger and W. Pohl (1953) as documented in Strahler and Strahler (1997), and the scheme devised by G. Trewartha as documented in Bartholomew *et al* (1990). The KCS is composed of six major climate regions that are further subdivided to more specifically defined climate types (Appendix One 16.1). While maps of the KCS show clearly defined boundaries, it is important to note that the system is just a guide to average climate trends, the areas between zones occur in reality as a gradual transition between climates.

4.2 The Biome Model and the Secondary Biome Model

The BM and SBM are developed to evaluate the RBM. The BM and SBM are formulated using modern day wildcat sightings data and biome data from the BIOME 6000 Project. *Biomes* are major regional or global biotic communities, such as a grassland or desert, that are chiefly characterised by the dominant forms of plant life and the prevailing climate. Intersecting the sightings data with the biome data identifies biomes wildcats inhabit. Statistical analysis is used to determine how meaningful each biome is for predicting wildcat presence. This information is used to rank the biomes relative to wildcat preference. Mapping the biome types yields the wildcat distribution because it indicates presence relative to preference for or against the biomes. The BM goes a step further than the SBM and reconstructs past distributions by ranking BIOME 6000 biome data for 9k and 18k.

4.2.1 The BIOME 6000 Project

The BIOME 6000 Project's primary aim is to compile a global data set of palaeodata that can be compared with the outputs of models that derive present and past climates (Prentice and Webb, 1998). Biomes are assigned to a region using the *biomization* process. Biomization occurs over a series of steps. The first step is to assign each plant species to one or more plant functional type (PFT) as based on the biology (e.g. leaf form, habit, phenology) and biogeography (e.g. bioclimatic tolerance) of the plant species. Examples of PFTs include tropical evergreen tree, temperate summer green tree, and cool-temperate conifer tree. PFTs are then assigned to one or more biome. There is no standard way of classifying plant taxons into PFTs or PFTs into biomes (Prentice and Webb, 1998). BIOME 6000 follows the biome classification scheme developed for BIOME1 as outlined in Prentice *et al* (1992). The only differences for the BIOME 6000 data being that 'warm grass/shrub' and 'cool grass/shrub' have been merged into 'steppe', and 'semi-desert' and 'hot desert' combined to form 'desert' (Prentice and Webb, 1998).

4.3 Inductive and Deductive Modelling

Scientific reasoning usually follows an inductive or deductive approach. This study uses both perspectives for creating its models. Inductive reasoning draws conclusions from the results of numerous observations (Harvey, 1973). For example, if every observed wildcat has been sighted in forest habitat an inductive reasoner would conclude that all wildcats live in forest habitat. The major fault of this approach is that it does not guarantee that all the succeeding wildcats spotted will be found in forest habitat. Whereas, deduction refers to the process of concluding that something must be true because it is a special case of a general principle that is already known to be true (Harvey, 1973). For instance, a hypothesis can be tested that wildcats are found in forest habitat. By analysing a number of observations of where wildcats have been sighted this hypothesis can be rejected or accepted. A disadvantage of the deductive approach is that it does not identify which hypotheses are meaningful ones to test. For example, if it was not known which habitat wildcats prefer, it might take a while to select a habitat for hypothesis testing that yields useful results.

Inductive and deductive reasoning can be used together to overcome their individual disadvantages (Harvey, 1973). Inductive reasoning can be part of a discovery process that may lead one to suspect that some general principle is true from observing a number of cases. Deductive reasoning can then be used to create hypotheses to test these suspicions and demonstrate with logical certainty whether they are true or not. Inductive and deductive approaches can also be kept separate to highlight the differences and similarities between the two approaches (Harvey, 1973), as is done in this study. The BM and SBM are inductive models. Using wildcat sightings data, all the habitats where wildcats are found are identified. The identified habitats are assumed to be the preferred wildcat habitats. This information is used to reconstruct present and past distributions. The BM and SBM can only create a minimum distribution because it is unrealistic to assume that an inconspicuous species such as the wildcat has been observed in every habitat they prefer. Furthermore, wildcats may be absent from a habitat they would normally inhabit because of human impact, such as the presence of a city or farmland. The RBM is a deductive model. Current knowledge about wildcats is used to identify all possible habitats wildcats may occupy. This creates a maximum distribution of wildcats because it identifies all areas wildcats might be regardless of wildcats having been observed there or not.

5 DATA SOURCES

5.1 Modern Day Wildcat Distribution

Description: Modern day wildcat distribution data is based on known sightings of living and dead wildcats. This information comes from a variety of sources. An Arc/Info 8.0 polygon coverage of wildcat sightings developed by Green (2000) was used which was based on *Catalogue of the genus Felis* (Pocock, 1951). The coverage represents 170 sightings geocoded with a latitude and longitude reference to the nearest 5 minutes. Data digitised for the study came from Harrison and Bates (1991), Kitchener (2002), Nowell and Jackson (1996), Zhang Yongzu *et al* (1997), to create point and polygon Arc/Info coverages. All sightings data was projected to an Eckert VI projection. Sightings data coming from literature was found in: *The Atlas of European Mammals* (Mitchell-Jones *et al*, 1999), *The Mammals of the Southern Africa Subregion* (Skinner and Smithers, 1990), and *Mammals of Pakistan* (Roberts, 1997).

Quality: It is impossible for the sightings data to be complete because it is unfeasible to obtain a complete census of such a shy and secretive species (Happold, 1987; Nowell and Jackson, 1996; Skinner and Smithers, 1990). Furthermore, there exists the possibility that feral or feral/wildcat hybrids have been mistakenly recorded as wildcats in the sightings data. Given these considerations, the electronic data sources are considered reasonably accurate for the enormous geographic resolution for which they are being applied. The quality of the digitised distribution is likely the poorest, with an estimated uncertainty of ± 200 km. Uncertainty arises from: the source maps not being high quality, data distortion developing from scanning in images for 'heads-up' digitising, and data distortion developing through the process of digitising.

5.2 Modern Day Leopard Cat Distribution

Description: Digitising distribution data from Nowell and Jackson (1996) created the modern day leopard cat distribution.

Quality: The quality is the same as the digitised wildcat distribution data, as described in Section 5.1.

5.3 Topography

Source: ftp://topex.ucsd.edu/pub/global_topo_2min/

Format: image file ('.img')

Data Type: integer

Cell Size: 30 metre

Projection: mrworld

Datum: sphere

Map Extent: global

Description: Version 8.2 contains topographic and bathimetric data as developed by Smith and Sandwell (1997) who derived the topographic data from http://edcwww.cr.usgs.gov/landdaac/gtopo30/gtopo30.html. Chris Place, the Research Computing Officer for the Department of Geography, converted this file into a 50-km Arc/Info grid file with an Eckert VI projection.

Quality: Version 8.2 is considered more accurate than ETOPO-5 because it includes detail that ETOPO-5 fails to capture. However, in Version 8.2 the depths have limited accuracy and should not be used for navigation!

5.4 Sea Level Change

Source: http://geochange.er.usgs.gov/pub/sea_level/

Format: ASCII grid

Data Type: integer

Cell Size: 1-degree (360 columns, 180 rows)

Projection: Geographic (degree decimal)

Map Extent: global

Description: Land topography of the world for 9k, 13k, and 18k. Each grid cell gives an estimate of the percentage of the corresponding area of the earth's surface that was covered by land. Global topography is based on ETOPO-5. The estimate of past topography is based on ICE4G by Professor W.R. Peltier from the University of Toronto, Canada.

Quality: Since the data is based on work from Professor W.R. Peltier, world renowned for his work on paleotopography, it should be of good quality compared to other past topographic reconstructions. However, there are some unexplained anomalies in the dataset. Some grid cells have values over 100 and below 0, which does not make sense if each grid cell value is supposed to be an estimate of the percentage of land in that grid cell.

5.5 Mean Annual Surface Temperature

Source: http://ipcc-ddc.cru.uea.ac.uk/cru_data/datadownload/download_index.html

Format: ASCII grid

Data Type: integer

Units: degrees Celsius multiplied by 10

Cell Size: 0.5-degree (720 columns, 360 rows)

Projection: Geographic (degree decimal)

Map Extent: global

Description: Mean annual surface temperature for the world averaged from 1931 to 1961 compiled by Mitchell *et al* (2002).

Quality: This data provides a good general picture of global temperature, but should not be trusted to give an accurate temperature reading at a point in space.

5.6 Mean January and July Surface Temperature

Source: http://www.pik-potsdam.de/~cramer/climate.htm

Format: ASCII grid

Data Type: floating point

Units: degrees Celsius

Resolution: 30-degree; longitude latitude point data

Projection: Geographic (degree decimal)

Map Extent: global

Description: Mean monthly global surface temperatures averaged from 1931 to 1960 compiled by Leemans and Cramer (1991).

Quality: This data provides a good general picture of global temperature, but should not be trusted to give an accurate temperature reading at a point in space.

5.7 Mean Annual Surface Precipitation

Source: http://ipcc-ddc.cru.uea.ac.uk/cru_data/datadownload/download_index.html

Format: ASCII grid

Data Type: integer

Units: millimetres per day multiplied by 10

Cell Size: 0.5-degree (720 columns, 360 rows)

Projection: Geographic (degree decimal)

Map Extent: global

Description: Mean annual global surface rainfall averaged from 1931 to 1960 compiled by Leemans and Cramer (1991).

Quality: This data provides a good general picture of global precipitation, but should not be trusted to give an accurate precipitation reading at a point in space.

5.8 Mean January and July Surface Precipitation

Source: http://www.pik-potsdam.de/~cramer/climate.htm

Format: ASCII

Data Type: floating point

Units: millimetres

Resolution: 30-degree; longitude latitude point data

Projection: Geographic (degree decimal)

Map Extent: global

Description: Mean monthly global precipitation averaged from 1931 to 1960 compiled by Leemans and Cramer (1991).

Quality: This data provides a good general picture of global precipitation, but should not be trusted to give an accurate precipitation reading at a point in space.

NOTE: Climate data from 1930 to 1961 is suitable for this study because it is regarded as being reflective of the weather representative of this point in the earth's history (i.e. post dates the Little Ice Age and precedes the recent global warming trend). Furthermore, climate records from Africa are more complete during this time period than any other.

5.9 Snow Cover and Depth

Source: Data on snow cover and depth comes from an old USSR atlas found in Drummond Library, University of Edinburgh. (The Great Soviet Atlas of the World. Volume I. 1933. Moscow. 159pp).

Data Quality: Data quality is considered to be sufficient for the purposes of this study. Furthermore, the time period from which the data is collected coincides with the other climate data used in this study.

5.10 BIOME 6000

Source: http://www.bgc-jena.mpg.de/bgc_prentice/databases/

Format: MS Excel97; longitude latitude point data

Map Extent: global

Description: The Global Palaeovegetation Mapping (BIOME 6000) Project is a community-wide effort that began in 1994 to construct a database of georeferenced biome type data. There are 25 different biome types. Prentice *et al* (1992) defines most of the biomes, except for temperature conifer forest (TECO), open conifer woodland (OC), savannah (SAVA), temperature microphyllous shrubland (TEMS), and temperate xerophtic woods/shrub (TEXE). The BIOME 6000 data also includes biomes WAMX and HODE, for which there was no description or definition provided in the downloaded data or by Prentice *et al* (1992). These two biomes were interpreted to represent *warm mixed forest* and *hot desert* respectively.

Quality: The data quality is considered to be quite good because it uses an objective method for assigning palaeoecological data to biomes in comparison to past palaeoecological data compilations (see Section 4.2.1). Furthermore, the BIOME model is able to account for plants reacting to climate change as individual species because it defines biomes by the climatic tolerances of individual plant types, rather than the apparent climatic distributions of biomes as they exist today. This is possible because BIOME 6000 biomes are created as a collection of plant types depending on their environmental limits to physiological factors such as cold tolerance, chilling/ heat/ and moisture requirements (Prentice *et al*, 1992); (see Section 4.2.1). However, there are several problems with the BIOME data. The BIOME 6000 project is unable to rule out the affect of human impact on modern day modelled biomes. In areas where cultivation is most intense (e.g. eastern China and western Europre) some biomes have been misclassified as desert or steppe instead of forest (Prentice and Jolly, 2000). Furthermore, data quality may be less for areas that experience monsoon climates. The BIOME model has difficulty simulating the effect of monsoons on vegetation because the strength of monsoons are not static over time (Jolly *et al*, 1998).

6 SOFTWARE

This study used ArcInfo version 8.0.2 and ArcGIS for geographical information analysis. The ArcInfo modules used in this study are: Arc, GRID, and Tables. The ArcGIS modules used in this study are: ArcMap, ArcCatalogue, and ArcToolbox. The RBM was written in AML and run in ArcInfo. The SBM was written in JAVA and run in Kawa version 4.10a. The BM was created in run in ArcInfo without having to write a program defining its processes. MS Excel was used for basic statistical analysis.

7 MODELLING SPECIFICATIONS

Before the models could be built there were number of data processing issues that had to be resolved. The following section gives an account of these issues.

7.1 Choosing a Projection

Of the mapping projections available in ArcInfo (Table 1), an Eckert VI projection was chosen because it preserves the area of data over a global extent. Eckert VI is a pseudocylindrical equal area projection that is recommended for mapping data covering large continents or the entire global area (Bugayevskiy, L.M., and Snyder, 1995; Robinson *et al*, 1995). Green (2000) used a Robinson's projection, which is also appropriate for mapping data with a global extent. However, because this projection does not preserve area it was not used. It is important to preserve area for this project because some of the analyse will compare results from different geographic regions. The results need to be standardised for the comparisons to be meaningful.

7.2 Geographic Coverage of Data

All of the raster data downloaded from the Internet was geographically split 180° from the Prime Meridian. Consequently, the very eastern portion of Russian was divided. To keep the Asian continent complete, the split was moved 12° to the east so the divide would fall within the Bering Straight (Figure 1).

Figure 1. An example of how the raster data was oriented so the geographic split would run along the Bering Straight.

Table 1. A comparison of ArcInfo mapping projections relevant for this study (ESRI, 2000).

Projection	Shape	Direction	Distance	Best For
Albers Conic (secant)	Accurate along standard parallels and minimally distorted in region between and just beyond.	Local angles correct along standard parallels.	Best in middle latitudes.	Regions predominantly east west in extent and at middle latitudes. North south range should not exceed 30-35 degrees. Used for small regions or countries, but not continents.
Lamber Azimuthal, Zenithal	Minimal distortion from point, but increases beyond that.	True direction radiating from the central point.	True at centre.	Not for data greater than a hemisphere in extent, because the software cannot handle it.
Lambert Cylindrical (type 1) (tangent)	True along standard parallels of normal aspect. Severe distortion near poles.	Local angles correct along std parallels.	Scale true along equator, but severely distorted near poles.	Narrow areas along central line. Best for equatorial regions.
Eckert IV (pseudo cylindrical)	Stretched 40% in north south direction relative to east west along equator. No stretching at 40°30′ N and S. Features become compressed near poles.	Local angles correct along 40°30′ N and S.	Scale distorted 40% in north south direction relative to east west along equator. No distortion at 40°30′ N and S. Features become compressed near poles.	Only useful as a world map, e.g. climatic thematic maps.
Eckert VI (pseudo cylindrical)	Stretched 29% in north south direction relative to east west along equator. No stretching at 49°16′ N and S. Features become compressed near poles.	Local angles correct along 49°16′ N and S.	Scale distorted 29% in north south direction relative to east west along equator. No distortion at 49°16′ N and S. Features become compressed near poles.	Only useful as a world map, e.g. climatic thematic maps.
Moll Weide (pseudo cylindrical)	Not distorted at central meridian and latitudes 40°44′ N and S. Distortion increases beyond these points and becomes severe.	Local angles true at central meridian and latitudes 40°44′ N and S.	Scale true along latitudes 40°44′ N and S. Distortion increases beyond these points and becomes severe.	Only useful as a world map, e.g. climatic thematic maps.

7.3 Cell Size

All grid surfaces used in the model were set to have a 50 km grid (area per cell: 2,500,000,000 m^2). This is a good resolution for processing data over such a large geographic area. Higher resolutions would take too long to process, and lower resolutions would yield less accurate results. Furthermore, this resolution is appropriate for capturing rate of migration through time, assuming wildcats can displace approximately 55 km between generations.

8 DATA PROCESSING

The modelling specifications outlined in Section 7 were followed to create grids of the environmental variable data. The file names are included in brackets.

8.1 Specific Issues Regarding Data Processing

8.1.1 Annual Temperature and Precipitation Data

Upon projecting the mean annual temperature and mean annual precipitation grids to an Eckert VI projection, ArcInfo delivered a message that 'Inversibility was 90 percent'. This means that data quality has been slightly compromised. Green (2000) encounter this problem while processing his data, and found that removing the America's eliminated the problem, thus, achieving 100 percent inversibility. This was tried in current study but did not change the degree of inversibility. Since the data quality still appears to be adequate, in the interest of saving time, 90 percent inversibility is accepted (*dissertation/data/precipitation/cpre3160_p* and *dissertation/data/temperature/ctmp3160_p*); (see Appendices 16.2.1 to 16.2.2 for commands).

8.1.2 Monthly Temperature and Precipitation Data

The grids created for mean January/July temperature/precipitation had to be interpolated from the original data sets. The original datasets are organised as point data for longitude and latitude. The point data forms an evenly spaced grid, however, it does not fit the 50 km specification of this study. Since the data points are created by interpolating information received from climate stations, it was best to choose an interpolation technique that resembled the original data points as closely as possible. The *inverse distance weighting* ArcInfo function was found to interpolate most successfully (*dissertation/data/precipitation/pjanpr4n* and *pjulprp4n* and *dissertation/data/temperature/tjanp4n* and *tjulp4n*); (see Appendices 16.2.3 to 16.2.6 for commands).

8.1.3 Temperature Range Data

The temperature range grid is derived by subtracting the January mean from the July mean in the Northern Hemisphere and vice versa for the Southern Hemisphere. This approach will not derive the true temperature range in monsoon areas where heavy rain during summer months act to reduce temperatures, resulting in spring being the warmest season (Strahler and Strahler, 1997). However, for most places in the study area this approach is deemed an appropriate way of deriving the mean temperature range (*dissertation/data/temperature/tmprng*); (see Appendix One 16.2.7 for commands).

8.1.4 Continentality Data

A continentality grid is created using the topographic data. The first step is to create a land/sea mask where land gets a value of 1 and water is assigned as being *no data* (grid cell value = -9999). The land/sea mask is used to derive a continentality grid with values ranging from > 0 (maritime) to 1 (continental), and grid cells covering water still being assigned as *no data*. Continentality values are calculated in ArcInfo GRID using the *distance weighted focal sum* instead of the *focal sum* operation. *Focal sum* treats all cells equalling within a search radius when calculating the value of a single cell, whereas the *distance weighted focal sum* enables nearer cells to have a greater influence than distance cells. Since the effect of water bodies on land climate is thought to decrease exponentially as distance from the coast increases, the *distance weighted focal sum* seems more appropriate. The exponential decrease was calculated with the function $y = x^{1/3}$[2]. The search neighbourhood used to process each cell value is a grid of 11 x 11 cells, which is equivalent to a grid of 550 km x 550 km, or a square search 'radius' ranging from 275 to approximately 390 km (Figure 2). This distance is sufficient because land further than 275 to 390 km from water likely does not experience a maritime effect (*dissertation/data/topography/cont_landn*); (see Appendix One 16.2.8 for commands).

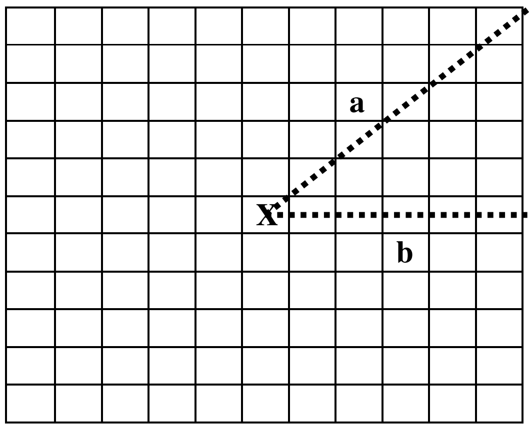


Figure 2. The size of the search neighbourhood of grid cells used to calculate the continentality value for a grid cell X. The search 'radius' ranges from 275 km (segment b) to approximately 390 km (segment a).

[2] The literature was not consulted to arrive at this value, rather, this function simply seemed like a reasonable approximation to how the effect of water bodies on land decreases with distance.

9 CREATING THE BIOME MODEL

The BM reconstructs the wildcat distribution from the correlation between the known wildcat sightings data (from Nowell and Jackson (1996)) and biome types from the BIOME 6000 Project. Data from both sources is sparse. The wildcat sightings data is only a general reconstruction of wildcat presence and absence. The BIOME 6000 data is not extensive or evenly distributed over Europe, Africa, and Asia. There is a much greater coverage of data in Europe than Africa. When finding the correlation between the two datasets, rankings of biome types should not be based on the frequency of occurrence in areas where wildcats are found or biomes in data poor areas such as Africa will be underrepresented. To overcome this, statistical analysis was performed on confusion matrix data created from the results of intersecting the two datasets, to identify biomes that are meaningful predictors of wildcat presence and absence.

A confusion matrix is calculated as a sum of the true positives (wildcats are found within biome x), true negatives (no wildcats are found outside of biome x), false positives (wildcats are found outside of biome x), and false negatives (no wildcats are found in biome x). To derive counts for the confusion matrices, the BIOME 6000 point coverages and the Nowell and Jackson (1996) polygon coverage were converted into grids. By assigning the Nowell and Jackson (1996) grid to have values of 1000 for wildcat presence and 100 for wildcat absence, subtracting the BIOME 6000 grid results in unique presence and absence values for every biome type. These values are used to form the confusion matrices. Kappa (K) statistical analysis is performed on the confusion matrices data to determine how well a biome type predicts wildcat presence. $K < 0.4$ are poor predictors, $0.4 < K < 0.75$ are good predictors and $K > 0.75$ are excellent predictors (Fielding and Bell, 1997). The biomes were organised into four groups by putting together biomes with *like* K values. The biome groupings were then ranked 0, 1, 2, or 3 for wildcat preference, such that the lowest K value group received 0 and the highest K group received 3. The probability of wildcat presence assigned to each biome type was calculated as the number of sightings within a biome divided by the total number of occurrences of that biome. The final step was to reclassify the BIOME 6000 0k, 9k, and 18k grids to have the calculated wildcat preference values in place of the corresponding biome types (see Appendix One 16.3 for commands).

A similar process was performed with the Green (2000) sightings data to evaluate how well the analysis between the Nowell and Jackson (1996) and BIOME data was at defining wildcat biome preferences. Since the data from Green (2000) is a polygon coverage, a point-in-polygon intersect operation was used to identify all the biomes coinciding with the sightings data.

9.1 Quality of the Biome Model

The low Kappa values suggest that no biomes are meaningful predictors of wildcat presence because all values are < 0.40. However, by grouping the biomes by *like* Kappa values to identify biomes that predict wildcat presence/absence, the results do seem to coincide well with current knowledge of wildcat habitat preferences. Furthermore, when a similar analysis was done comparing the biome types to the modern day wildcat sightings coverage developed by Green (2000), it was found that there was a good overall agreement between the two wildcat sightings datasets and the biome data (Table 2).

Table 2. Correspondence in the biome types selected for the Biome Model as based on the Nowell and Jackson (1996) wildcat sightings data and the biome types coinciding with the Green (2000) wildcat sightings data. *Biome types not used in creating the Biome Model because they have fewer then 25 records, therefore, were not considered able to produce meaningful presence of Kappa results.

Biome Type	Biomes coinciding with sightings data from Nowell and Jackson (1996)	Biomes coinciding with sightings data from Green (2000)
xerophytic woods/scrub	x	
savannah	x	x
steppe	x	x
warm mixed forest	x	x
desert	x	x
broadleaved evergreen/warm mixed forest	x	x
tropical dry forest/savannah *	x	x
temperate xerophytic woods/shrubs *	x	x
cold mixed forest	x	x
temperate microphyllous shrubland *	x	
tropical rain forest *	x	
warm grass/shrub *	x	x
tropical seasonal forest	x	
tropical xerophytic bush/savannah *	x	
hot desert *	x	
tundra	x	x
cool conifer forest	x	
cold deciduous forest	x	
temperate deciduous forest	x	x
cool mixed forest	x	x
taiga	x	x

10 CREATING THE SECONDARY BIOME MODEL

The wildcat distribution created by the BM does not cover the entire study area. Distribution data is only present where the data points from the BIOME 6000 dataset exist. The SBM was created to derive a wildcat distribution that completely covers the study area. The aim was to create an outcome that provides a more meaningful result with which to evaluate the 0k RBM outcome. To create the SBM, correlations between key environmental variables and the biome types are determined. Once the correlations are known, the biome type for each 50 km grid cell are derived as a function of the environmental variables. The biome types are then ranked relative to wildcat preference in the same manner described for the BM in Section 9. Producing a grid of the ranked habitats reconstructs the present day wildcat distribution.

Basic statistical calculations were performed in MS Excel to learn how the environmental data relates to the biome types. Minimum, maximum, mean, and standard deviation for each biome type relative to the eight environmental variables were calculated (see Appendix Two for results). Mean annual surface temperature (ctmp), mean annual surface precipitation (cpre), and temperature range (trange) were chosen as the most important variables for deriving biome type because these variables can act as proxies for the other environmental variables. For example, temperature data can act as a proxy for elevation and temperature range. A number of statistical packages were investigated (MS Excel, MiniTab, and S-PLUS) to find a statistical technique, such as cluster analysis or nearest neighbour, that could assign a biome type to a location as a function of the environmental variable data for that location. Since no statistical technique was found, a program was written in JAVA to perform the analysis.

10.1 Processing the Data for the Secondary Biome Model

To reduce the complexity of the program the input data[3] (environmental variable and BIOME 6000 data) was processed to fulfil a set standard. The environmental variable grids were clipped to have the same geographical extents and modified to have the same number of *no data* (-9999) values occurring in the same locations. *No data* values occurring in each grid were accumulated to produce an overall grid of *no data* values, which was then used to set the *no data* values of each environmental variable grid. This was achieved in ArcInfo GRID through a series of grid multiplications (see Appendix 0ne 16.4.1 for commands). The final stage of formatting

---

[3] Even though only three environmental variables are used as input data, all the environmental variable data was processed for use in the SBM in case this data was ever required at a later time.

the environmental variable data was to export the grids as ASCII files and save them with a 'txt' extension in TextPad. The ASCII files contain standard header information for a raster that was subsequently used by the JAVA program. Example of header information:

```
ncols         454
nrows         327
xllcorner     -8750000
yllcorner     -8150000
cellsize      50000
NODATA_value  -9999
```

These grids were saved in the directory *dissertation/statistics* under the names: cpre, ctmp, cont, dem, pjan, pjul, tjan, tjul, and trange.

As well as having environmental variable data that completely covers the study area, the SBM also requires environmental variable data for each BIOME 6000 biome location. To achieve this, point coverages for each environmental variable were created that correspond with the biome locations. This was done by intersecting a polygon coverage of each environmental variable with a point coverage of the biome data. The environmental variable attribute data for each resulting point coverage (biome type, longitude, latitude, environmental variable data) were then exported as CSV files and imported into MS Excel. Within MS Excel the data was sorted by biome type in ascending order, and then by location. It is important to ensure that the values of all data used in the SBM are in the same geographic order. After the data was sorted all information was deleted except for the environmental variable data. The data was then exported as text files saved in the directory *dissertation/statistics* under the names: biome_cont.txt, biome_cpre.txt, biome_ctmp.txt, biome_dem.txt, biome_pjan.txt, biome_pjul.txt, biome_tjan.txt, biome_tjul.txt, and biome_trange.txt (see Appendix One 16.4 for commands).

10.2 Statistical Functionality of the Secondary Biome Model

A nearest neighbour statistical approach was used to predict the biome types for each 50 km grid cell in the study area. All of the biome types were used in the model except for *open conifer woodland* and *cool grass/shrub*. The former being discarded because it has no data records and the latter being discarded because it has only one data record. The first step was to calculate a difference statistic for every environmental variable relative to each biome type using the equation:

$$\mathbf{DF = [((biomeX_{variableY} - test\ point_{variableY})/stdev_{biomeX,\ variableY})^2] / n}$$

Every value in the environmental variable study area grids was tested. Each value (test point $_{variableY}$) was cycled through the environmental variable data that geographically coinciding with the biome locations (biomeX $_{\text{variableY}}$). The difference between the environmental variable Y occurring at in biome type X (biomeX $_{\text{variableY}}$) and the test point was calculated. The standard deviation[4] for all the values of environmental variable Y coinciding with the biome locations was also calculated for each biome type X (stdev $_{\text{biomeX, variableY}}$). A *difference statistic* (DF) was calculated by dividing the calculated difference by the standard deviation and squaring the product to produce an absolute value. The DF's were summed for each biome and then divided by the number of records in the biome (n) to standardise the values to each other. The standardised DF sums were then summed for each environmental variable for the respective biome to produce an *overall difference statistic* for each biome type. The biome type with the lowest *overall difference statistic* value was designated as the biome type occurring at the same location as the corresponding test point (Table 3); (see Appendix Three for JAVA code).

Table 3. An example of processing the data to determine the biome type for each test point. One test point at a time is cycled through all the calculations to arrive at the overall difference statistics. The biome type with the lowest overall difference statistics is the biome type assigned to the test point. For example, if R<Q<S than the biome type at the geographical location coinciding with test point i would be biome type X+1. DF = difference statistic.

Biome Type	Environmental Variable Y	Environmental Variable Y+1	Environmental Variable Y+1	Overall Difference Statistic
X	DF calculation	DF calculation	DF calculation	
X	DF calculation	DF calculation	DF calculation	for Biome X:
X	DF calculation	DF calculation	DF calculation	Q = (a + b + c)/4
X	DF calculation	DF calculation	DF calculation	
DF Sum:	a	b	c	
X+1	DF calculation	DF calculation	DF calculation	
X+1	DF calculation	DF calculation	DF calculation	for Biome X+1:
X+1	DF calculation	DF calculation	DF calculation	R = (d + e + f)/5
X+1	DF calculation	DF calculation	DF calculation	
X+1	DF calculation	DF calculation	DF calculation	
DF sum:	d	e	f	
X+2	DF calculation	DF calculation	DF calculation	
X+2	DF calculation	DF calculation	DF calculation	for Biome X+2:
X+2	DF calculation	DF calculation	DF calculation	S = (g + h + i)/3
DF sum:	g	h	i	

[4] Standard deviation is calculated using every value (x) in an entire population (n) using the equation:

$$\text{standard deviation} = \sqrt{\frac{n\sum x^2 - \left(\sum x\right)^2}{n^2}}$$

Running the JAVA program produced a grid of the study area where every 50 km grid cell has a biome value. Different weightings of the environmental variables were tried to find the optimal reconstruction relative to known global biotic regimes. It was found that dividing the mean annual surface temperature by three, places more importance on temperature as a determinate of biome type and that this produced the most ideal outcome. The optimal SBM grid is saved as: *dissertation/statistics/biomesw3*. The same procedure as that described for the BM in Section 9, was used to complete the SBM. That is, the Nowell and Jackson (1996) wildcat data was intersected with the SBM biome grid and Kappa analysis was performed on the results of the intersection to derive wildcat preference rankings. The rankings were then assigned to the biome types. Mapping the ranked biome types produces the modern day distribution.

10.3 Quality of the Secondary Biome Model

The quality of the SBM is considered to be surprisingly good for predicting biome type as a factor of only three environmental variables. This is supported by the fact that there is a high level of correspondence between the biomes preferred by wildcats as selected by the SBM using sightings data from Nowell and Jackson (1996) and from Green (2000); (Table 4).

Table 4. Correspondence in the biome types selected for the Secondary Biome Model as based on the Nowell and Jackson (1996) wildcat sightings data and the biome types coinciding with the Green (2000) wildcat sightings data.

Biome Type	Biomes coinciding with sightings data from Nowell and Jackson (1996)	Biomes coinciding with sightings data from Green (2000)
tropical dry forest/savannah	x	x
savannah	x	x
tropical xerophytic bush/savannah	x	x
temperate xerophytic woods/shrubs	x	x
cold mixed forest	x	x
temperate microphyllous shrubland	x	x
cool mixed forest	x	x
temperate deciduous forest	x	x
tropical seasonal forest	x	x
tropical rain forest	x	x
xerophytic woods/scrub	x	x
steppe	x	x
desert	x	x
warm mixed forest	x	x
warm grass/shrub	x	x
tundra	x	x
taiga	x	
cold deciduous forest	x	
cool conifer forest	x	x
broadleaved evergreen/warm mixed forest	x	x
hot desert	x	x

Despite good correspondence with the sightings data, uncertainty in the SBM results may exist for a several reasons:

- The quality of the monthly environmental data used to create *trange* may differ from the quality of the annual environmental data of *ctmp* and *cpre*. This is likely because the monthly and annual data come from two separate organisations. Scepticism over the correspondence in data quality arises because of odd associations between the two data sources. For example, in some areas of the Sahara Desert the mean annual precipitation is zero, however, the monthly mean precipitation for January and/or July is above zero.
- Odd associations also exist between the BIOME 6000 data and the environmental variable data (see Appendix Two for details of values).
    - Annual precipitation values for desert are higher than expected, and are also greater than savannah and steppe, which are technically wetter biomes.
    - Tropical rain forest biomes have very low mean annual precipitation values.

11 CREATING THE RULE BASED MODEL

RBM is created by forming environmental regions (e.g. savannah, tundra) and then ranking these regions by known wildcat preferences. In general, wildcats prefer areas with a good source of small animals, such as grasslands, temperate woodlands, and dry woodlands of Africa. They avoid areas with wet hot forest, really dry desert, cold snowy forest, and high exposed mountains. The environmental regions are formulated as a function of some or all of the nine environmental variables. This was done two ways:

- The geographic area of an environmental region is queried using the ArcMap 'identity' tool to define the range of each environmental variable[5].
- Information from literature/atlases/expert knowledge is used to define the environmental variable ranges of each environmental region.

An AML was written to create grids of each environmental region as defined by the nine environmental variables. For environmental regions that are only specific to certain regions (e.g. wildcat elevation preferences for Europe) a mask was created for that region so the rules could be restricted to it. Within the AML, the environmental region grids are ranked relative to wildcat habitat preference. The ranking follows the same four-point system (i.e. 0 to 3) as used in the BM and RBM. The grids are then merged to produce a final grid of habitat preferences. Since the environmental regions are not all geographically exclusive, some grids have to take precedence over the others. Grids with a wildcat preference of zero, (e.g. elevation greater than 3000 in Europe) are given the highest priority so these areas are immediately excluded from the reconstructed distribution. The remaining grids are ordered so the output grid best matches the modern day Koppen Climate System. Clearly, depending on how the grids are ranked influences the final wildcat preference output. This is not a severe issue when creating the modern day climate coverage, because it is easy to re-order the grids to match the Koppen Climate System. However, when reconstructing past climates, it is not known whether one climate type should take priority over another when an overlap occurs. Consequently, the grid orders were not changed for past reconstructions.

---

[5] An alternative approach not employed by this study would be to digitise a map of the Koppen climates and isolate the environmental variables values for each climate region. Statistical procedures could then be used to determine the ranges of the environmental variables within each environmental region. However, this would be a time consuming process, and the accuracy of the resultant digitised map, as well as the source Koppen climate map, may not be high enough to make this approach worthwhile.

Two versions of wildcat habitat preference rankings were compared and contrasted to create an optimal output of the RBM. One version is based primarily on information from Kitchener (2002) and the other version is derived predominately from Nowell and Jackson (1996). The RBM outcomes of the two versions were assessed visually for correspondence with the known wildcat sightings data. Furthermore, the difference in wildcat preference was calculated between the two outcome grids to identify areas were preference differed. The wildcat preference rules were adjusted to maximise the coverage of high preference rankings corresponding with known wildcat sightings data. The resulting wildcat habitat preference rankings were used to create an optimal RBM that best reconstructs the 0k distribution. This version of the RBM was used for subsequent experiments and for reconstructing past wildcat distributions. The optimal RBM 0k grid outcome is saved as: *dissertation/aml/cat0k* (see Appendix Four for the AML producing the 0k outcome).

12 REASONS FOR USING MULTIPLE MODELS

This study tests the ability of the RBM to reconstruct the evolutionary history of the wildcats. The BM and SBM are being used to help evaluate the results produced by the RBM. The BM and SBM can tackle evolutionary history reconstruction from a different perspective than the RBM because the BM and SBM BIOME 6000 data develops from *vegetative* point of view, while RBM Koppen Climate Scheme data is based on *climatic* differences. The BM is particularly useful because it is created from completely unrelated data than the RBM. Factors in the RBM data that may increase the uncertainty in accuracy of the results will not affect the BM. Furthermore, the BM is the only means with which to evaluate the past RBM wildcat distributions. Past BM outcomes should be a good means to measure the accuracy of the RBM outcomes because of the high quality of the BIOME 6000 data. The SBM uses unrelated known sightings and BIOME 6000 data, but also uses the same environmental variable data as the RBM. The SBM is not considered as objective as the BM for assessing the RBM, however, the SBM does have an advantage over the BM. The SBM can reconstruct the wildcat distribution for every grid cell in the study area, unlike the BM. Therefore, the SBM provides more data with which to evaluate the RBM

13 ASSESSING THE RULE BASED MODEL

The 0k RBM outcome was assessed to establish the level of confidence that could be afforded to the present and past reconstructions. This was done by comparing the 0k RBM outcome to the known wildcat sightings data, and to the 0k outcomes of the BM and SBM. Resulting files are indicated in brackets.

13.1 <u>0k RBM Outcome versus the Known Wildcat Sightings Data</u>

13.1.1 Visual Comparison

The known wildcat sightings data was laid over the 0k RBM distribution to visually investigate the level of correspondence between the two. The known leopard cat distribution was also overlaid to see how its location corresponded with the RBM predicted distribution and known wildcat sightings data.

13.1.2 Intersecting

The GIS analytical *intersect* function was used between the RBM 0k outcome and the known wildcat sightings data to quantify their level of agreement. The Nowell and Jackson (1996) and Green (2000) sightings data exist as a polygon coverages (*dissertation/data/wildcat_sightings/nj_cats; /presence_p*). To make a comparison with the 0k RBM grid data, these polygon coverages were converted into grids where values of 1 represent wildcat presence and values of *no data* represent wildcat absence (*dissertation/evaluation/nj_cats; dissertation/evaluation/sg*). Each sightings grid was multiplied separately with the 0k RBM grid to derive grids giving a count of cells where RBM wildcat preference values coincide with the respective sightings data (*dissertation/evaluation/0k_nj; dissertation/evaluation/0k_sg*). This information was used to calculate the percent correspondence between the predicted distribution and the sightings data. A similar procedure was used to calculate the percent correspondence between the RBM 0k outcome and the sightings data from the Harrison and Bates (1991), Kitchener (2002), and Zhang Yongzu *et al* (1997); (*dissertation/data/wildcat_sightings/arabian_cats; /kitchener_p; /china*). However, because this sightings data exists as point coverages, the 0k RBM outcome had to be converted into a polygon coverage in order to make the comparisons (*dissertation/evaluation/njcats*). Intersecting the 0k RBM polygon coverage with the sightings point data yields a count of the number of sightings points falling within and outside of the predicted wildcat preference zones (*dissertation/evaluation/evalarabia; /evalkitch; /evalchina*).

13.2 RBM 0k Outcome versus the BM and SBM 0k Outcomes

The second approach for evaluating the 0k RBM outcome was to compare it with the 0k outcomes produced by the BM and the SBM. This was done by subtracting the BM and SBM grids, respectively, from the RBM grid (RBM - BM: *dissertation/evaluation/bio0rcl; /bio9rcl; /bio18rcl;* RBM - SBM: *dissertation/evaluation/sbmrcl*). Since all models use a four point wildcat preference ranking system, the values in the resulting difference grids quantify the level of correspondence between the reconstructions. Negative values are cells where the RBM has a lower preference. Positive values are cells where the RBM has a higher preference. Zero values occur where there is no difference in the wildcat preference.

14 EXPERIMENTING WITH THE RULE BASED MODEL

14.1 Simulating Climate Change with Temperature

Over the past 20,000 years the mean global temperature has ranged from being about 10°C cooler to approximately 3°C warmer than present day temperatures (Williams *et al*, 1998). The simplest way to simulate a change in global temperature is to increase or decrease the temperature grids by a set amount. However, this is not realistic because shifts in temperature would have been more extreme at the poles than at the equator. A straightforward way to simulate this is to steadily increase the temperature change as distance from the equator increases. For example, a global temperature drop of 6°C can be expressed as a temperature decrease from 4°C at the equator to 10°C at the poles using the formula:

$$((1 + ((\text{latitude} - 90°) / 90°)) \times 6) + 4 \qquad \text{Equation 1}$$

where *latitude* is a grid of latitude values in degrees.

The present study compared both approaches. Subtracting 10°C from the temperature grids used in creating the 0k RBM outcome simulated a uniformly global drop in temperature (*dissertation/aml/catprefl1b;* run with AML: *dissertation/aml/cat6.aml*). A non-uniform mean global temperature drop of 10°C was accomplished by using Equation 1 to create temperature change grids that were then subtracted from the corresponding 0k RBM temperature grids. Re-running the model produces a grid with a non-uniform temperature drop (*dissertation/aml/catprefl3b;* run with AML: *dissertation/aml/cat7.aml*). The resulting uniform and non-uniform temperature change outcomes are most different from each other along the tropics, such that, a uniform temperature change causes a much greater change in the wildcat distribution than a non-uniform temperature change. Since the tropics were more climatically stable through time than the areas near the poles (Williams *et al*, 1998), this study will simulate climate change with a non-uniform change in temperature (see Appendix One 16.5.1 for commands).

14.2 Creating 9k, 13k, and 18k Rule Based Model Wildcat Distributions

The RBM was used to reconstruct the wildcat distribution at the last glacial (18k) and interglacial (9k) maxima, as well as the approximate midpoint between the two time periods (13k). Altering the temperature and continentality environmental variables simulates climate change. Temperature change is expressed as a non-uniform shift in global temperature (see Section 14.1 for details). New continentality grids are derived from the 9k, 13k, and 18k sea level data. The original sea level grid cell values are expressed as the percent of land covering the grid cells. To incorporate this data into the model, the values are changed so that all grid cells having ≥ 50% land coverage are given a value of 1, and all grid cells with < 50% land coverage are assigned a value of *no data*. The result creates land/sea masks for 9k, 13k, and 18k from which continentality is derived as discussed in Section 8.1.4. An AML for each time period is created as a copy of the 0k RBM AML but that uses temperature and continentality environmental variable grids for the respective time period. To show where sea level differs from the modern day and the 9k, 13k, and 18k outcomes, the land/sea masks for the respective time periods are included in the final merging of the environmental region grids. The land/sea masks are given the lowest priority in the merging process so to not overlap any of the predicted wildcat preference data (see Appendix One 16.5.2.1 and 16.5.2.2 for commands and Appendix Four for a copy of the AMLs producing the 9k, 13k, and 18k outcomes).

14.2.1 Assessing the 9k and 18k Rule Based Model Outcomes

The 9k and 18k RBM outcomes are assessed by comparing them to the 9k and 18k BM outcomes. The 13k RBM outcome cannot be assessed because there is no 13k BIOME 6000 data from which to create a 13k BM outcome. The procedure used to make the comparisons is identical to that used for comparing the 0k RBM and 0k BM and SBM outcomes (see Section 13.2 for details and Appendix One 16.5.2.3 for commands).

14.2.2 Investigating the Changing Distribution over the Late Quaternary

The 0k, 9k, 13k, and 18k RBM outcomes were analysed to assess the level of change in the wildcat distribution over the Late Quaternary. Visual comparisons were made between the time periods and a difference grid was also calculated between distribution extremes (i.e. 9k grid - 18k grid) as preliminary measures for detecting differences in the geographic locations of wildcat preferences (difference grid between 9k and 18k saved as: *dissertation/aml/diff189*).

A more meaningful approach was to calculate the percent coverage of each wildcat preference rank per continent, to investigate how the area of habitat suitability altered with time. Multiplying a continent *mask* grid (e.g. Europe) of values 1 for the continent and *no data* for the remaining study area by the RBM outcome (0k, 9k, 13k, and 18k) results in a count of the wildcat preference cells for the respective continent (see Appendix One 16.5.2.4 for commands). The percent coverage of habitat suitability for the entire area of is also calculated to determine how the area of the entire distribution has altered over time.

The major problem with this type of analysis is the RBM cannot predict wildcat preference for past distributions in areas of land occurring outside the modern day sea level. As a consequence, analysis for each time period was restricted to the modern day land extent. This standardises the results to enable objective comparisons. However, without information about past wildcat preferences outside the modern land extent there cannot be a complete picture of how the area of wildcat habitat preference has changed through time.

15 REFERENCES AND BIBLIOGRAPHY

15.1 References

Bartholomew, J.C., Christie, J.H., Ewington, A., Geelan, P.J.M., Lewis OBE, H.A.G., Middleton, P., and Winkleman, B. (Editorial Directors). 1990. *The Times Atlas of the World.* London: Times Books. 225pp.

Balharry, D., Daniels, M., and Barratt, E.M. 1997. Wildcats: can genetics help their conservation? The role of genetics in conserving small populations. Edited by Tew, T.E., Crawford, T.J., Spencer, J.W., Stevens, D.P., Usher, M.B., and Warren, J. Peterborough: JNCC. 203 pp.

Bernhart, D.M. 1996. The occurrence of wildcats in the southern Swiss Jura Mountains. *Acta Therologica.* **41(2)**: 205-209.

Beamont, M., Barratt, E.M., Gottell, D. Kitchener, A.C., Daniels, M.J., Pritchards, J.K., and Bruford, M.W. 2001. Genetic diversity and introgression in the Scottish wildcat. *Molecular Ecology.* **10**: 319-336.

Begon, M. Harper, J.L., and Townsend, C.R. 1990. Ecology: Individuals, Populations and Communities. 2nd Edition. Boston: Blackwell Scientific Publications. 945pp.

Bugayevskiy, L.M., and Snyder, J.P. *Map Projections: A Reference Manual.* 1995. London: Taylor and Francis. 328 pp.

Corbet, G.B. 1978. The Mammals of the Palaearctic Region: a taxonomic review. British Museum (National History), Cornell University Press. 314pp.

Corbet, G.B. and Hill, J.E. 1992. *The Mammals of the Indomalayan Region: A Systematic Review.* Oxford: Oxford University Press. 488pp.

Daniels, M.J., Beaumont, M.A., Johnson, P.J., Balharry, D., MacDonald, D.W., and Barrat, E. 2001. Ecology and genetics of wild-living cats in the north-east of Scotland and the implications for the conservation of the wildcat. *Journal of Applied Ecology.* **38**: 146-161.

Environmental Systems Research Institute (ESRI) Inc. 2000. *Arc/Info Help, Version 8.0.2.* Redlands, USA.

Fielding, A.H. and Bell, J.F. 1997. A review of methods for the assessment of prediction errors in conservation presence/absence models. *Environmental Conservation.* **24(1)**: 38-49.

Freeman, S. and Herron, J.C. 2001. *Evolutionary Analysis.* 2nd edition, Prentice Hall, Upper Saddle River, NJ. 704 pp.

Futuyma, D.J. 1998. *Evolutionary Biology*. 3rd edition, Sinauer Association, Inc., Sunderland, Massachusetts. 763 pp.

Green, S.E. 2000. *Modelling the global distribution of the wildcat, Felis silvestris.* Unpublished M.Sc. Dissertation, University of Edinburgh.

Happold, D.C.D. 1987. The Mammals of Nigeria. Oxford: Clarendon Press. 402pp.

Harrison, D.L. and Bates, P.J.J. 1991. *The Mammals of Arabia, 2nd Edition*. Kent: Harrison Zoological Museum. 354pp.

Harvey, D. 1973. *Explanation in Geography*. London: Edward Arnold. 521pp.

Jolly, D., Prentice, C., Bonnefille, R., Ballouche, A., Bengo, M., Brenac, P., Buchet, G. Burney, D., Cazet, J.P., Cheddadi, R., Edorh, T., Elenga, H., Elmoutaki, S., Guiot, J., Laarif, F., Lamb, H., Lezine, A.M., Maley, J., Mbenza, M., Peyron, O., Reille, M., Reynaud-Farrera, I., Riollet, G., Ritchie, J.C., Roche, E., Scott, L., Ssemmanda, I., Straka, H., Umer, M., Van Campo, E., Vilimumbalo, S., Vincens, A., and Waller, M. 1998. Biome reconstruction from pollen and plant macrofossils data for Africa and the Arabian peninsula at 0 and 6000 years. *Journal of Biogeography*. **25**: 1007-1027.

Kitchener, A.C. 2002. Department of Zoology, National Museums of Scotland. Personal communication.

Leemans, R. and Cramer, W. 1991. The IIASA database for mean monthly values of temperature, precipitation and cloudiness of a global terrestrial grid. *International Institute for Applied Systems Analysis (IIASA)*. RR-91-18.

Nowell, K. and Jackson, P. 1996. *Status Survery and Conservation Action Plan: Wild Cats.* Gland, Switzerland: International Union for Conservation of Nature and natural Resources. 382pp.

Marsh, A. 2000. *Square One research*. Welsh School of Architecture. Bute Building, King Edward VII Ave CARDIFF, CF10 3NB Wales, United Kingdom. www.squ1.com

Mitchell-Jones, A.J., Amori, G., Bogdanowicz, W., Krystufek, B., Reijnders, P.J.H., Spitzenberger, F., Stubbe, M., Thissen, J.B.M., Vohralik, V., and Zima, J. 1999. *The Atlas of European Mammals*. London: T & AD Poyser Ltd.

Mitchell, T.D., Hulme, M., and New, M., 2002: Climate data for political areas. *Area* **34**:109-112.

Prentice, I.C., W. Cramer, S.P. Harrison, R. Leemans, R.A. Monserud and A.M. Solomon (1992). A global biome model based on plant physiology and dominance, soil properties and climate. *Journal of Biogeography* **19**, 117-134.

Prentice, C. and Jolly, D. 2000. Mid-Holocene and glacial-maximum vegetation geography of the northern continents and Africa. Ms. for *Journal of Biogeography*, BIOME 6000 special issue. http://www.bgc-jena.mpg.de/bgc_prentice/projects/biome6000/index.html.

Prentice, I.C. and T. Webb III (1998). BIOME 6000: reconstructing global mid-Holocene vegetation patterns from palaeoecological records. *Journal of Biogeography*, **25**: 997-1005.

Roberts, T.J. 1997. *The Mammals of Pakistan*. Oxford: Oxford University Press. 525pp.

Robinson, A.H., Morrison, J.L., Muehrcke, P.C., Kimerling, A.J., and Guptill, S.C. *Elements of Cartography, Sixth* Edition. New York: John Wiley & Sons, Inc. 1995.

Skinner, J.D. and Smithers, R.H.N. 1990. The Mammals of the Southern African Subregion. Pretoria, Transvaal, Republic of South Africa: University of Pretoria. 769pp.

Smith, W. H. F. and D. T. Sandwell. 26 September, 1997. Global Seafloor Topography from Satellite Altimetry and Ship Depth Soundings. *Science*. 277: 1956-1962.

Strahler, Alan and Strahler, Arthur. 1997. *Physical Geography: Science and Systems of the Human Environment*. New York: John Wiley & Sons, Inc. 637pp.

Williams, M., Dunkerley, D., De Deckker, P., Kershaw, P., and Chappell J. 1998. *Quaternary Environments, 2nd Edition*. London: Arnold. 329pp.

Zhang Yongzu, Jin Sangke, Quan Guogiang, Li Shihau, Ye Zhongyao, Wang Fenggui, and Zhang Manli. 1997. *Distribution of Mammalian Species in China*. Beijing: China Forestry Publishing House.7. 280pp.

15.2 Bibliography

Lockwood, J. G. 1974. *World Climatology: An environmental approach*. London: Edward Arnold Ltd. 330 pp.

Martyn, D. 1992. *Climates of the World*. Warszawa: Polish Scientific Publishers. 435 pp.

Pocock, R. I. 1951. *Catalogue of the genus Felis*. London: British Museum (Natural History).

Riley, D. and Spolten, L. 1974. *World weather and climate*. Cambridge: Cambridge University Press. 120 pp.

16 APPENDIX ONE

16.1 Koppen Climate Scheme Classification Rules

Table 5. Climate classification rules defined by Strahler and Strahler (1997) and Marsh (2000). Data from the latter source is enclosed in brackets.

Koppen Type:	Koppen Sub-Type:	
A tropical rainy -all months avg. temp >18°C -no winter -large annual precip and exceeds evap. -(annual precip > 1500 mm)	**Af tropical rainforest** -precip in driest month ≥ 6 cm -(precip. all year long) -(monthly temp. varies < 3°C) -(daily highs ~32°C and night time avg ~22°C)	
	Am monsoon variety of Af -precip in driest month is < 6 cm -(very little rain during dry season) -(annual rainfall is ≥ Af, but falls in 7 to 9 hottest months) -(annual precip exceeds 1250 mm)	
	Aw tropical savanna -at least one month with precip < 6 cm -(extended dry season during winter) -(precip during wet season usually < 1000 ml, and only during the summer)	
B dry climates -on avg. precip < evap throughout year -(receives < 860 mm of annual precip) -(max summer temps 40-50°C) -(min night time temps 15-25°C)	**BS steppe climate** -semiarid, grasslands -zone between BW and A,C, and D -(receives more precip than BW)	**BSh dry-hot** -mean annual temp > 18°C
		BSk dry-cold -mean annual temp < 18°C
	BW desert climate -arid climate -annual precip usually < 40 cm	**BWh dry-hot** - mean annual temp > 18°C
		BWk dry-cold -mean annual temp < 18°C
C mild, humid -have a summer and winter -coldest month has avg temp < 18°C but above -3°C -at least one month has an avg temp > 10°C -(warmest month > 10°C) -(max summer temps 33-37°C) -(summer nights 18-20°C)	**Cf no dry season** -precip in driest month avg > 3 cm	**Cfa hot summer** - warmest month >22°C
		Cfb warm summer - warmest month <22°C
		Cfc cool, short summer - fewer than 4 months > 10°C
	Cw dry winter -wettest month of summer has 10x more precip than driest month of winter -alternatively, ≥70% mean annual precip falls in warmer 6 months	**Cwa hot summer** -warmest month >22°C
		Cwb warm summer -warmest month <22°C
		Cwc cool, short summer -fewer than 4 months > 10°C
	Cs dry summer -precip of driest month of summer < 3cm -total precip at least 3x	**Csa hot summer** -warmest month >22°C
		Csb warm summer - warmest month <22°C

	as much as driest summer month -alternatively, ≥70% mean annual precip falls in winter 6 months	**Csc cool, short summer** - fewer than 4 months > 10°C
D snowy-forest -coldest month has avg < -3 °C -warmest month has avg >10°C -forest not normally found where warmest month is below 10°C	**Df moist winter** -no dry season	**Dfa hot summer** - warmest month >22°C
		Dfb warm summer - warmest month <22°C
		Dfc cool, short summer - fewer than 4 months > 10°C
		Dfd cold winter -coldest month < -38°C
	Dw dry winter -	**Dwa hot summer** - warmest month >22°C
		Dwb warm summer - warmest month <22°C
		Dwc cool, short summer - fewer than 4 months > 10°C
		Dwd cold winter - coldest month < -38°C
E polar climates -warmest month avg < 10°C	**ET tundra** -warmest month avg > 0°C and < 10 °C	
	EF perpetual frost -ice sheet climate -all months avg < 0°C	
H highland climates	-cool to cold, moist climates that occupy mountains and high plateaus -similar in seasonality to surrounding lowland climates	

16.2 <u>Data Processing</u>

16.2.1 Mean Annual Precipitation Data

(source data: dissertation/data/precipitation/cpre3160.dat)

TextPad Commands:

- In TextPad and header information was added.

`ncols 720`
`nrows 360`
`xllcenter -167.75`(0.25° offset because each coordinate represents the centre of a grid cell)
`yllcenter -89.75`
`cellsize 0.50`
`nodata_value -9999`

- The first 1920 columns are cut and pasted to be the last half of the data. This is done to split the data down the Bering Straight and not down the Greenwich Meridian. (Working: 180° + 12° = 192° x 2 columns/° = 384 columns x 5 = 1920 columns; 180° to split the world in half + 12° to push the divide into the Bering Straight waters; every two columns of data represents one degree; multiply by 5 to accord to the TextPad column numbering). The Block Select Mode is used from the configure menu (This allows columns to be highlighted instead of rows).
- Data is saved as 'cpre6190_fit'

ArcInfo Commands:

- 'cpre3160_fit' imported in Arc:

```
Arc: asciigrid cpre3160_fit cpre3160 int
```
- Projecting ‘cpre3160’ in Arc to an Eckert VI projection:
- Defining projection for ‘cpre3160’ data:

```
Arc: projectdefine grid cpre3160
Define Projection
Project: projection geographic
Project: units dd
Project: parameters
```
- To project ‘cpre3160’ data:

```
Arc: project grid cpre3160 cpre3160_p # cubic 50000 0 0
Project: output
Project: projection eckertvi
Project: units meters
Project: parameters
Longitude of projection center            [   0  0  0.000 ]: 50 0
0.000
Project: end
```
ArcInfo Message: Inversibility is 90 per cent

16.2.2 Mean Annual Surface Temperature

(source data: dissertation/data/temperature/ctmp3160.dat)

TextPad Commands:

- In TextPad and header information was added.

```
ncols 720
nrows 360
```
`xllcenter -167.75`(0.25° offset because each coordinate represents the centre of a grid cell)

```
yllcenter -89.75
cellsize 0.50
nodata_value -9999
```
- The first 1920 columns are cut and pasted to be the last half of the data. This is done to split the data down the Bering Straight and not down the Greenwich Meridian. (Working: 180° + 12° = 192° x 2 columns/° = 384 columns x 5 = 1920 columns; 180° to split the world in half + 12° to push the divide into the Bering Straight waters; every two columns of data represents one degree; multiply by 5 to accord to the TextPad column numbering). The Block Select Mode is used from the configure menu (This allows columns to be highlighted instead of rows).
- Data is saved as ‘ctmp3160_fit’

ArcInfo Commands:

- ‘ctmp3160_fit’ imported in Arc:

```
Arc: asciigrid ctmp3160_fit ctmp3160 int
```
- Projecting ‘ctmp3160’ in Arc to an Eckert VI projection:
- Defining projection for ‘ctmp3160’ data:

```
Arc: projectdefine grid ctmp3160
Define Projection
Project: projection geographic
Project: units dd
Project: parameters
```
- To project ‘ctmp3160’ data:

```
Arc: project grid ctmp3160 ctmp3160_p # cubic 50000 0 0
Project: output
```

```
Project: projection eckertvi
Project: units meters
Project: parameters
Longitude of projection center          [   0  0  0.000 ]: 50 0
0.000
Project: end
```

ArcInfo Message: `Inversibility is 90 per cent`

16.2.3 Mean January Precipitation

(source data: dissertation/data/precipitation/prec.grd.gz)

UNIX Commands:

- Unzipped the file.

```
bash-2.02$ gunzip prec.grd.gz
```

- File now saved as: prec.grd

Excel Commands:

- Excel is used to add the header information (x, y, jan) and change the file from being space delimited to tab delimited.
- At line 55739, data had to be shifted over one column to correspond to the preceding data.
- File was saved as: prec_jan.txt

TextPad Commands:

- TextPad is used to save the file with a '.tab' extension (This requires putting the name in double quotes: "prec_jan.tab").

ArcCatalogue Commands:

- Right clicked on the file and selected 'Create Feature Class/ From XY Table…"
  - Chose 'jan' as the Z field.
  - Saved as: 'prec_jan.shp'.
- Converted shape file to a point coverage.
  - Right clicked on the file and selected 'Export / Shapefile to Coverage'
    - Selected 'Define Mapping' option.
    - Defined item to be mapped as 'jan'.
    - Creates a point coverage.
    - Saved as: 'prec_jan'.

ArcInfo Commands:

- Defining the projection for the point coverage.

```
Arc: projectdefine cover prec_jan
Define Projection
Project: projection geographic
Project: units dd
Project: parameters
```

- Project the point coverage to an Eckert VI projection.

```
Arc: project cover prec_jan prec_janp
Project: output
Project: projection eckertvi
Project: units meters
Project: parameters
Longitude of projection center          [   0  0  0.000 ]: 50 0
0.000
Project: end
```

- Convert the Eckert VI projected point coverage to a grid. It is necessary to interpolate the points to fill the grid completely for a resolution of 50 km. Otherwise, using a 'pointgrid' command will leave land grid cells empty (no data) where there was no data point.
- Three attempts are tried to create a 50 km grid that best represents the point data. Attempt three was deemed to be the best (Table 2).

1) A grid was created by interpolating from points was to use the 'pointinterp' GRID command.

```
Grid: pjan_gr_p2 = pointinterp(prec_janp,jan,50000,idw_smooth)
```

  - An inverse distance weighting with a smoothing factor, rather than an exponential weighting is used to determine the grid cell value. A default search neighbourhood is used, which turns out to be 5107289.548 metres.
  - The resultant grid does not interpolate well, and is not a 50 km grid.

- The window was set so the next two resultant grids fall in line with the topography grid.

```
Grid: setwindow ~msc0228/dissertation/data/topography/topo82_e6
```

2) Created a grid by interpolated from the point coverage 'prec_janp'.

```
Grid: pjan_pr_p3 = idw(prec_janp, jan, #, #, sample, 1, 50000,
50000,#)
```

  - This grid follows the point data very closely, by assigning a grid cell the value of the point that falls within it. If a grid cell does not contain a point, it is assigned the value of the closest point. However, if an empty grid cell is greater than 50000 m from a point it is assigned a no data value. There are several no data areas and this is unacceptable. The resultant grid needs to have a value for every land grid cell.

3) The procedure from attempt 2 was repeated, expect, if no point value falls within a grid cell the search for another point will extend to 60,000 instead of 50,000 metres. This results in only a few nodata values occurring on the land. Since it is not known for sure whether these areas represent small lakes, this grid is deemed acceptable.

```
Grid: pjan_pr_p4 = idw(prec_janp, jan, #, #, sample, 1, 60000,
50000,#)
```

  - Grid 'pjan_pr_p4' is multiplied with the land sea mask grid to refine 'pjan_pr_p4' to a more accurate coverage of the land.

```
Grid: pjanprp4n = pjan_pr_p4 *
~msc0228/dissertation/data/topography/mask0k_null
```

Table 6. Comparing the mean precipitation values from the original point coverage 'prec_janp' to the grid values created over three attempts through various interpolation specifications. The grid file names are cited in brackets.

Precipitation Values			
Original Data (prec_janp)	**Attempt One (pjan_pr_p2)**	**Attempt Two (pjan_pr_p3)**	**Attempt Three (pjan_pr_p4)**
198	126	198	198
-	104	198	198
77	101	77	77
112	90.6	112	112
59	77.4	59	59
152	126.9	152	152
89	106.9	89	89
-	88.3	89	89
67	76.4	67	67
56	61.8	56	56

16.2.4 Mean July Precipitation

(source data: dissertation/data/precipitation/prec.grd.gz)

Excel Commands:
- Excel is used to add the header information (x, y, jul) and change the file from being space delimited to tab delimited.
- At line 55739, data had to be shifted over one column to correspond to the preceding data.
- File was saved as: prec_jul.txt

TextPad Commands:
- TextPad is used to save the file with a '.tab' extension (This requires putting the name in double quotes: "prec_jul.tab").

ArcCatalogue Commands:
- Right clicked on the file and selected 'Create Feature Class/ From XY Table…"
  - Saved as: 'prec_jul.shp'.
- Converted shape file to a coverage file.
  - Right clicked on the file and selected 'Export/ Shapefile to Coverage'
    - Selected 'Define Mapping' option.
    - Defined item to be mapped as 'jul'.
    - Creates a point coverage.
    - Saved as: 'prec_jul'.

ArcInfo Commands:
- Defining the projection for the point coverage.

```
Arc: projectdefine cover prec_jul
Define Projection
Project: projection geographic
Project: units dd
Project: parameters
```

- Project the point coverage to an Eckert VI projection.

```
Arc: project cover prec_jul prec_julp
Project: output
Project: projection eckertvi
Project: units meters
Project: parameters
Longitude of projection center          [   0  0  0.000 ]: 50 0
0.000
Project: end
```

ArcInfo GRID Commands:
- The window was set so the resultant grid falls in line with the topography grid.

```
Grid: setwindow ~msc0228/dissertation/data/topography/topo82_e6
```

- Interpolating a 50 km grid from the projected point coverage 'prec_julp'.

```
Grid: pjul_pr_p4 = idw(prec_julp, july, #, #, sample, 1, 60000,
50000,#)
```

    - Grid 'pjul_pr_p4' is multiplied with the land sea mask grid to refine 'pjan_pr_p4' to a more accurate coverage of the land.

```
Grid: pjulprp4n = pjul_pr_p4 *
~msc0228/dissertation/data/topography/mask0k_null
```

16.2.5 Mean January Temperature

(source data: dissertation/data/temperature/tmean.grd.gz)

UNIX Commands:
- Unzipped the file.

```
bash-2.02$ gunzip tmean.grd.gz
```

- File now saved as: tmean.grd

Excel Commands:

- Excel is used to add the header information (x,y,jan) and change the file from being space delimited to tab delimited.
- At line 55739, data had to be shifted over one column to correspond to the preceding data.
- File was saved as: temp_jan.txt

TextPad Commands:

- TextPad is used to save the file with a '.tab' extension (This requires putting the name in double quotes: "temp_jan.tab").

ArcCatalogue Commands:

- Right clicked on the file and selected 'Create Feature Class/ From XY Table…"
  - Chose an Eckert VI Projection.
  - Modified projection to have a central meridian at 50°.
  - Saved as: 'temp_jan.shp'.
- Converted shape file to a coverage file.
  - Right clicked on the file and selected 'Export / Shapefile to Coverage'
    - Selected 'Define Mapping' option.
    - Defined item to be mapped as 'jan'.
    - Creates a point coverage.
    - Saved as: 'temp_jan'.

ArcInfo Commands:

- Defining the projection for the point coverage.

```
Arc: projectdefine cover temp_jan
Define Projection
Project: projection geographic
Project: units dd
Project: parameters
```

- Project the point coverage to an Eckert VI projection.

```
Arc: project cover temp_jan temp_janp
Project: output
Project: projection eckertvi
Project: units meters
Project: parameters
Longitude of projection center           [   0  0  0.000 ]: 50 0
0.000
Project: end
```

ArcInfo GRID Commands:

- The window was set so the resultant grid falls in line with the topography grid.

```
Grid: setwindow ~msc0228/dissertation/data/topography/topo82_e6
```

- Interpolating a 50 km grid from the projected point coverage 'temp_janp'.

```
Grid: tjan_pr_p4 = idw(temp_janp, jan, #, #, sample, 1, 60000,
50000,#)
```

- Grid 'tjan_pr_p4' is multiplied with the land sea mask grid to refine 'tjan_pr_p4' to a more accurate coverage of the land.

```
Grid: tjanp4n = tjan_pr_p4 *
~msc0228/dissertation/data/topography/mask0k_null
```

16.2.6 Mean July Temperature

(source data: dissertation/data/temperature/tmean.grd)

Excel Commands:

- Excel is used to add the header information (x, y, jul) and change the file from being space delimited to tab delimited.
- At line 55739, data had to be shifted over one column to correspond to the preceding data.
- File was saved as: temp_jul.txt

TextPad Commands:

- TextPad is used to save the file with a '.tab' extension (This requires putting the name in double quotes: "temp_jul.tab").

ArcCatalogue Commands:

- Right clicked on the file and selected 'Create Feature Class/ From XY Table…"
    - Saved as: 'temp_jul.shp'.
- Converted shape file to a coverage file.
    - Right clicked on the file and selected 'Export/ Shapefile to Coverage'
        - Selected 'Define Mapping' option.
        - Defined item to be mapped as 'jul'.
        - Creates a point coverage.
        - Saved as: 'temp_jul'.

ArcInfo Commands:

- Defining the projection for the point coverage.

```
Arc: projectdefine cover temp_jul
Define Projection
Project: projection geographic
Project: units dd
Project: parameters
```

- Project the point coverage to an Eckert VI projection.

```
Arc: project cover temp_jul temp_julp
Project: output
Project: projection eckertvi
Project: units meters
Project: parameters
Longitude of projection center          [   0  0  0.000 ]: 50 0
0.000
Project: end
```

ArcInfo GRID Commands:

- The window was set so the resultant grid falls in line with the topography grid.

```
Grid: setwindow ~msc0228/dissertation/data/topography/topo82_e6
```

- Interpolating a 50 km grid from the projected point coverage 'temp_julp'.

```
Grid: tjul_pr_p4 = idw(temp_julp, july, #, #, sample, 1, 60000,
50000,#)
```

    - Grid 'tjul_pr_p4' is multiplied with the land sea mask grid to refine 'tjul_pr_p4' to a more accurate coverage of the land.

```
Grid: tjulp4n = tjul_pr_p4 *
~msc0228/dissertation/data/topography/mask0k_null
```

16.2.7 Mean Temperature Range:

The temperature range can be derived by subtracting the January mean from the July mean in the Northern Hemisphere and vice versa for the Southern Hemisphere. For most places in the study area this approach will be an appropriate means of deriving the mean. However, this is less appropriate in monsoon areas where heavy rain during summer months act to reduce temperatures, resulting in spring being the warmest season (Strahler and Strahler, 1997).

Excel Commands

The 'temp_jan.txt' and 'temp_jul.txt' files stored in \temperature were used to create a 'trange_grd.txt', which is also stored in \temperature. To create this file 'trange' was calculated as tjul - tjan when tjul > tjan and tjan - tjul when tjan > tjan.

TextPad Commands:

- TextPad is used to save the file with a '.tab' extension (This requires putting the name in double quotes: "trange_grd.tab").

ArcCatalogue Commands:

- Right clicked on the file and selected 'Create Feature Class/ From XY Table…"
  - Saved as: 'tmp_r.shp'.
- Converted shape file to a coverage file.
  - Right clicked on the file and selected 'Export/ Shapefile to Coverage'
    - Selected 'Define Mapping' option.
    - Defined item to be mapped as 'trange'.
    - Creates a point coverage.
    - Saved as: 'tmp_r'

ArcInfo Commands:

- Defining the projection for the point coverage.

```
Arc: projectdefine cover tmp_r
Define Projection
Project: projection geographic
Project: units dd
Project: parameters
```

- Project the point coverage to an Eckert VI projection.

```
Arc: project cover tmp_r tmp_rp
Project: output
Project: projection eckertvi
Project: units meters
Project: parameters
Longitude of projection center          [   0  0  0.000 ]: 50 0 
0.000
Project: end
```

ArcInfo GRID Commands:

- The window was set so the resultant grid falls in line with the topography grid.

```
Grid: setwindow ~msc0228/dissertation/data/topography/topo82_e6
```

- Interpolating a 50 km grid from the projected point coverage '`tmp_rp`'.

```
Grid: tmp_rpg = idw(tmp_rp, trange, #, #, sample, 1, 60000, 
50000,#)
```

  - Grid '`trange_pg`' is multiplied with the land sea mask grid to refine '`trange_pg`' to a more accurate coverage of the land.

```
Grid: tmprng = tmp_rpg * 
~msc0228/dissertation/data/topography/mask0k_null
```

This file is save in \temperature. A copy of this grid is saved as: `~msc0228/dissertation/aml/trange.`

16.2.8 Continentality Grid and the Land/Sea Mask

ArcInfo GRID commands:

- To create a continentality grid it is necessary to make a land/sea mask. The topography data (file: topo82_e6) was used as the source data. From GRID all cells greater than zero were set to 1 and all cells less than or equal to zero were set to 0.

```
Grid: mask0k = con(topo82_e6 > 0, 1, 0).
```

The water values in the 'mask0k grid are then set to nodata.
`Grid: mask0k_null = setnull(mask0k = 0, mask0k)` to create 'mask0k_null'.

- Continentality values are derived using a distance weighted focal sum.

```
Grid: cont_setup = focalsum(mask0k, weight, weight_values, data)
```

- 'mask0k' is the land/sea mask grid of 0 and 1 values. 'weight_values' is an ASCII grid file defining the weightings to be used. This values of this file are calculated using the function $y = x^{1/3}$. This file is found at 'dissertation/data/topography/weight_values. It was created in Excel and then saved as a space deliminated file. Some of the spaces were removed in TextPad to make the format ArcInfo friendly, and this file also saved for UNIX formatting.
- To convert the grid to values ranging from 0 to 1 range, ('focalsum' only adds up the values), the grid was divided by its maximum value, 85.744.

```
Grid: cont_final = cont_setup / 85.744
```

- A second grid was created that only has continentality values for the land. The water grid cells are assigned 'nodata' values. This grid is saved as 'cont_landn'.

```
Grid: cont_land = mask0k * cont_final
Grid: cont_landn = setnull(cont_land == 0, cont_land)
```

See the Excel file 'dissertation/data/topography/continent_distance_weight.xls' for a comparison of functions.

16.3 <u>Commands for Creating the Biome Model</u>

16.3.1 Creating the point coverages of biome 0k data

(source data: *dissertation/data/biome/biome6000berengia xls*)
All files are saved in the directory *dissertation/data/biome*.

Excel Commands:

- Deleted all data that was not Eurasian or African and saved as: 'biome_eurasia_af_raw.xls'.
- Deleted all columns but latitude and longitude. Switched latitude and longitude columns and labelled longitude as 'x' and latitude as 'y'.
- A column was added in called 'biome', that identifies each biome type by a unique integer value defined in the sheet labelled 'Erin's Key'. This is saved as: 'biome0k_key.xls'.
- A text file of only the coordinates and integer biome value from 'biome0k_key.xls' is saved as 'bio0k.txt'.

TextPad Commands:

- TextPad is used to save the .txt file as a .tab file. This is necessary for subsequent data processing in ArcGIS.

ArcCatalogue Commands:

- Right clicked on the file and selected 'Create Feature Class/ From XY Table…"
    - Saved as: 'bio0k.shp'.

ArcMap Commands:

- Select only the data that applies to the study area and right click on the shape file name to 'Data/Export Data' this data as a shapefile.

ArcCatalogue Commands:

- Converted shape file to a coverage file.
    - Right clicked on the file and selected 'Export/ Shapefile to Coverage'
        - Selected 'Define Mapping' option.
        - Defined item to be mapped as 'Grid0k'.
        - Creates a point coverage.
        - Saved as: 'bio0k'.

ArcInfo Commands:

- Defining the projection of 'export_bio0k'

```
Arc: projectdefine cover bio0k
Define Projection
Project: projection geographic
Project: units dd
Project: parameters
```

- Projecting ‘export_bio0k’ in Arc to an Eckert VI projection:

```
Arc: project cover bio0k bio0k_p
Project: output
Project: projection eckertvi
Project: units meters
Project: parameters
Longitude of projection center          [   0  0  0.000 ]: 50
0 0.000
Project: end
```

16.3.2 Identifying Biome Types Coinciding with Known Wildcat Sightings Data

16.3.2.1 Producing grids of the BIOME 6000 data

Example of ArcInfo GRID Commands for the 0k BIOME 6000 data:

- The point coverage 'bio0k_p' was converted into a raster coverage. The output grid specifications were set to be the same as the 'dem' grid used for statistics. The value item mapped was 'Grid0k', and cell size was set as 50000. The grid was saved as *dissertation/data/biome/bio0*.

```
Grid: setwindow ~msc0228/dissertation/statistics/dem
Grid: bio0 = pointgrid(bio0k_p,grid0k,#,#,50000)
```

- The grids for 6k, 9k, and 18k BIOME 6000 data were saved as: *dissertation/data/biome/bio6; /bio9; /bio18*.

16.3.2.2 Intersecting the BIOME 6000 0k data with sightings data from Nowell and Jackson (1996)

- The Nowell and Jackson (1996) polygon coverage is converted into a grid of values 1000 and 100 (*dissertation/evaluation/njcats2*). The BIOME 6000 0k grid is subtracted from the Nowell and Jackson (1996) grid to derive the confusion matrix count (*dissertation/evaluation/hb_contmtx*).

16.3.3 Ranking Wildcat Preference of the BIOME 6000 data

- Biome types in the grids containing BIOME 6000 data were reclassified to a four point scale from 0 to 3 for wildcat preference. The reclassified values are defined by the UNIX formatted text file: *dissertation/data/biome/hab_rcl*.

```
Grid: bio0rcl = reclass(bio0,hab_rcl,nodata)
Grid: bio6rcl = reclass(bio6,hab_rcl,nodata)
Grid: bio9rcl = reclass(bio9,hab_rcl,nodata)
Grid: bio18rcl = reclass(bio18,hab_rcl,nodata)
```

16.3.4 Assessing results with sightings data from Green (2000)

- Intersecting the modern day wildcat sightings coverage *dissertation/data/wildcat_sightings/presence_p* with the point coverage of BIOME 6000 0k data, *dissertation/data/biome/bio0k_p* identifies the biome types where modern day wildcats have been found.

```
Arc: intersect bio0k_p
~msc0228/dissertation/data/wildcat_sightings/presence_p hab_rules
point
```

The intersect command produces a point coverage (*dissertation/data/biome/hab_rules*).

16.4 Commands for Creating the Secondary Biome Model

16.4.1 Data Processing the Environmental Variable Grids

- **Clipping the environmental variable data to the same geographic extents:**

ArcInfo Commands:

```
Arc: gridclip ~msc0228/dissertation/aml/cpre cpre_s -8750000 -8165497 13950000 8184503
Arc: gridclip ~msc0228/dissertation/aml/ctmp ctmp_s -8750000 -8165497 13950000 8184503
Arc: gridclip ~msc0228/dissertation/aml/trange trange_s -8750000 -8165497 13950000 8184503
```

The resultant grids have 327 rows and 454 columns, giving a total of 148458 grid cells. These grids are saved in: *dissertation/statistics*.

- **Setting the grids to have the same number of 'no data' (-9999) values and that occur in the same location:**

ArcInfo GRID Commands:

- Each environmental variable grid was changed to values of 1 and -9999 by first removing all zero values and then by dividing the resulting grid by itself.
- Removing the zero values:

```
Grid: cpre0 = con(cpre_s < 1,1,cpre_s)
Grid: ctmp0 = con(ctmp_s == 0,1,ctmp_s)
Grid: trange0 = con(trange_s == 0, 1, trange_s)
```

These grids are saved in: 'dissertation/statistics'.

- Producing Grids of Values 1imb and -9999:

```
Grid: cprebin = cpre0 / cpre0
Grid: ctmpbin = ctmp0 / ctmp0
Grid: trangebin = trange0 / trange0
```

These grids are saved in: 'dissertation/statistics'. Multiplying these grids together creats a single grid containing all -9999 values occurring in every environmental variable grid.

```
Grid: bin = cprebin * ctmpbin * trangebin
```

This grid was saved as *dissertation/statistics/bin*.

- Multiplying *dissertation/statistics/bin* by each environmental variable grid yields environmental variable grids with the same number of *no data* values and occurring in the same location.

```
Grid: cpreb = cpre_s * bin
Grid: ctmpb = ctmp_s * bin
Grid: trangeb = trange_s * bin
```

- These grids were then clipped to the study area.

```
Grid: setwindow -8750000 -8165497 13950000 8184503
Grid: caprebb = cpreb * ~msc0228/dissertation/maps/study_area
Grid: ctmpbb = ctmpb * ~msc0228/dissertation/maps/study_area
Grid: trangebb = trangeb * ~msc0228/dissertation/maps/study_area
```

- These grids were then exported as ASCII files, and opened and saved in TextPad with the 'txt' extension.

```
Arc: gridascii cprebb gridcpreb
Arc: gridascii ctmpbb gridctmpb
Arc: gridascii trangebb gridtrangeb
```

16.4.2 Getting environmental variable data for each biome data point

16.4.2.1 Mean Annual Surface Precipitation

ArcInfo Commands:

- To convert a grid to a polygon the grid data must be integer data types. The 'int' command in grid will do this by truncating each cell value.

```
Grid: ~msc0228/dissertation/statistics/cpreb_int =
int(~msc0228/dissertation/statistics/cpreb)
```

- Converting the grid into polygons:

```
Arc: gridpoly cpreb_int cpreb_poly
```

- Overlaying the biome point coverage with the grid-shaped polygon coverage to assign the polygon values to each point.

```
Arc: identity ~msc0228/dissertation/data/biome/BIO0K_P ~
Arc: ~msc0228/dissertation/statistics/cpreb_poly cpre_final point
```

- The resulting point coverage includes data on longitude (X), latitude (Y), biome type (biome0k), and precipitation value (grid-code).
- Exporting the point coverage data into MS Excel.

ArcInfo TABLES Commands:

```
Tables: sel cpre_final.pat
```

- Create an ascii file of selected item data from the point coverage.

```
Tables: unload data_cpre  x y biome0k grid-code
```

File is saved as: dissertation/statistics/data_cpre.

16.4.2.2 Temperature Range

Excel Commands

In Excel temperature range was calculated as tjul - tjan when above 0° latitude, and tjan - tjul when below 0° latitude.

- File was saved as: trange.txt

TextPad Commands:

- TextPad is used to save the file with a '.tab' extension (This requires putting the name in double quotes: "trange.tab").

ArcCatalogue Commands:

- Right clicked on the file and selected 'Create Feature Class/ From XY Table…"
    - Saved as: 'trange.shp'.
- Converted shape file to a coverage file.
    - Right clicked on the file and selected 'Export/ Shapefile to Coverage'
        - Selected 'Define Mapping' option.
        - Defined item to be mapped as 'trange'.
        - Creates a point coverage.
        - Saved as: 'trange'.

ArcInfo Commands:

- Defining the projection for the point coverage.

```
Arc: projectdefine cover trange
Define Projection
Project: projection geographic
Project: units dd
Project: parameters
```

- Project the point coverage to an Eckert VI projection.

```
Arc: project cover trange trange_p
Project: output
Project: projection eckertvi
Project: units meters
Project: parameters
```

```
Longitude of projection center          [   0  0  0.000 ]: 50 0
0.000
Project: end
```

ArcInfo GRID Commands:

- The window was set so the resultant grid falls in line with the topography grid.

```
Grid: setwindow ~msc0228/dissertation/data/topography/topo82_e6
```

- Interpolating a 50 km grid from the projected point coverage `'trange_p'`.

```
Grid: trange_pg = idw(trange_p, trange, #, #, sample, 1, 60000,
50000,#)
```

  - Grid '`trange_pg`' is multiplied with the land sea mask grid to refine '`trange_pg`' to a more accurate coverage of the land.

```
Grid: trange = trange_pg *
~msc0228/dissertation/data/topography/mask0k_null
```

This file is saved as ~msc0228/dissertation/data/biome/trange.

ArcInfo Commands:

- To convert a grid to a polygon the grid data must be integer data types. The 'int' command in grid will do this by truncating each cell value.

```
Grid: trange_int = int(trangeb)
```

File is saved as: ~msc0228/dissertation/statistics/trange_int.

- Converting the grid into polygons:

```
Arc: gridpoly trange_int trange_poly
```

- Overlaying the biome point coverage with the grid-shaped polygon coverage to assign the polygon values to each point.

```
Arc: identity ~msc0228/dissertation/data/biome/BIO0K_P ~
Arc: trange_poly trange_final point
```

File is saved as: ~msc0228/dissertation/statistics/trange_final

- The resulting point coverage includes data on longitude (X), latitude (Y), biome type (biome0k), and precipitation value (grid-code).
- Exporting the point coverage data into MS Excel.

ArcInfo TABLES Commands:

```
Tables: sel trange_final.pat
```

- Create an ascii file of selected item data from the point coverage.

```
Tables: unload data_trange  x y biome0k grid-code
```

File is saved as: dissertation/statistics/data_trange.

16.4.2.3 Mean Annual Surface Temperature

ArcInfo Commands:

- To convert a grid to a polygon the grid data must be integer data types. The 'int' command in grid will do this by truncating each cell value.

```
Grid: ctmpb_int = int(ctmpb)
```

File is saved as: dissertation/statistics/ctmpb_int

- Converting the grid into polygons:

```
Arc: gridpoly ctmpb_int ctmp_poly
```

- Overlaying the biome point coverage with the grid-shaped polygon coverage to assign the polygon values to each point.

```
Arc: identity ~msc0228/dissertation/data/biome/BIO0K_P ~
Arc: ctmp_poly  ctmp_final point
```

- The resulting point coverage includes data on longitude (X), latitude (Y), biome type (biome0k), and precipitation value (grid-code).

ArcInfo TABLES Commands:

```
Tables: sel ctmp_final.pat
```

- Create an ascii file of selected item data from the point coverage.

```
Tables: unload data_ctmp  x y biome0k grid-code
```

16.4.3 Identifying Biome Types Coinciding with Known Wildcat Sightings Data

- Correspondence with sightings data from Green (2000):

Arc: intersect biosw3 ~msc0228/dissertation/data/wildcat_sightings/presence_p sbm poly

- Correspondence with sightings data from Nowell and Jackson (1996):

Grid: sbm_cmtx = ~msc0228/dissertation/evaluation/njcats2 - ~msc0228/dissertation/statistics/biomesw3

16.4.4 Ranking Wildcat Preference of the BIOME 6000 data

- Biome types in the grids containing BIOME 6000 data were reclassified to a four point scale from 0 to 3 for wildcat preference. The reclassified values are defined by the UNIX formatted text file: *dissertation/data/biome/hab_rcl2*.

```
Grid: sbmrcl = reclass(~msc0228/dissertation/statistics/biomesw3,
hab_rcl2, nodata)
```

This grid is saved as: *dissertation/data/biome/sbmrcl.*

16.5 <u>Experiments</u>

All files are saved in: *dissertation/aml.*

16.5.1 Comparing Uniform Temperature Change to a Linear Temperature Change

- **Uniform Temperature Drop Commands:**

Arc Info GRID Commands:

```
Grid: ctmp10 = ctmp - 100
Grid: tjan10 = tjan - 10
Grid: tjul10 = tjul - 10
```

TextPad:

In TextPad the search and replace operation is used to replace these grids in place of the temperature grids used in the 0k outcome: *dissertation/aml/cat0k.aml*. For instance, ctmp10 for ctmp, tjan10 for tjan, and tjul10 for tjul.

- **Non-uniform Temperature Drop Commands:**

Arc Info GRID Commands:

A grid with positive values running from 90°N to 90°S was created by manipulating the 'latitude' grid whose values run from 90°N to -90°S.

```
Grid: latpos = con(latitude < 0, latitude * -1, latitude)
```

This grid was then used to formulate a temperature shift grid.

```
Grid: lat10down = ((1 + ((latpos - 90) / 90)) * 6) + 4
```

The temperature shift grid 'lat10down' was then applied to 'ctmp', 'tjan', and 'tjul' to simulate a 10°C to 4°C drop in global temperature.

```
Grid:  ctmp10down = ctmp - (lat10down * 10)
Grid: tjan10down = tjan - lat10down
Grid: tjul10down = tjul - lat10down
```

(Note: 'lat10down' was multiplied by 10 when creating the 'ctmp10down' grid to keep the units as °C * 10).

TextPad:

In TextPad the search and replace operation is used to replace these grids in place of the temperature grids used in the 0k outcome: *dissertation/aml/cat0k.aml*. For instance, ctmp10down for ctmp, tjan10down for tjan, and tjul10down for tjul.

- When the AML's to create the temperature change outcomes the following GRID message occurs for some of the environmental regions (e.g. dfd):

```
All cells in grid
/import/tsunami15c/msc_stud/msc0228/dissertation/aml/dfd have
NODATA value
VAT will not be built
```

This happens because there are no grid cells that can be classified under rules for the environmental region, thus, a grid for the environmental region cannot be created. For a 10°C uniform drop in temperature this occurs for 'am', 'dfa', 'dfd', and 'dwa'. For a 10°C non-uniform drop in temperature this occurs for 'dfd', and 'dwa'. This makes sense since the more northern environments 'dfd' and 'dwa' would have experienced a more extreme temperature change for both trials, thus, being too drastically different to have any grid cells belonging to their class. For the more southern environments, 'am' and 'dfa', a less extreme temperature change occurred in the non-uniform temperature drop trial, therefore, some cells could still be classified under these environmental classifications. However, it is believed that because changes in precipitation were not taken into account unrealistic conditions were created that did not correspond with any environmental type.

16.5.2 Creating the Rule Based Model 9ka, 13ka, and 18ka Wildcat Distributions

16.5.2.1 Creating Sea Level Change Data for the Rule Based Model

- Sea level change data stored in the directory *dissertation/aml* under the names: *topo*, *topo9*, *topo13*, and *topo18* for the respective 0k, 9k, 13k, and 18k time periods. The following code is an example of processing the 18k sea level change data to produce a grid of 18k sea level change values for use in the RBM.

(source data: dissertation/data/sea_level_change/grid18ke.txt)

TextPad Commands:

- Header data was added to the file.

ncols 360
nrows 180
xllcenter -168
yllcenter -90
cellsize 1

- Data was cut and pasted to create a geographic split at 168°W (or –168°). File saved as 'grid18_fit'.

ArcInfo Commands:

- Importing 'grid18_fit' into Arc:

```
Arc: asciigrid grid18_fit grid18 int
```

- Defining projection for '`grid18`' data:

```
Arc: projectdefine grid grid18
Define Projection
Project: projection geographic
Project: units dd
Project: parameters
```

- Project the '`grid18`' data to an Eckert VI projection:

```
Arc: project grid grid18 grid18_p # cubic 50000 0 0
Project: output
Project: projection eckertvi
Project: units meters
Project: parameters
Longitude of projection center          [   0  0  0.000 ]: 50 0
0.000
Project: end
```

ArcInfo Message: Inversibility is 90 per cent

All grids are saved in as: *dissertation/aml*.

16.5.2.2 Creating Topography and Continentality Grids 9ka, 13ka, and 18ka

ArcInfo GRID Commands:

```
Grid: grid18c = con(grid18 < 50, 0, 1)
Grid: grid13c = con(grid13_p < 50, 0, 1)
Grid: grid9c = con(grid9_p < 50, 0, 1)
Grid: grid18f = setnull(grid18c == 0, grid18c)
Grid: grid13f = setnull(grid13c == 0, grid13c)
Grid: grid9f = setnull(grid9c == 0, grid9c)
```

- New continentality grids are calculated using these grids.

```
Grid: grid18ff = focalsum(grid18f, weight,
~msc0228/dissertation/data/topography/weight_values, data)
Grid: grid13ff = focalsum(grid13f, weight,
~msc0228/dissertation/data/topography/weight_values, data)
Grid: grid9ff = focalsum(grid9f, weight,
~msc0228/dissertation/data/topography/weight_values, data)
Grid: grid18fff = grid18ff / 85.744
Grid: grid13fff = grid13ff / 85.744
Grid: grid9fff = grid9ff / 85.744
```

16.5.2.3 Assessing the 9k and 18k Rule Based Model Outcomes

```
Grid: eval9k = ~msc0228/dissertation/aml/cat9k -
~msc0228/dissertation/data/biome/bio9rcl
Grid: eval18k = ~msc0228/dissertation/aml/cat18k -
msc0228/dissertation/data/biome/bio18rcl
```

16.5.2.4 Comparing the 0k, 9k, 13k, and 18k Outcomes to Each Other

All grids are saved in as: *dissertation/evaluation*.

```
Grid: 0keuro = ~msc0228/dissertation/aml/europe *
~msc0228/dissertation/aml/cat0k
Grid: 0kafr = ~msc0228/dissertation/aml/africa *
~msc0228/dissertation/aml/cat0k
Grid: 0kasia = ~msc0228/dissertation/aml/asia *
~msc0228/dissertation/aml/cat0k
Grid: 9keuro = ~msc0228/dissertation/aml/europe *
~msc0228/dissertation/aml/cat9k
Grid: 9kafr = ~msc0228/dissertation/aml/africa *
~msc0228/dissertation/aml/cat9k
Grid: 9kasia = ~msc0228/dissertation/aml/asia *
~msc0228/dissertation/aml/cat9k
Grid: 13keuro  = ~msc0228/dissertation/aml/europe *
~msc0228/dissertation/aml/cat13k
Grid: 13kafr = ~msc0228/dissertation/aml/africa *
~msc0228/dissertation/aml/cat13k
Grid: 13kasia = ~msc0228/dissertation/aml/asia *
~msc0228/dissertation/aml/cat13k
Grid: 18keuro = ~msc0228/dissertation/aml/europe *
~msc0228/dissertation/aml/cat18k
Grid: 18kafr  = ~msc0228/dissertation/aml/africa *
~msc0228/dissertation/aml/cat18k
Grid: 18kasia = ~msc0228/dissertation/aml/asia *
~msc0228/dissertation/aml/cat18k
```

17 APPENDIX TWO

18 APPENDIX THREE

This appendix contains the all the code required to run the Secondary Biome Model. The program consists of four classes. The BCWeight Class is the main class and the Raster, SimpleIO and Stats classes are supporting classes. The JAVA program is called: BiomeClassifier. All the files are stored in *dissertation/statistics*. Erin E. Rees wrote the majority of the code. Other contributors are acknowledged beside the sections of code they wrote. Comments in the code are prefixed with: //

18.1 The BCWeight Class

The BCWeight Class is the main class for running the Secondary Biome Model. It was written by Erin E. Rees.

```
import java.util.*;
import java.math.*;
import java.io.*;

public class BCWeight {

	//Declaring the variables.

	//Create a SimpleIO object.
	SimpleIO sio = new SimpleIO();

	//Create a Stats object.
	Stats stats = new Stats();

	//Create an empty constructor.
	public BCWeight(){
	}

	//Method to change a vector into an array.
	public static float[] vectArray(Vector v){

		//Declare variables.
		int l;
		float[] a;

		//Calculate the length of the vector.
		l = v.size();

		//Create an array.
		a = new float[l];

		//Add the each vector element to the array.
		for (int i=0; i < l; i++){

			//Remove element i from the vector.
			String element = (String) v.elementAt(i);

			//Convert the string element to a float data type.
			//Create a double object and past it the string element.

			float element2 = (float)
			Double.valueOf(element).doubleValue();

			//Add element to the array.
```

```
                a[i] = element2;

          }

          return a;
    }

    //Method to put all the biome variable ASCII file names in
    //a vector so they can be recalled in a loop in the bioVarStdev method.
    private Vector bioVarGridNames(){

          /*Declare the variables.*/
          Vector v = new Vector();

          /*Add the grid file names to the vector.*/
          v.addElement("U:/dissertation/statistics/bio_cpre.txt");
          v.addElement("U:/dissertation/statistics/bio_ctmp.txt");
          v.addElement("U:/dissertation/statistics/bio_trange.txt");

          return v;
    }

    //Method to put all the environmental variable grid file names in
    //a vector so they can be recalled in a loop in the bioVarStdev
    //method.
    private Vector envVarGridNames(){

          /*Declare the variables.*/
          Vector v = new Vector();

          /*Add the grid file names to the vector.*/
          v.addElement("U:/dissertation/statistics/gridcpreb.txt");
          v.addElement("U:/dissertation/statistics/gridctmpb.txt");
          v.addElement("U:/dissertation/statistics/gridtrangeb.txt");

          return v;
    }

    //Method to create an array to hold the number of values in each
    //biome.
    public static int[] numberBiomes(int n){

          //Create the array.
          int[] a = new int[n];

          //Add values to the array that represent the number of variables
          //per biome.
          a[0] = 297;   //biome 2
          a[1] = 496;  //biome 3
          a[2] = 212;  //biome 4
          a[3] = 92;  //biome 5
          a[4] = 199;  //biome 6
          a[5] = 835;  //biome 7
          a[6] = 745;  //biome 8
          a[7] = 3;  //biome 9
          a[8] = 829;  //biome 10
          a[9] = 50;  //biome 11
          a[10] = 118; //biome 12
          a[11] = 38; //biome 13
          a[12] = 728; //biome 15
          a[13] = 347; //biome 16
          a[14] = 1218; //biome 17
          a[15] = 325; //biome 18
          a[16] = 396; //biome 19
```

```
            a[17] = 6; //biome 20
            a[18] = 4; //biome 21
            a[19] = 11; //biome 23
            a[20] = 10; //biome 24
            a[21] = 8; //biome 25

            return a;
      }

      //Method to create an array to hold the position of where to retrieve
      //data from the biome variable array.
      private int[] positionBioVar(int n){

            //Create the array.
            int[] a = new int[n];

            //Add values to the array that represent the position.
            a[0] = 0;      //biome 2
            a[1] = 297;    //biome 3
            a[2] = 793;   //biome 4
            a[3] = 1005;   //biome 5
            a[4] = 1097;   //biome 6
            a[5] = 1296;   //biome 7
            a[6] = 2131;   //biome 8
            a[7] = 2876;   //biome 9
            a[8] = 2879;   //biome 10
            a[9] = 3708;   //biome 11
            a[10] = 3758; //biome 12
            a[11] = 3876; //biome 13
            a[12] = 3914; //biome 15
            a[13] = 4642; //biome 16
            a[14] = 4989; //biome 17
            a[15] = 6207; //biome 18
            a[16] = 6532; //biome 19
            a[17] = 6928; //biome 20
            a[18] = 6934; //biome 21
            a[19] = 6938; //biome 23
            a[20] = 6949; //biome 24
            a[21] = 6959; //biome 25

            return a;
      }

//Method to calculate the difference statistic (DF) for variable i of
//each biome type and put this data into an array.
public Vector bioVarStdev(Vector bioFile, Vector envVarFile, int[] noBios,
int[] position,int nbio){

      //Declare Variables.
      Vector vFinal = new Vector();

      //Counter for the fileNames.
      int f = 0;

      //Calculate the difference statistic for test point i for all the
      //environmental variables by looping through each environmental variable
      //grid.
      int s = envVarFile.size();

      for (int w=0; w<s; w++){

            //Declare Variables.
            double stdevBiome,power,sumStat,stat;
            float value;
```

```
            value = 0;
            stdevBiome = 0;
            Raster raster;
            raster = null;
            Vector v = new Vector();
            SimpleIO sio = new SimpleIO();
            int no = 0;
            int pos = 0;
            int count = 0;
            int c = 0;
            int p = 0;

            //Get the biome variable ASCII file name from the vector of file
            //names.
            String bioVFile = (String) bioFile.elementAt(f);

            //Calling a method from the SimpleIO class to create a vector of
            //lines from the biome variable ASCII file.
            Vector lines;
            lines = null;
            try{lines = sio.getLines(bioVFile);}catch (Exception eee)
            {System.out.println("Error - cannot create vector on lines from
            file");System.exit(1);}

            //Calling method to change the vector of lines into an array.
            float[] a = BCWeight.vectArray(lines);

            //Get the environmental variable GRID file name from the vector of
            //file names.
            String eVFile = (String) envVarFile.elementAt(f);

            //Create a Raster object.
            try{raster = new Raster(eVFile);}catch (Exception ee)
{System.out.println("Error - cannot create raster object");System.exit(1);}

            //Change the 2D array into a 1D array.
            float[] rasterOneD = raster.toOneD(raster);

            //Calculate the difference statistic sum for each environmental
            //variable value.
            for (int i=0; i<rasterOneD.length; i++){

                  //Create an array to hold the difference statistic data for
                  //each biome type for test varible i.  Set the size the
                  //array being returned for each test point as the size of
                  //the number of biomes.
                  double[] varI = new double[nbio];

                  //Reset p and c.
                  p = 0;
                  c = 0;

                  //Loop through the ASCII environmental variable data.

                  for (int r=0; r<noBios.length; r++){

                        //Retrieve the 'no' and 'pos' values.
                        no = noBios[p];
                        pos = position[p];

                        sumStat = 0;

                        //Create an array of the biome.
```

```
                        float[] bioArray = new float[no];
                        for (int j=0; j<no; j++){
                              bioArray[j] = a[j+pos];
                        }

                        //Calculate the standard devation of the biome.
                        stdevBiome = stats.sDev(bioArray);

                        //Set 'value' equal to the current value in
                        //rasterOneD.
                        value = rasterOneD[i];

                        //Do not calculate difference statistic for -9999 (no
                        //data) values.
                        if (value == -9999)
                              sumStat = -9999;
                        else
                        //Calculate the difference statistic for the biome.
                              for (int m=0; m<no; m++){
                  stat = Math.pow((bioArray[m] - rasterOneD[i])/stdevBiome,2);
                                    sumStat = sumStat + stat;
                              }

                        //Add this value to array varI.
                        varI[c] = sumStat;

                        //Add 1 to the counter 'c' so the values added to the
                        //array proceed one by one.
                        c = c + 1;

                        //Add 1 to the counter 'p' so the next 'no' and 'pos'
                        //values are accessed.
                        p = p + 1;

                  }
            //Add the array of biome difference statistics for test point i to
            //the vector.
            v.addElement(varI);

            }
            //Add the vector of difference statistics for variable i a vector
            //that contain all the data.
            vFinal.addElement(v);

            //Next files.
            f = f + 1;

      }
      return vFinal;
}

// Method add up the difference statistics of each test point for
//all the environmental variables.
public double[] sumDF(Vector vFinal, int nbio, double weight){

      //Declare the variables.
      Vector vDFSum = new Vector();
      double DFSumBio;
      //Variable 'c' keeps track of the position for arrayPt.
      int c = 0;

      //Create an array that has the number of records per biome.
      //Use these values to standardise the summed difference statistics
      //calculated in this method.
```

```
	int[] noBioRecords = BCWeight.numberBiomes(nbio);

	//Size of the array is number of biomes * number of test points
	//double[] arrayPt = new double[75];
	double[] arrayPt = new double[3266076];

	//Extract the vectors containing the environmental variable data.
	Vector cpre = (Vector) vFinal.elementAt(0);
	Vector ctmp = (Vector) vFinal.elementAt(1);
	Vector trange = (Vector) vFinal.elementAt(2);

	//Loop through each test point.
	for (int j=0; j<cpre.size(); j++){
		double[] arrayTstPtCpre = (double[]) cpre.elementAt(j);
		double[] arrayTstPtCtmp = (double[]) ctmp.elementAt(j);
		double[] arrayTstPtTrange = (double[]) trange.elementAt(j);

		//Where the threshold nbio is the number of biomes.
		for (int i=0; i<nbio; i++){
			double cpr = arrayTstPtCpre[i];
			double ctp = arrayTstPtCtmp[i];
			double trng = arrayTstPtTrange[i];

			//Do not include no data values when summing the
			//difference statistics.  Keep the nodata value as
			//-9999.
			if (cpr == -9999)
				DFSumBio = -9999;
			else
			//Multiply values by 1000000 to reduce the size of the
			//cpr, ctp, and trng values.
	DFSumBio =  cpr/(noBioRecords[i]*1000000/3) +
	ctp/(noBioRecords[i]*1000000) + trng/(noBioRecords[i]*1000000);

			int position = c + i;

			arrayPt[position] = DFSumBio ;

		}

			//Increase c so to progress the array to the next position
			c = c + nbio;
	}

	return arrayPt;
}

//Method to return the position of the minimum value in a double array[]
	public static double position(double array[]){
		double min, value;
		min = Double.MAX_VALUE;
		double posmin = 0;

		for (int i = 0; i< array.length; i++){
			value = array[i];

			if (value !=-9999){
				min = Math.min(min, array[i]);

					if (min == array[i]){
						posmin = i;
					}
			}
			if (value == -9999)
```

```
                        posmin = -9999;
                }
        return posmin;
        }

//Method to return the position of the minimum value in array of DF Sums
//for all test points.
public double[] posVector(double array[],int nbio){

        //Declare the variables.
        //Length of the vector to hold the biome types is equal
        //to the number of test points
        double[] posV = new double[148458];
        double[] currentArray = new double[nbio];

        //Create a counter to proceed through the array of DF Sums one by one.
        int c = 0;

        for (int j=0; j<148458; j++){

                //Create an array of the DF sums for each test point j.
                //Where nbio is the number of DF sums (i.e. number of biomes) for
                //each test point.
                        for (int i=0; i<nbio; i++){
                                currentArray[i] = array[i+c];
                        }

                        //Find the position of the minimum value.
                        double pos = BCWeight.position(currentArray);

                        //Add this value to the array.
                        posV[j] = pos;

                        //Move to the next 3 values in the array of DF sums.
                        c = c + nbio;
        }
        return posV;
}

//Method to change the minimum position values into biome values.
public double[] posToBiome(double[] array){

        //Create an array to hold the assigned biome values.
        double[] bioArray = new double[array.length];

        //Loop through each value in the array containing the minimum position
        //data.
        for (int i=0; i<array.length; i++){

                if (array[i] == 0)
                        bioArray[i] = 2;

                else if (array[i] == 1)
                        bioArray[i] = 3;

                else if (array[i] == 2)
                        bioArray[i] = 4;

                else if (array[i] == 3)
                        bioArray[i] = 5;

                else if (array[i] == 4)
                        bioArray[i] = 6;
```

```
            else if (array[i] == 5)
                  bioArray[i] = 7;

            else if (array[i] == 6)
                  bioArray[i] = 8;

            else if (array[i] == 7)
                  bioArray[i] = 9;

            else if (array[i] == 8)
                  bioArray[i] = 10;

            else if (array[i] == 9)
                  bioArray[i] = 11;

            else if (array[i] == 10)
                  bioArray[i] = 12;

            else if (array[i] == 11)
                  bioArray[i] = 13;

            else if (array[i] == 12)
                  bioArray[i] = 15;

            else if (array[i] == 13)
                  bioArray[i] = 16;

            else if (array[i] == 14)
                  bioArray[i] = 17;

            else if (array[i] == 15)
                  bioArray[i] = 18;

            else if (array[i] == 16)
                  bioArray[i] = 19;

            else if (array[i] == 17)
                  bioArray[i] = 20;

            else if (array[i] == 18)
                  bioArray[i] = 21;

            else if (array[i] == 19)
                  bioArray[i] = 23;

            else if (array[i] == 20)
                  bioArray[i] = 24;

            else if (array[i] == 21)
                  bioArray[i] = 25;

            else if (array[i] == -9999)
                  bioArray[i] = -9999;

      }
      return bioArray;
}


//Method to change the biome values held in a 1D array into a 2D array.
//for all test points.
public double[][] twoDArray(double[] OneDArray){
      //Create a 2D raster to hold the OneDArray.
      //The dimensions of the 2D raster are the dimensions of grid
```

```
        //to be created.
        double[][] twoD = new double[327][454];

        //Create a variable to keep track of the OneDArray values.
        int c = 0;

        for (int i=0; i<327; i++){
                for (int j=0; j<454; j++){
                        twoD[i][j] = OneDArray[c];
                        c = c + 1;
                }
        }
        return twoD;
}

//Method to write a Raster object out to a text file.
//for all test points.
//This method was created by Hadeway Sint (M.Sc. Student 2002),
//and was modified by Erin E. Rees.
public void exportRaster(Raster biomeValues, String fileName){

            //Declare the variables.
            int i,j,k;
            Vector lines = new Vector();
            String s;

            //Variables that define the raster object.
            double[][] array = biomeValues.getRas();
            int nCols = biomeValues.getNCols();
            int nRows = biomeValues.getNRows();
            int xll = biomeValues.getXll();
            int yll = biomeValues.getYll();
            int cellSize = biomeValues.getCellSize();

            s="ncols         "+nRows;
            lines.addElement(s);

            s="nrows         "+nCols;
            lines.addElement(s);

            s="xllcorner     "+xll;
            lines.addElement(s);

            s="yllcorner     "+yll;
            lines.addElement(s);

            s="cellsize      "+cellSize;
            lines.addElement(s);

            s="NODATA_value  -9999";
            lines.addElement(s);

            for (i=0;i<nCols;i++){

                  s="";

                  for(j=0;j<nRows;j++){
                        s=s+" "+array[i][j];
                  }

                  lines.addElement(s);
            }
            try{sio.writeLinesFile(fileName, lines);
```

```
            }catch (Exception e){System.out.println("file not found");
System.exit(1);}

}

//THIS IS THE MAIN METHOD
// Launches the program by creating a new BCWeight object.
public static void main(String args[])throws Exception{

//Creating a new instance of the BCWeight object.
BCWeight bc = new BCWeight();

//Method to put the biome variable file names in a vector.
Vector bioFileNames = bc.bioVarGridNames();

//Method to put the environmental variable file names in a vector.
Vector fileNames = bc.envVarGridNames();

//Method to create an array of the number of varibles per biome.
int[] numbBios = bc.numberBiomes(22);

//Method to create an array of the position to retrieve data from
//the biome variable array.
int[] positBioVar = bc.positionBioVar(22);

//Method to calculate the difference statistics.
Vector t= bc.bioVarStdev(bioFileNames, fileNames, numbBios, positBioVar, 22);

//Method to sum the difference statistics.
double[] sum = bc.sumDF(t, 22, 4);

//Method to calculate the minimum positions in the array of DF Sums for each
//test point.
double[] posV = bc.posVector(sum,22);

//Method to convert the minimum position values into biome types.
double[] bio = bc.posToBiome(posV);

//Method to change a 1D array into a 2D array.
double[][] twoD = bc.twoDArray(bio);

//Method to create a raster from a 2D int array.
Raster r = new Raster(twoD,327,454,-8750000,-8165497,50000);

//Method to write the raster object to an ASCII file.
bc.exportRaster(r, "U:/dissertation/statistics/biomes.asc");

}
```

18.2 The Raster Class

The constructor and methods called upon from the Raster Class:

```
//Create an overloaded constructer so a Raster object can be created
//with the following information: 2D array, number of columns, number of rows,
//lower left x and y coordinates, and the grid cell size.
public Raster(double [][] array,int nCols,int nRows,int xllCorner,int
yllCorner,int cellSize){
            this.ras = array;
            this.nCols = nCols;
            this.nRows = nRows;
            this.xllCorner = xllCorner;
            this.yllCorner = yllCorner;
            this.cellSize = cellSize;
}
```

```
// Method which loads a raster
private void loadRaster(String fileName) throws Exception{

      SimpleIO sio = new SimpleIO();

      // Get all the data in our file
      Vector data = sio.getLines(fileName);

      this.readHeader(data,sio);

      this.readData(data,sio);

}

// Read the header of a raster file
private void readHeader(Vector data, SimpleIO sio) throws Exception{

      /*From the metadata we know that the header is of the following form
            NCOLS 401
            NROWS 401
            XLLCORNER 200000
            YLLCORNER 760000
            CELLSIZE 50
            NODATA_value -9999
            */
            //We fish out the different elements and set the instance
            //variables
            String s = (String) data.elementAt(0);

            Vector line = sio.parseLine(s," ");

            // Fish out the second element, i.e the number
            this.nCols = Integer.parseInt((String) line.elementAt(1));

            s = (String) data.elementAt(1);
            line = sio.parseLine(s," ");
            // Fish out the second element, i.e the number
            this.nRows = Integer.parseInt((String) line.elementAt(1));

            s = (String) data.elementAt(2);
            line = sio.parseLine(s," ");
            // Fish out the second element, i.e the number
            this.xllCorner= Integer.parseInt((String) line.elementAt(1));

            s = (String) data.elementAt(3);
            line = sio.parseLine(s," ");
            // Fish out the second element, i.e the number
            this.yllCorner = Integer.parseInt((String) line.elementAt(1));

            s = (String) data.elementAt(4);
            line = sio.parseLine(s," ");
            // Fish out the second element, i.e the number
            this.cellSize = Integer.parseInt((String) line.elementAt(1));

            // Now we can set the size of the array
            this.raster = new float[this.nCols][this.nRows];

      }

//Read the data of a raster file
private void readData(Vector data, SimpleIO sio)throws Exception{
            // First 6 rows of data (0-5) are header
            int startRow = 6;
```

```
            Vector line;
            String s, valStr;
            //float value;
            float value;
            // Loop through all the data, getting each line from the data file
            // and parseing it
            //Note that since the data are ints but array stores floats
            //they are cast
            for (int j=0; j < nRows; j++){
                  s = (String) data.elementAt(startRow+j);

                  line = sio.parseLine(s," ");

                  for(int i=0; i < nCols; i++){

                        valStr = (String) line.elementAt(i);
                        value = sio.floatInput(valStr);
                        raster[i][j] = value;

                  }
            }
}
//Method to change a raster into a one dimensional array.
//This method was written by Jimmy Batchelor (M.Sc. student 2000),
//and modified by Erin E. Rees.
public float[] toOneD(Raster r){
      float qsArray[];
      int count = 0;
      int nRows, nCols;

      nRows = this.nRows;
      nCols = this.nCols;

      //iterate through the raster
            for (int j=0; j < nRows; j++){
                  for(int i=0; i < nCols; i++){
                        //count all elements if not = -9999
                        if (this.raster[i][j] !=-9999){
                              count++;
                        }
                  }
            }

            int k;
            //set new 1d array size to be equal to the count of all
            //cells != -9999
            qsArray = new float[count];

            k = 0;
            while (k!=count){
                  for (int j=0; j < nRows; j++){
                        for(int i=0; i < nCols; i++){
                              //if the cell contains -9999 don't add to 1d
                              //array
                              qsArray[k] = this.raster[i][j];
                              k++;
                        }
                  }
            }
            //return the converted array
            return qsArray;
      }
```

18.3 The SimpleIO Class

All methods are written by Keith Morrison and Dr. Nick Hulton, unless otherwise specified. Amendments were written by Erin E. Rees:

```
import rspkwm.gis.teaching.*;
import java.io.*;
import java.util.*;

public class SimpleIO {
	// Our instance variables
	/** out Instance variable for the PrintStream class.*/
	PrintStream out;
	/** in Instance variable for the InputStream class.*/
	InputStream in;

	/** Empty constructor.*/
	public SimpleIO() {
		this.out = System.out;
		this.in = System.in;

	}
	/**	Method to write a string to the monitor.*/
	public void write(String s){
		System.out.println(s);
		System.out.flush();
	}

	/**	Method to read an integer from the keyboard.
		@return Integer data type.
		*/
	public int readInt(String s) throws Exception{
		String readLine;
		int i;

		// Use our readKeyBaord method to get the string
		readLine = this.readKeyBoard(s);
		i = Integer.parseInt(readLine);
		return i;
	}

	/**	Method to read a string from the keyboard.
		@return String data type.
		*/
	public String readKeyBoard(String s) throws Exception{
		InputStreamReader isr;
		BufferedReader br;
		String inputLine;
		// We need a stream and a reader to get the data
		isr = new InputStreamReader(System.in);
		br = new BufferedReader(isr);
		this.write(s);
		inputLine = br.readLine();
		return inputLine;
	}

	/**	Method to write a line to a named file.*/
	public void writeFile(String fileName, String s) throws Exception{
		// Variable declarations
		File outputFile;
		FileOutputStream outputStream;
		PrintWriter p;

		// Instansiate the objects we need
```

```
        outputFile = new File(fileName);
        outputStream = new FileOutputStream(outputFile);
        p = new PrintWriter(outputStream);

        // Print a line to the buffer, and flush it to ensure
        // it ends up in the file
        p.println(s);
        p.flush();

        // Close the stream, so we can read from the file if we want
        outputStream.close();
}

/** Method to print all of a file to the screen.*/
public void displayFile(String fileName) throws Exception{
        // Variable declarations
        File inputFile;
        FileInputStream inputStream;
        InputStreamReader inputStreamRead;
        BufferedReader br;
        String readLine;

        // Instansiate the objects we need
        inputFile = new File(fileName);
        inputStream = new FileInputStream(inputFile);
        inputStreamRead = new InputStreamReader(inputStream);
        br = new BufferedReader(inputStreamRead);

        // read all the lines from the file
        while ((readLine = br.readLine()) != null)
              System.out.println(readLine);

        // Close the stream, so we can write to the file if we want
        inputStream.close();
}


/**     Method to read a line from a file.
        @return File line as a String data type.
        */
public String readFile(String fileName) throws Exception{
        // Variable declarations
        File inputFile;
        FileInputStream inputStream;
        InputStreamReader inputStreamRead;
        BufferedReader br;
        String readLine;

        // Instansiate the objects we need
        inputFile = new File(fileName);
        inputStream = new FileInputStream(inputFile);
        inputStreamRead = new InputStreamReader(inputStream);
        br = new BufferedReader(inputStreamRead);

        // read a line from the file
        readLine = br.readLine();

        // Close the stream, so we can write to the file if we want
        inputStream.close();

        // Return that line
        return readLine;
}
```

```
//Method to create an InputStreamReader.  This is necessary for completing
//the process of reading data from a file.

      private InputStreamReader openReader (String fileName) throws Exception{

            // Variable declarations
            File inputFile;

            FileInputStream inputStream;

            // Instansiate the objects we need
            inputFile = new File(fileName);

            inputStream = new FileInputStream(inputFile);

            // return an InputStreamReader
            return new InputStreamReader(inputStream);

      }


      /**   Method to read all the lines from a file.
            @return File lines as a Vector of String objects.
            */
      public Vector getLines (String fileName) throws Exception{

            BufferedReader br;
            String s;
            Vector lines;
            lines = new Vector();   // Create lines Vector
            br = new BufferedReader(openReader(fileName)); //Create br
            s = br.readLine(); // Read a line to s

            while(s!=null){

                  lines.addElement(s);
                  s = br.readLine();

            }
            // No more data to read, return the Vector

            return lines;
      }

      /** Method to parse a given line.
            @return The parsed file line as a Vector of Strings.
            */
      public Vector parseLine(String line, String delim){
            Vector strings = new Vector();
            StringTokenizer st = new StringTokenizer(line, delim);
    while (st.hasMoreTokens()) {
         strings.addElement(st.nextToken());
    }
    // If read no data in we will return a null
            if (strings.isEmpty())
                  strings = null;
            return strings;
      }
```

Methods of the SimpleIO Class written by Erin E. Rees:

```
      /**   Method to read a float from the keyboard.
```

```
        @return float data type.
        */
    public float floatInput(String input) throws Exception{
        //reuse the code from readKeyBoard
        Double d;
        //String s;
        float x;
        //s = this.readKeyBoard(input);
        //convert the string to a double object to a double primitive
        //value
        d = Double.valueOf(input);
        x = (float) d.doubleValue();
        return x;
    }

    /**   Method to read a double (from the keyboard if uncommented),
        * otherwise, returns a double from a String input.
        @return Double data type.
        */
    public double doubleInput(String input) throws Exception{
        //reuse the code from readKeyBoard
        Double d;
        //String s;
        double x;
        //s = this.readKeyBoard(input);
        //convert the string to a double object to a double primitive
        //value
        d = Double.valueOf(input);
        x = d.doubleValue();
        return x;
    }
```

Method of the SimpleIO Class written by Hadeway Sint (M.Sc. student 2002):

```
/** Method to write multiple lines to a named file.*/
public void writeLinesFile(String fileName, Vector lines) throws Exception{

            // Variable declarations
            File outputFile;
            FileOutputStream outputStream;
            PrintWriter p;
            Enumeration e = lines.elements() ;
            String s;

            // Instansiate the objects we need
            outputFile = new File(fileName);
            outputStream = new FileOutputStream(outputFile);
            p = new PrintWriter(outputStream);

            // Print a line to the buffer, and flush it to ensure
            // it ends up in the file
            while (e.hasMoreElements())
            {
                  s = (String) e.nextElement();
            p.println(s);
            }
            p.flush();

            // Close the stream, so we can read from the file if we want
            outputStream.close();
}

}
```

18.4 The Stats Class

The Stats Class was written by Steve Bowley (M.Sc. student 2002) and amendments were made by Erin E. Rees.

```
/** Returns the Standard Deviation of a array of floats
      *     @param float array[]
      *     @return float Standard Deviation of the array
      *
      */
      public static float sDev(float array[])
      {
            //create equation variables
            float sd, a, b, c, d, e;

            c = Stats.mean(array);

            d = c*c;

            //Create an identical array.
            int size = array.length;
            float[] term = new float[size];

            for (int i=0; i<array.length;i++){
                  term[i] = array[i];
            }


            a = Stats.sum(Stats.square(term));

            b = a/array.length;

            e = b - d;

            sd = (float)Math.sqrt(e);

            return sd;

      }

/** Calculates the sum of an array
      *     @param float array[]
      *     @return sum of the array as a float
      */
      public static float sum(float array[])
      {
            //set initial value of sum to 0
            float sum = 0;

            //loop through array
            for(int i = 0; i < array.length; i++)
            {
                  //as long as value in array is not -9999(null value)
                  //add it to the previous sum value
                  if (array[i] != -9999)
                        sum = sum + array[i];
            }

            //return float
            return sum;
      }

/** Calculates the mean of the array and returns it as a float
      *     @param float array[]
```

```
*     @return mean of the array as a float
*/
public static float mean(float array[])
{
      float mean;
      //use sum method from above
      mean = sum(array)/array.length;

      //return float
      return mean;
}
```

19 APPENDIX FOUR

This appendix contains the AML's used to create the Rule Based Model. The AML's are saved as UNIX formatted files; 0k outcome: *dissertation/aml/cat0k.aml*, 9k outcome: *dissertation/aml/cat9k.aml*, 13k outcome: *dissertation/aml/cat0k.aml*, and 18k outcome: *dissertation/aml/cat0k.aml*. Erin E. Rees is the author of all the code. Comments in the code are prefixed with: /*

19.1 The AML used to Produce the 0k Outcome

```
/******************************************************************
/*This code is used to create the Rule Based coverage of wildcat
/*presence as defined by expert knowledge from literature and
/*Dr. Andrew Kitchener.
/*Wildcat preference is ranked from 0 to 3.
/*Rules about highland specifics and the limit of the distribution
/*relative to African precipitation have been added.
/*
/*The AML code to run this code is: &r cat0k
/*
/*Author: Erin E. Rees; August 13, 2002.
/******************************************************************
/*MAIN METHOD
/******************************************************************
/*Command to run the bailout routine if an error occurs.
&severity &error &routine bailout
/*The call order is also the order the grids should be layered.
&call snow
&call afr_rain
&call h0a
&call h0b
&call h0c
&call h2a
&call h2b
&call h3a
&call h3b
&call h4a
&call h4b
&call am
&call af
&call aw
&call bwh
&call bsh
&call cfb
&call csb
&call dfa
&call cfa
&call cwa
&call cfc
&call csa
&call dfd
&call dwd
&call dwa
&call dfbc
&call bsk
&call bwk
&call dwbc
&call et
&call ef
&call merge
```

```
&type MAIN METHOD COMPLETED
&return
/******************************************************************
/*ROUTINES
/******************************************************************
/*Routine to bailout of the AML if an error occurs.
&routine bailout

/*Error message to be displayed if an error occurs:
&setvar errormessage = ERROR in AML - program closing.

/*Print error message to the screen.
&type %errormessage%

/*Close all files.
&call closeFiles

/*Stop the 'cat' AML and return control to the command line.
&stop
/******************************************************************
/*Routine to close all files that are open.
&routine closeFiles
&setvar close = [close -all]
&return
/******************************************************************
/******************************************************************
/******************************************************************
/*Routines to reclassify the environmental grids according to the
/*Koppen Climate Classification scheme.
/******************************************************************
&routine snow
/*Delete grid if it already exists.
&if [exists snow -grid] &then
kill snow

/*Reclassify the grids with the DOCELL command.
docell
/*Snow depth beyond preferred cat levels (i.e. > 30 cm)
 if (tjan < -5 & tjul <= 20) snow = 0
 end
&return
/******************************************************************
&routine afr_rain
/*Delete grid if it already exists.
&if [exists afr_rain -grid] &then
kill afr_rain

/*Reclassify the grids with the DOCELL command.
docell
/*Highland beyond preferred cat levels
 if (africa == 1 & cpre < 365 & pjan < 100 & pjul < 1000) ~
  afr_rain = 0
 end
&return
/******************************************************************
&routine h0a
/*Delete grid if it already exists.
&if [exists h0a -grid] &then
kill h0a

/*Reclassify the grids with the DOCELL command.
docell
/*Highland beyond preferred cat levels
if (europe == 1 & dem > 3000) h0a = 0
```

```
end
&return
/*****************************************************************
&routine h0b
/*Delete grid if it already exists.
&if [exists h0b -grid] &then
kill h0b

/*Reclassify the grids with the DOCELL command.
docell
/*Highland beyond preferred cat levels
if (africa == 1 & dem > 3500) h0b = 0
end
&return
/*****************************************************************
&routine h0c
/*Delete grid if it already exists.
&if [exists h0c -grid] &then
kill h0c

/*Reclassify the grids with the DOCELL command.
docell
/*Highland beyond preferred cat levels
if (asia == 1 & dem > 3000) h0c = 0
end
&return
/*****************************************************************
&routine h2a
/*Delete grid if it already exists.
&if [exists h2a -grid] &then
kill h2a

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (africa == 1 & dem < 2000) h2a = 3
 end
&return
/*****************************************************************
&routine h2b
/*Delete grid if it already exists.
&if [exists h2b -grid] &then
kill h2b

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (asia == 1 & dem < 2000) h2b = 3
 end
&return
/*****************************************************************
&routine h3a
/*Delete grid if it already exists.
&if [exists h3a -grid] &then
kill h3a

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (africa == 1 & dem >= 2000 & dem < 2500) h3a = 2
 end
&return
/*****************************************************************
&routine h3b
```

```
/*Delete grid if it already exists.
&if [exists h3b -grid] &then
kill h3b

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (asia == 1 & dem >= 2000 & dem < 2500) h3b = 2
 end
&return
/*****************************************************************
&routine h4a
/*Delete grid if it already exists.
&if [exists h4a -grid] &then
kill h4a

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (africa == 1 & dem >= 2500 & dem < 3500) h4a = 1
 end
&return
/*****************************************************************
&routine h4b
/*Delete grid if it already exists.
&if [exists h4b -grid] &then
kill h4b

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (asia == 1 & dem >= 2500 & dem < 3000) h4b = 1
 end
&return
/*****************************************************************
&routine am
/*Delete grid if it already exists.
&if [exists am -grid] &then
kill am

/*Reclassify the grids with the DOCELL command.
docell
/*saved as am
/*Am monsoon variety of Af
 if (ctmp > 180 & cpre < 5840 & pjan < 500 ~
  & pjul > 9000) am = 1
 end
&return
/*****************************************************************
&routine aw
/*Delete grid if it already exists.
&if [exists aw -grid] &then
kill aw

/*Reclassify the grids with the DOCELL command.
docell
/*saved as aw
/*Aw tropical savanna
if (ctmp >= 180 & tjan >= 18 & tjul >= 18 ~
 & trange < 10 & (pjan > (1.5 * pjul) or pjul > (1.5 * pjan))) aw = 3
end

&return
/*****************************************************************
```

```
&routine af
/*Delete grid if it already exists.
&if [exists af -grid] &then
kill af

/*Reclassify the grids with the DOCELL command.
docell
/*saved as af
/*Af tropical rainforest
if (ctmp >= 180 & pjan >= 18 & pjul >= 18 & trange <= 7 & pjul > 400 ~
 & pjan > 500 & cpre > 1000) af = 0
end
&return
/*****************************************************************
&routine bsh
/*Delete grid if it already exists.
&if [exists bsh -grid] &then
kill bsh

/*Reclassify the grids with the DOCELL command.
docell
/*save as bsh
/*BSh dry-hot steppe climate
if ((latitude >= 0 & tjul >= 22 & tjul < 35 & tjan >= 3 & ~
 ctmp > 35 & pjan < 6000 & pjul < 4500 & (pjul + pjan) > 450 & ~
 cpre < 5000) ~
 or (latitude < 0 & tjul >= 7 & tjan >= 22 & ~
 pjul < 1500 & pjan < 4500 & trange <= 17 & (pjul + pjan) > 450)) ~
 bsh = 3
end
&return
/*****************************************************************
&routine bwh
/*Delete grid if it already exists.
&if [exists bwh -grid] &then
kill bwh

/*Reclassify the grids with the DOCELL command.
docell
/*save as bwh
/*BWh dry-hot desert climate
if (tjan > 0 & tjul > 0 & pjan < 1500 & pjul < 1500 & cpre < 4000 ~
) bwh = 2
end
&return
/*****************************************************************
&routine csb
/*Delete grid if it already exists.
&if [exists csb -grid] &then
kill csb

/*Reclassify the grids with the DOCELL command.
docell
/*save csb
/*Rainy climates with mild winters;dry winters;warmest month < 22
if ((latitude >= 0 & tjan > 0 & tjan <= 10 & tjul <= 23 & pjan <= 10000 ~
 & pjul < 1500 & dem < 1000 & cont < 1 & trange < 20) ~
 or (latitude < 0 & tjul > 0 & tjul <= 10 & tjan <= 23 & pjul <= 10000 ~
 & pjan < 1500 & dem < 1000 & cont < 1 & trange < 20)) csb = 2
end
&return
/*****************************************************************
&routine cfb
/*Delete grid if it already exists.
```

```
&if [exists cfb -grid] &then
kill cfb

/*Reclassify the grids with the DOCELL command.
docell
/*save as cfb
/*Rainy climates with mild winters;constantly moist;warmest month < 22
if ((latitude >= 0 & tjan >= -3 & tjan < 18 & tjul > 10 & ~
 tjul <= 22 & pjan >= 60 & pjul >= 60) ~
 or (latitude < 0 & tjul >= -3 & tjul < 18 & tjan > 10 & ~
 tjan <= 22 & pjul >= 60 & pjan >= 60)) cfb = 3
end
&return
/*******************************************************************
&routine dfa
/*Delete grid if it already exists.
&if [exists dfa -grid] &then
kill dfa

/*Reclassify the grids with the DOCELL command.
docell
/*save as dfa
/*Rainy climates with severe winters; constantly moist; warmest month > 22
if ((latitude >= 0 & tjan >= -5 & tjan < 0 & tjul >= 22 & pjan >= 2000 ~
 & pjul >= 1900 & trange < 30) ~
 or (latitude < 0 & tjul >= -5 & tjul < 0 & tjan >= 22 & pjul >= 2000 ~
 & pjan >= 1900 & trange < 30)) dfa = 1
end
&return
/*******************************************************************
&routine cfa
/*Delete grid if it already exists.
&if [exists cfa -grid] &then
kill cfa

/*Reclassify the grids with the DOCELL command.
docell
/*save as cfa
/*Rainy climates with mild winters;constantly moist;warmest month > 22;
if ((latitude >= 0 & tjan >= -3 & tjul >= 22 & trange >= 18 & trange < 30 ~
 & cpre > 1000 & cpre < 13000 & pjan > 400 & pjul > 1000) ~
 or (latitude <= 0 & tjul >= -3 & tjan >= 22 & trange >= 5 ~
 & cpre > 1000 & cpre < 17000 & pjul > 600 & pjan > 1800 & pjan > ~
 (1.5 * pjul))) cfa = 2
end
&return
/*******************************************************************
&routine cwa
/*Delete grid if it already exists.
&if [exists cwa -grid] &then
kill cwa

/*Reclassify the grids with the DOCELL command.
docell
/*save as cwa
/*Rainy climates with mild winters;dry winters;warmest month > 22;
if ((latitude >= 0 & tjan >= -3 & tjul > 15 & ctmp < 230 & trange > 3 ~
& pjan < 1000 & pjul > 1000 & cpre > 300) ~
or (latitude < 0 & tjul >= -3 & tjul <= 19 & tjan > 15 & trange > 2 & ~
trange < 14 & pjan > 3500 & pjul < 500)) cwa = 2
end
&return
/*******************************************************************
&routine cfc
```

```
/*Delete grid if it already exists.
&if [exists cfc -grid] &then
kill cfc

/*Reclassify the grids with the DOCELL command.
docell
/*save as cfc
/*Rainy climates with mild winters;constantly moist;<4 months > 10 degrees;
if ((latitude >= 0 & tjan >= -3 & tjan < 10 & tjul < 10 & pjan >= 60 & ~
 pjul >= 60 & dem < 1000) ~
 or (latitude < 0 & tjul >= -3 & tjul < 10 & tjan < 15 & ~
 pjul >= 60 & pjan >= 60 & dem < 1000)) cfc = 2
end
&return
/******************************************************************
&routine csa
/*Delete grid if it already exists.
&if [exists csa -grid] &then
kill csa

/*Reclassify the grids with the DOCELL command.
docell
/*save csa
/*Rainy climates with mild winters;dry summers;warmest month > 22
if ((latitude >= 0 & tjan > -3 & tjan < 18 & tjul > 18 & ~
 pjul < 500 & pjan > 1000) ~
 or (latitude >= 0 & tjul > -3 & tjul < 18 & tjan > 18 & ~
 pjan < 500 & pjul > 1000)) csa = 2
end
&return
/******************************************************************
&routine dfd
/*Delete grid if it already exists.
&if [exists dfd -grid] &then
kill dfd

/*Reclassify the grids with the DOCELL command.
docell
/*save as dfd
/*Rainy climates with severe winters; constantly moist;coldest month < -38
if (latitude >= 0 & tjan <= -38 & tjul > 10 & (4 * pjan) >= pjul) dfd = 0
end
&return
/******************************************************************
&routine dwd
/*Delete grid if it already exists.
&if [exists dwd -grid] &then
kill dwd

/*Reclassify the grids with the DOCELL command.
docell
/*save as dwd
/*Rainy climates with severe winters; dry winters; coldest month < -38
if ((latitude >= 0 & (4 * pjan) < pjul & tjan <= -38) ~
 or (latitude < 0 & (4 * pjul) < pjan & tjul <= -38)) dwd = 0
end
&return
/******************************************************************
&routine dfbc
/*Delete grid if it already exists.
&if [exists dfbc -grid] &then
kill dfbc

/*Reclassify the grids with the DOCELL command.
```

```
docell
/*save as dfbc
/*Rainy climates with severe winters; constantly moist; warmest month < 22 (b)
/*Rainy climates with severe winters; constantly moist;>4 months > 10 degrees
(c)
if ((latitude >= 0 & tjan > -38 & tjan < -3 & tjul < 22 & pjul > 1500 & ~
 pjan > 500) ~
 or (latitude < 0 & tjul > -38 & tjul < -3 & tjan < 22 & pjan > 1500 & ~
 pjul > 500)) dfbc = 1
end
&return
/*******************************************************************
&routine bsk
/*Delete grid if it already exists.
&if [exists bsk -grid] &then
kill bsk

/*Reclassify the grids with the DOCELL command.
docell
/*save as bsk
/*BSk dry-cold steppe climate
if ((latitude >= 0 & tjan <= 1 & tjul > 10 & tjul < 24 ~
 & pjan <= 2000 & pjul <= 2000) ~
 or (latitude < 0 & tjul <= 1 & tjan > 10 & tjan < 24 ~
 & pjul <= 2000 & pjul <= 2000)) bsk = 2
end
&return
/*******************************************************************
&routine bwk
/*Delete grid if it already exists.
&if [exists bwk -grid] &then
kill bwk

/*Reclassify the grids with the DOCELL command.
docell
/*save as bwk
/*BWk dry-cold desert climate
if ((latitude >= 0 & tjul >= 18 & ctmp > -150 & tjan < 5 & ~
 tjan > -20 & cpre < 4000 & pjul < 1000 & pjan < 1000) ~
 or (latitude < 0 & tjan >= 18 & ctmp > -150 & tjul < 5 & ~
 tjul > -20 & cpre < 4000 & pjan < 1000 & pjan < 1000)) bwk = 2
end
&return
/*******************************************************************
&routine dwbc
/*Delete grid if it already exists.
&if [exists dwbc -grid] &then
kill dwbc

/*Reclassify the grids with the DOCELL command.
docell
/*save as dwbc
/*Rainy climates with severe winters; dry winters; warmest month < 22
if ((latitude >= 0 & tjul > 10 & tjul < 22 & tjan < -3 ~
 & (4 * pjan) < pjul) ~
 or (latitude < 0 & tjan > 10 & tjan < 22 & tjul < -3 ~
 & (4 * pjul) < pjan)) dwbc = 1
end
&return
/*******************************************************************
&routine dwa
/*Delete grid if it already exists.
&if [exists dwa -grid] &then
kill dwa
```

```
/*Reclassify the grids with the DOCELL command.
docell
/*save as dwa
/*Rainy climates with severe winters; dry winters; warmest month > 22
if ((latitude >= 0 & pjan < 500 & pjul > 2000 & tjan < -3 & tjul > 22) ~
or (latitude < 0 & pjul < 500 & pjan > 2000 & tjul < -3 & tjan > 22)) dwa = 1
end
&return
/*******************************************************************
&routine et
/*Delete grid if it already exists.
&if [exists et -grid] &then
kill et

/*Reclassify the grids with the DOCELL command.
docell
/*save as et
/*Polar climates with tundra
if ((latitude >= 0 & tjul > 0 & tjul < 10 ) ~
 or (latitude < 0 & tjan > 0 & tjan < 10 )) et = 0
end
&return
/*******************************************************************
&routine ef
/*Delete grid if it already exists.
&if [exists ef -grid] &then
kill ef

/*Reclassify the grids with the DOCELL command.
docell
/*save as ef
/*Polar climates with perpetual frost
if (tjan < 0 & tjul < 0 & ctmp < 0) ef = 0
end
&return
/*******************************************************************
&routine merge
/*Delete grid if it already exists.
&if [exists cat0k -grid] &then
kill cat0k

/*Routine to combine all the preference grids together.  Order of
/*grids matters for this operation.  Preceding grid values will
/*be preserved over succeeding grid values that overlap.
cat0k = merge(snow,afr_rain,h0a,h0b,h0c,am,af,aw,bsh,bwh,cfb,csb,dfa,cfa,~
cwa,cfc,csa,dfd,dwd,dwa,dfbc,bsk,bwk,dwbc,et,ef,h2a,h2b,h3a,h3b,~
h4a,h4b,nodata)

&return
/*******************************************************************
```

19.2 The AML used to Produce the 9k Outcome

```
/******************************************************************
/*This code is used to create the Rule Based coverage of wildcat
/*presence as defined by expert knowledge from literature and
/*Dr. Andrew Kitchener for 9,000 years ago.
/*Wildcat preference is ranked from 0 to 3.
/*Rules about highland specifics and the limit of the distribution
/*relative to African precipitation have been added.
/*
/*The AML code to run this code is: &r cat9k
/*
```

```
/*Author: Erin E. Rees; August 13, 2002.
/*****************************************************************
/*MAIN METHOD
/*****************************************************************
/*Command to run the bailout routine if an error occurs.
&severity &error &routine bailout
/*The call order is also the order the grids should be layered.
&call snow
&call afr_rain
&call h0a
&call h0b
&call h0c
&call h2a
&call h2b
&call h3a
&call h3b
&call h4a
&call h4b
&call am
&call af
&call aw
&call bwh
&call bsh
&call csb
&call dfa
&call cfa
&call cwa
&call cfc
&call csa
&call dfd
&call dwd
&call dwa
&call dfbc
&call bsk
&call bwk
&call dwbc
&call et
&call ef
&call topo9
&call merge
&type MAIN METHOD COMPLETED
&return
/*****************************************************************
/*ROUTINES
/*****************************************************************
/*Routine to bailout of the AML if an error occurs.
&routine bailout

/*Error message to be displayed if an error occurs:
&setvar errormessage = ERROR in AML - program closing.

/*Print error message to the screen.
&type %errormessage%

/*Close all files.
&call closeFiles

/*Stop the 'cat' AML and return control to the command line.
&stop
/******************************************************************
/*Routine to close all files that are open.
&routine closeFiles
&setvar close = [close -all]
&return
```

```
/*************************************************************
/*************************************************************
/*************************************************************
/*Routines to reclassify the environmental grids according to the
/*Koppen Climate Classification scheme.
/*************************************************************
&routine snow
/*Delete grid if it already exists.
&if [exists snow -grid] &then
kill snow

/*Reclassify the grids with the DOCELL command.
docell
/*Snow depth beyond preferred cat levels (i.e. > 30 cm)
 if (tjan2up2 < -5 & tjul2up2 <= 20) snow = 0
 end
&return
/*************************************************************
&routine afr_rain
/*Delete grid if it already exists.
&if [exists afr_rain -grid] &then
kill afr_rain

/*Reclassify the grids with the DOCELL command.
docell
/*Highland beyond preferred cat levels
 if (africa == 1 & cpre < 365 & pjan < 100 & pjul < 1000) ~
   afr_rain = 0
 end
&return
/*************************************************************
&routine h0a
/*Delete grid if it already exists.
&if [exists h0a -grid] &then
kill h0a

/*Reclassify the grids with the DOCELL command.
docell
/*Highland beyond preferred cat levels
if (europe == 1 & dem > 3000) h0a = 0
end
&return
/*************************************************************
&routine h0b
/*Delete grid if it already exists.
&if [exists h0b -grid] &then
kill h0b

/*Reclassify the grids with the DOCELL command.
docell
/*Highland beyond preferred cat levels
if (africa == 1 & dem > 3500) h0b = 0
end
&return
/*************************************************************
&routine h0c
/*Delete grid if it already exists.
&if [exists h0c -grid] &then
kill h0c

/*Reclassify the grids with the DOCELL command.
docell
/*Highland beyond preferred cat levels
if (asia == 1 & dem > 3000) h0c = 0
```

```
end
&return
/*************************************************************
&routine h2a
/*Delete grid if it already exists.
&if [exists h2a -grid] &then
kill h2a

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (africa == 1 & dem < 2000) h2a = 3
 end
&return
/*************************************************************
&routine h2b
/*Delete grid if it already exists.
&if [exists h2b -grid] &then
kill h2b

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (asia == 1 & dem < 2000) h2b = 3
 end
&return
/*************************************************************
&routine h3a
/*Delete grid if it already exists.
&if [exists h3a -grid] &then
kill h3a

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (africa == 1 & dem >= 2000 & dem < 2500) h3a = 2
 end
&return
/*************************************************************
&routine h3b
/*Delete grid if it already exists.
&if [exists h3b -grid] &then
kill h3b

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (asia == 1 & dem >= 2000 & dem < 2500) h3b = 2
 end
&return
/*************************************************************
&routine h4a
/*Delete grid if it already exists.
&if [exists h4a -grid] &then
kill h4a

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (africa == 1 & dem >= 2500 & dem < 3500) h4a = 1
 end
&return
/*************************************************************
&routine h4b
```

```
/*Delete grid if it already exists.
&if [exists h4b -grid] &then
kill h4b

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (asia == 1 & dem >= 2500 & dem < 3000) h4b = 1
 end
&return
/******************************************************************
&routine am
/*Delete grid if it already exists.
&if [exists am -grid] &then
kill am

/*Reclassify the grids with the DOCELL command.
docell
/*saved as am
/*Am monsoon variety of Af
 if (ctmp2up > 180 & cpre < 5840 & pjan < 500 ~
  & pjul > 9000) am = 0
 end
&return
/******************************************************************
&routine aw
/*Delete grid if it already exists.
&if [exists aw -grid] &then
kill aw

/*Reclassify the grids with the DOCELL command.
docell
/*saved as aw
/*Aw tropical savanna
if (ctmp2up >= 180 & tjan2up2 >= 18 & tjul2up2 >= 18 ~
 & trange < 10 & (pjan > (1.5 * pjul) or pjul > (1.5 * pjan))) aw = 3
end

&return
/******************************************************************
&routine af
/*Delete grid if it already exists.
&if [exists af -grid] &then
kill af

/*Reclassify the grids with the DOCELL command.
docell
/*saved as af
/*Af tropical rainforest
if (ctmp2up >= 180 & pjan >= 18 & pjul >= 18 & trange <= 7 & ~
 pjul > 400 & pjan > 500 & cpre > 1000) af = 0
end
&return
/******************************************************************
&routine bsh
/*Delete grid if it already exists.
&if [exists bsh -grid] &then
kill bsh

/*Reclassify the grids with the DOCELL command.
docell
/*save as bsh
/*BSh dry-hot steppe climate
if ((latitude >= 0 & tjul2up2 >= 22 & tjul2up2 < 35 & tjan2up2 >= 3 ~
```

```
 & ctmp2up > 35 & pjan < 6000 & pjul < 4500 & (pjul + pjan) > 450 & ~
 cpre < 5000) or (latitude < 0 & tjul2up2 >= 7 & tjan2up2 >= 22 & ~
 pjul < 1500 & pjan < 4500 & trange <= 17 & (pjul + pjan) > 450)) ~
 bsh = 3
end
&return
/*****************************************************************
&routine bwh
/*Delete grid if it already exists.
&if [exists bwh -grid] &then
kill bwh

/*Reclassify the grids with the DOCELL command.
docell
/*save as bwh
/*BWh dry-hot desert climate
/*Formula for cpre when precip is even throughout the year:
/* cpre_f (in cm) = ctmp2up (celsius) + 7
if (tjan2up2 > 0 & tjul2up2 > 0 & pjan < 1500 & pjul < 1500 & ~
 cpre < 4000) bwh = 2
end
&return
/*****************************************************************
&routine csb
/*Delete grid if it already exists.
&if [exists csb -grid] &then
kill csb

/*Reclassify the grids with the DOCELL command.
docell
/*save csb
/*Rainy climates with mild winters;dry winters;warmest month < 22
if ((latitude >= 0 & tjan2up2 > 0 & tjan2up2 <= 10 & tjul2up2 <=  ~
 23 & pjan <= 10000 & pjul < 1500 & dem < 1000 & grid9fff < 1 & ~
 trange < 20) or (latitude < 0 & tjul2up2 > 0 & tjul2up2 <= 10 & ~
 tjan2up2 <= 23 & pjul <= 10000 & pjan < 1500 & dem < 1000 & ~
 grid9fff < 1 & trange < 20)) csb = 2
end
&return
/*****************************************************************
&routine cfb
/*Delete grid if it already exists.
&if [exists cfb -grid] &then
kill cfb

/*Reclassify the grids with the DOCELL command.
docell
/*save as cfb
/*Rainy climates with mild winters;constantly moist;warmest month < 22
if ((latitude >= 0 & tjan2up2 >= -3 & tjan2up2 < 18 & tjul2up2 > ~
 10 & tjul2up2 <= 22 & pjan >= 60 & pjul >= 60)
 or (latitude < 0 & tjul2up2 >= -3 & tjul2up2 < 18 & tjan2up2 > ~
 10 & tjan2up2 <= 22 & pjul >= 60 & pjan >= 60)) cfb = 3
end
&return
/*****************************************************************
&routine dfa
/*Delete grid if it already exists.
&if [exists dfa -grid] &then
kill dfa

/*Reclassify the grids with the DOCELL command.
docell
/*save as dfa
```

```
/*Rainy climates with severe winters; constantly moist; warmest month > 22
if ((latitude >= 0 & tjan2up2 >= -5 & tjan2up2 < 0 & tjul2up2 >= ~
 22 & pjan >= 2000 & pjul >= 1900 & trange < 30) ~
 or (latitude < 0 & tjul2up2 >= -5 & tjul2up2 < 0 & tjan2up2 >= ~
 22 & pjul >= 2000 & pjan >= 1900 & trange < 30)) dfa = 1
end
&return
/******************************************************************
&routine cfa
/*Delete grid if it already exists.
&if [exists cfa -grid] &then
kill cfa

/*Reclassify the grids with the DOCELL command.
docell
/*save as cfa
/*Rainy climates with mild winters;constantly moist;warmest month > 22;
if ((latitude >= 0 & tjan2up2 >= -3 & tjul2up2 >= 22 & trange >=  ~
 18 & trange < 30 & cpre > 1000 & cpre < 13000 & pjan > 400 & ~
 pjul > 1000) ~
 or (latitude < 0 & tjul2up2 >= -3 & tjan2up2 >= 22 & trange >= 5 ~
 & cpre > 1000 & cpre < 17000 & pjul > 600 & pjan > 1800 & pjan > ~
 (1.5 * pjul))) cfa = 2
end
&return
/******************************************************************
&routine cwa
/*Delete grid if it already exists.
&if [exists cwa -grid] &then
kill cwa

/*Reclassify the grids with the DOCELL command.
docell
/*save as cwa
/*Rainy climates with mild winters;dry winters;warmest month > 22;
if ((latitude >= 0 & tjan2up2 >= -3 & tjul2up2 > 15 & ctmp2up < ~
 230 & trange > 3 & pjan < 1000 & pjul > 1000 & cpre > 300) ~
 or (latitude < 0 & tjul2up2 >= -3 & tjul2up2 <= 19 & tjan2up2 > ~
 15 & trange > 2 & trange < 14 & pjan > 3500 & pjul < 500)) cwa = 2
end
&return
/******************************************************************
&routine cfc
/*Delete grid if it already exists.
&if [exists cfc -grid] &then
kill cfc

/*Reclassify the grids with the DOCELL command.
docell
/*save as cfc
/*Rainy climates with mild winters;constantly moist;<4 months > 10 degrees;
if ((latitude >= 0 & tjan2up2 >= -3 & tjan2up2 < 10 & tjul2up2 < ~
 10 & pjan >= 60 & pjul >= 60 & dem < 1000) ~
 or (latitude < 0 & tjul2up2 >= -3 & tjul2up2 < 10 & tjan2up2 < 15 & ~
 pjul >= 60 & pjan >= 60 & dem < 1000)) cfc = 2
end
&return
/******************************************************************
&routine csa
/*Delete grid if it already exists.
&if [exists csa -grid] &then
kill csa

/*Reclassify the grids with the DOCELL command.
```

```
docell
/*save csa
/*Rainy climates with mild winters;dry summers;warmest month > 22
if ((latitude >= 0 & tjan2up2 > -3 & tjan2up2 < 18 & tjul2up2 > 18 & ~
 pjul < 500 & pjan > 1000) ~
 or (latitude < 0 & tjul2up2 > -3 & tjul2up2 < 18 & tjan2up2 > 18 & ~
 pjan < 500 & pjul > 1000)) csa = 1
end
&return
/*****************************************************************
&routine dfd
/*Delete grid if it already exists.
&if [exists dfd -grid] &then
kill dfd

/*Reclassify the grids with the DOCELL command.
docell
/*save as dfd
/*Rainy climates with severe winters; constantly moist;coldest month < -38
if (latitude >= 0 & tjan2up2 <= -38 & tjul2up2 > 10 & (4 * pjan) >= ~
 pjul) dfd = 0
end
&return
/*****************************************************************
&routine dwd
/*Delete grid if it already exists.
&if [exists dwd -grid] &then
kill dwd

/*Reclassify the grids with the DOCELL command.
docell
/*save as dwd
/*Rainy climates with severe winters; dry winters; coldest month < -38
if ((latitude >= 0 & (4 * pjan) < pjul & tjan2up2 <= -38) ~
 or (latitude < 0 & (4 * pjul) < pjan & tjul2up2 <= -38)) dwd = 0
/*SEEMS GOOD
end
&return
/*****************************************************************
&routine dfbc
/*Delete grid if it already exists.
&if [exists dfbc -grid] &then
kill dfbc

/*Reclassify the grids with the DOCELL command.
docell
/*save as dfbc
/*Rainy climates with severe winters; constantly moist; warmest month < 22 (b)
/*Rainy climates with severe winters; constantly moist;>4 months > 10 degrees
(c)
if ((latitude >= 0 & tjan2up2 > -38 & tjan2up2 < -3 & tjul2up2 < 22 ~
 & pjul > 1500 & pjan > 500) ~
 or (latitude < 0 & tjul2up2 > -38 & tjul2up2 < -3 & tjan2up2 < 22 & ~
 pjan > 1500 & pjul > 500)) dfbc = 1
end
&return
/*****************************************************************
&routine bsk
/*Delete grid if it already exists.
&if [exists bsk -grid] &then
kill bsk

/*Reclassify the grids with the DOCELL command.
docell
```

```
/*save as bsk
/*BSk dry-cold steppe climate
if ((latitude >= 0 & tjan2up2 <= 1 & tjul2up2 > 10 & tjul2up2 < 24 ~
 & pjan <= 2000 & pjul <= 2000) ~
 or (latitude < 0 & tjul2up2 <= 1 & tjan2up2 > 10 & tjan2up2 < 24 ~
 & pjul <= 2000 & pjul <= 2000)) bsk = 2
end
&return
/*****************************************************************
&routine bwk
/*Delete grid if it already exists.
&if [exists bwk -grid] &then
kill bwk

/*Reclassify the grids with the DOCELL command.
docell
/*save as bwk
/*BWk dry-cold desert climate
if ((latitude >= 0 & tjul2up2 >= 18 & ctmp2up > -150 & tjan2up2 < ~
 5 & tjan2up2 > -20 & cpre < 4000 & pjul < 1000 & pjan < 1000) ~
 or (latitude < 0 & tjan2up2 >= 18 & ctmp2up > -150 & tjul2up2 < ~
 5 & tjul2up2 > -20 & cpre < 4000 & pjan < 1000 & pjan < 1000)) bwk = 2
end
&return
/*****************************************************************
&routine dwbc
/*Delete grid if it already exists.
&if [exists dwbc -grid] &then
kill dwbc

/*Reclassify the grids with the DOCELL command.
docell
/*save as dwbc
/*Rainy climates with severe winters; dry winters; warmest month < 22
if ((latitude >= 0 & tjul2up2 > 10 & tjul2up2 < 22 & tjan2up2 < -3 ~
 & (4 * pjan) < pjul) ~
 or (latitude < 0 & tjan2up2 > 10 & tjan2up2 < 22 & tjul2up2 < -3 ~
 & (4 * pjul) < pjan)) dwbc = 1
end
&return
/*****************************************************************
&routine dwa
/*Delete grid if it already exists.
&if [exists dwa -grid] &then
kill dwa

/*Reclassify the grids with the DOCELL command.
docell
/*save as dwa
/*Rainy climates with severe winters; dry winters; warmest month > 22
if ((latitude >= 0 & pjan < 500 & pjul > 2000 & tjan2up2 < -3 & ~
 tjul2up2 > 22) ~
 or (latitude < 0 & pjul < 500 & pjan > 2000 & tjul2up2 < -3 & ~
 tjan2up2 > 22)) dwa = 1
end
&return
/*****************************************************************
&routine et
/*Delete grid if it already exists.
&if [exists et -grid] &then
kill et

/*Reclassify the grids with the DOCELL command.
docell
```

```
/*save as et
/*Polar climates with tundra
if ((latitude >= 0 & tjul2up2 > 0 & tjul2up2 < 10 ) ~
 or (latitude < 0 & tjan2up2 > 0 & tjan2up2 < 10 )) et = 0
end
&return
/*******************************************************************
&routine ef
/*Delete grid if it already exists.
&if [exists ef -grid] &then
kill ef

/*Reclassify the grids with the DOCELL command.
docell
/*save as ef
/*Polar climates with perpetual frost
if (tjan2up2 < 0 & tjul2up2 < 0 & ctmp2up < 0) ef = 0
end
&return
/*******************************************************************
&routine topo9
/*Delete grid if it already exists.
&if [exists topo9 -grid] &then
kill topo9

/*Reclassify the topographic grid with the DOCELL command.
docell
if (grid9f == 1) topo9 = 9
end
&return
/*******************************************************************
&routine merge
/*Delete grid if it already exists.
&if [exists cat9k -grid] &then
kill cat9k

/*Routine to combine all the preference grids together.  Order of
/*grids matters for this operation.  Preceding grid values will
/*be preserved over succeeding grid values that overlap.
cat9k = merge(snow,afr_rain,h0a,h0b,h0c,am,af,aw,bsh,bwh,cfb,~
csb,dfa,cfa,cwa,cfc,csa,dfd,dwd,dwa,dfbc,bsk,bwk,dwbc,et,ef,h2a,~
h2b,h3a,h3b,h4a,h4b,topo9)
&return
/*******************************************************************
```

19.3 The AML used to Produce the 13k Outcome

```
/******************************************************************
/*This code is used to create the Rule Based coverage of wildcat
/*presence as defined by expert knowledge from literature and
/*Dr. Andrew Kitchener for 13,000 years ago.
/*Wildcat preference is ranked from 0 to 3.
/*Rules about highland specifics and the limit of the distribution
/*relative to African precipitation have been added.
/*
/*The AML code to run this code is: &r cat13k
/*
/*Author: Erin E. Rees; August 13, 2002.
/******************************************************************
/*MAIN METHOD
/******************************************************************
/*Command to run the bailout routine if an error occurs.
&severity &error &routine bailout
/*The call order is also the order the grids should be layered.
&call snow
```

```
&call afr_rain
&call h0a
&call h0b
&call h0c
&call h2a
&call h2b
&call h3a
&call h3b
&call h4a
&call h4b
&call am
&call af
&call aw
&call bwh
&call bsh
&call csb
&call dfa
&call cfa
&call cwa
&call cfc
&call csa
&call dfd
&call dwd
&call dwa
&call dfbc
&call bsk
&call bwk
&call dwbc
&call et
&call ef
&call topo13
&call merge
&type MAIN METHOD COMPLETED
&return
/*****************************************************************
/*ROUTINES
/*****************************************************************
/*Routine to bailout of the AML if an error occurs.
&routine bailout

/*Error message to be displayed if an error occurs:
&setvar errormessage = ERROR in AML - program closing.

/*Print error message to the screen.
&type %errormessage%

/*Close all files.
&call closeFiles

/*Stop the 'cat' AML and return control to the command line.
&stop
/******************************************************************
/*Routine to close all files that are open.
&routine closeFiles
&setvar close = [close -all]
&return
/******************************************************************
/******************************************************************
/******************************************************************
/*Routines to reclassify the environmental grids according to the
/*Koppen Climate Classification scheme.
/******************************************************************
&routine snow
/*Delete grid if it already exists.
```

```
&if [exists snow -grid] &then
kill snow

/*Reclassify the grids with the DOCELL command.
docell
/*Snow depth beyond preferred cat levels (i.e. > 30 cm)
 if (tjan2down < -5 & tjul2down <= 20) snow = 0
 end
&return
/*****************************************************************
&routine afr_rain
/*Delete grid if it already exists.
&if [exists afr_rain -grid] &then
kill afr_rain

/*Reclassify the grids with the DOCELL command.
docell
/*Highland beyond preferred cat levels
 if (africa == 1 & cpre < 365 & pjan < 100 & pjul < 1000) ~
   afr_rain = 0
 end
&return
/*****************************************************************
&routine h
/*Delete grid if it already exists.
&if [exists h -grid] &then
kill h

/*Reclassify the grids with the DOCELL command.
docell
/*Highland beyond preferred cat levels
 if (dem > 3500) h = 0
 end
&return
/*****************************************************************
&routine h0a
/*Delete grid if it already exists.
&if [exists h0a -grid] &then
kill h0a

/*Reclassify the grids with the DOCELL command.
docell
/*Highland beyond preferred cat levels
if (europe == 1 & dem > 3000) h0a = 0
end
&return
/*****************************************************************
&routine h0b
/*Delete grid if it already exists.
&if [exists h0b -grid] &then
kill h0b

/*Reclassify the grids with the DOCELL command.
docell
/*Highland beyond preferred cat levels
if (africa == 1 & dem > 3500) h0b = 0
end
&return
/*****************************************************************
&routine h0c
/*Delete grid if it already exists.
&if [exists h0c -grid] &then
kill h0c
```

```
/*Reclassify the grids with the DOCELL command.
docell
/*Highland beyond preferred cat levels
if (asia == 1 & dem > 3000) h0c = 0
end
&return
/*******************************************************************
&routine h2a
/*Delete grid if it already exists.
&if [exists h2a -grid] &then
kill h2a

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (africa == 1 & dem < 2000) h2a = 3
 end
&return
/*******************************************************************
&routine h2b
/*Delete grid if it already exists.
&if [exists h2b -grid] &then
kill h2b

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (asia == 1 & dem < 2000) h2b = 3
 end
&return
/*******************************************************************
&routine h3a
/*Delete grid if it already exists.
&if [exists h3a -grid] &then
kill h3a

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (africa == 1 & dem >= 2000 & dem < 2500) h3a = 2
 end
&return
/*******************************************************************
&routine h3b
/*Delete grid if it already exists.
&if [exists h3b -grid] &then
kill h3b

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (asia == 1 & dem >= 2000 & dem < 2500) h3b = 2
 end
&return
/*******************************************************************
&routine h4a
/*Delete grid if it already exists.
&if [exists h4a -grid] &then
kill h4a

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (africa == 1 & dem >= 2500 & dem < 3500) h4a = 1
```

```
 end
&return
/*****************************************************************
&routine h4b
/*Delete grid if it already exists.
&if [exists h4b -grid] &then
kill h4b

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (asia == 1 & dem >= 2500 & dem < 3000) h4b = 1
 end
&return
/*****************************************************************
&routine am
/*Delete grid if it already exists.
&if [exists am -grid] &then
kill am

/*Reclassify the grids with the DOCELL command.
docell
/*saved as am
/*Am monsoon variety of Af
 if (ctmp2down > 180 & cpre < 5840 & pjan < 500 ~
  & pjul > 9000) am = 0
 end
&return
/*****************************************************************
&routine aw
/*Delete grid if it already exists.
&if [exists aw -grid] &then
kill aw

/*Reclassify the grids with the DOCELL command.
docell
/*saved as aw
/*Aw tropical savanna
if (ctmp2down >= 180 & tjan2down >= 18 & tjul2down >= 18 ~
 & trange < 10 & (pjan > (1.5 * pjul) or pjul > (1.5 * pjan))) aw = 3
end

&return
/*****************************************************************
&routine af
/*Delete grid if it already exists.
&if [exists af -grid] &then
kill af

/*Reclassify the grids with the DOCELL command.
docell
/*saved as af
/*Af tropical rainforest
if (ctmp2down >= 180 & pjan >= 18 & pjul >= 18 & trange <= 7 & pjul > 400 ~
& pjan > 500 & cpre > 1000) af = 0
end
&return
/*****************************************************************
&routine bsh
/*Delete grid if it already exists.
&if [exists bsh -grid] &then
kill bsh

/*Reclassify the grids with the DOCELL command.
```

```
docell
/*save as bsh
/*BSh dry-hot steppe climate
if ((latitude >= 0 & tjul2down >= 22 & tjul2down < 35 & tjan2down >= 3 ~
 & ctmp2down > 35 & pjan < 6000 & pjul < 4500 & (pjul + pjan) > 450 ~
 & cpre < 5000) ~
 or (latitude < 0 & tjul2down >= 7 & tjan2down >= 22 & ~
 pjul < 1500 & pjan < 4500 & trange <= 17 & (pjul + pjan) > 450)) bsh = 3
end
&return
/*******************************************************************
&routine bwh
/*Delete grid if it already exists.
&if [exists bwh -grid] &then
kill bwh

/*Reclassify the grids with the DOCELL command.
docell
/*save as bwh
/*BWh dry-hot desert climate
/*Formula for cpre when precip is even throughout the year:
/* cpre_f (in cm) = ctmp2down (celsius) + 7
if (tjan2down > 0 & tjul2down > 0 & pjan < 1500 & pjul < 1500 & cpre < 4000 ~
) bwh = 2
end
&return
/*******************************************************************
&routine csb
/*Delete grid if it already exists.
&if [exists csb -grid] &then
kill csb

/*Reclassify the grids with the DOCELL command.
docell
/*save csb
/*Rainy climates with mild winters;dry winters;warmest month < 22
if ((latitude >= 0 & tjan2down > 0 & tjan2down <= 10 & tjul2down <= ~
 23 & pjan <= 10000 & pjul < 1500 & dem < 1000 & grid9fff < 1 & trange < 20) ~
 or (latitude < 0 & tjul2down > 0 & tjul2down <= 10 & tjan2down <= ~
 23 & pjul <= 10000 & pjan < 1500 & dem < 1000 & grid9fff < 1 & trange ~
 < 20)) csb = 2
end
&return
/*******************************************************************
&routine cfb
/*Delete grid if it already exists.
&if [exists cfb -grid] &then
kill cfb

/*Reclassify the grids with the DOCELL command.
docell
/*save as cfb
/*Rainy climates with mild winters;constantly moist;warmest month < 22
if ((latitude >= 0 & tjan2down >= -3 & tjan2down < 18 & tjul2down > 10 ~
 & tjul2down <= 22 & pjan >= 60 & pjul >= 60) ~
 or (latitude < 0 & tjul2down >= -3 & tjul2down < 18 & tjan2down > 10 ~
 & tjan2down <= 22 & pjul >= 60 & pjan >= 60)) cfb = 3
end
&return
/*******************************************************************
&routine dfa
/*Delete grid if it already exists.
&if [exists dfa -grid] &then
kill dfa
```

```
/*Reclassify the grids with the DOCELL command.
docell
/*save as dfa
/*Rainy climates with severe winters; constantly moist; warmest month > 22
if ((latitude >= 0 & tjan2down >= -5 & tjan2down < 0 & tjul2down >= ~
 22 & pjan >= 2000 & pjul >= 1900 & trange < 30) ~
 or (latitude < 0 & tjul2down >= -5 & tjul2down < 0 & tjan2down >= ~
 22 & pjul >= 2000 & pjan >= 1900 & trange < 30)) dfa = 1
end
&return
/*******************************************************************
&routine cfa
/*Delete grid if it already exists.
&if [exists cfa -grid] &then
kill cfa

/*Reclassify the grids with the DOCELL command.
docell
/*save as cfa
/*Rainy climates with mild winters;constantly moist;warmest month > 22;
if ((latitude >= 0 & tjan2down >= -3 & tjul2down >= 22 & trange >= 18 ~
 & trange < 30 & cpre > 1000 & cpre < 13000 & pjan > 400 & pjul > 1000) ~
 or (latitude < 0 & tjul2down >= -3 & tjan2down >= 22 & trange >= 5 ~
 & cpre > 1000 & cpre < 17000 & pjul > 600 & pjan > 1800 & pjan > ~
 (1.5 * pjul))) cfa = 2
end
&return
/*******************************************************************
&routine cwa
/*Delete grid if it already exists.
&if [exists cwa -grid] &then
kill cwa

/*Reclassify the grids with the DOCELL command.
docell
/*save as cwa
/*Rainy climates with mild winters;dry winters;warmest month > 22;
if ((latitude >= 0 & tjan2down >= -3 & tjul2down > 15 & ctmp2down < ~
 230 & trange > 3 & pjan < 1000 & pjul > 1000 & cpre > 300) ~
 or (latitude < 0 & tjul2down >= -3 & tjul2down <= 19 & tjan2down > ~
 15 & trange > 2 & trange < 14 & pjan > 3500 & pjul < 500)) cwa = 2
end
&return
/*******************************************************************
&routine cfc
/*Delete grid if it already exists.
&if [exists cfc -grid] &then
kill cfc

/*Reclassify the grids with the DOCELL command.
docell
/*save as cfc
/*Rainy climates with mild winters;constantly moist;<4 months > 10 degrees;
if ((latitude >= 0 & tjan2down >= -3 & tjan2down < 10 & tjul2down < ~
 10 & pjan >= 60 & pjul >= 60 & dem < 1000) ~
 or (latitude < 0 & tjul2down >= -3 & tjul2down < 10 & tjan2down < 15 & ~
 pjul >= 60 & pjan >= 60 & dem < 1000)) cfc = 2
end
&return
/*******************************************************************
&routine csa
/*Delete grid if it already exists.
&if [exists csa -grid] &then
```

```
kill csa

/*Reclassify the grids with the DOCELL command.
docell
/*save csa
/*Rainy climates with mild winters;dry summers;warmest month > 22
if ((latitude >= 0 & tjan2down > -3 & tjan2down < 18 & tjul2down > 18 & ~
 pjul < 500 & pjan > 1000) ~
 or (latitude < 0 & tjul2down > -3 & tjul2down < 18 & tjan2down > 18 & ~
 pjan < 500 & pjul > 1000)) csa = 1
end
&return
/*******************************************************************
&routine dfd
/*Delete grid if it already exists.
&if [exists dfd -grid] &then
kill dfd

/*Reclassify the grids with the DOCELL command.
docell
/*save as dfd
/*Rainy climates with severe winters; constantly moist;coldest month < -38
if (latitude >= 0 & tjan2down <= -38 & tjul2down > 10 & (4 * pjan) ~
 >= pjul) dfd = 0
end
&return
/*******************************************************************
&routine dwd
/*Delete grid if it already exists.
&if [exists dwd -grid] &then
kill dwd

/*Reclassify the grids with the DOCELL command.
docell
/*save as dwd
/*Rainy climates with severe winters; dry winters; coldest month < -38
if ((latitude >= 0 & (4 * pjan) < pjul & tjan2down <= -38) ~
 or (latitude < 0 & (4 * pjul) < pjan & tjul2down <= -38)) dwd = 0
end
&return
/*******************************************************************
&routine dfbc
/*Delete grid if it already exists.
&if [exists dfbc -grid] &then
kill dfbc

/*Reclassify the grids with the DOCELL command.
docell
/*save as dfbc
/*Rainy climates with severe winters; constantly moist; warmest month < 22 (b)
/*Rainy climates with severe winters; constantly moist;>4 months > 10 degrees
(c)
if ((latitude >= 0 & tjan2down > -38 & tjan2down < -3 & tjul2down < 22 ~
 & pjul > 1500 & pjan > 500) ~
 or (latitude < 0 & tjul2down > -38 & tjul2down < -3 & tjan2down < 22 ~
 & pjan > 1500 & pjul > 500)) dfbc = 1
end
&return
/*******************************************************************
&routine bsk
/*Delete grid if it already exists.
&if [exists bsk -grid] &then
kill bsk
```

```
/*Reclassify the grids with the DOCELL command.
docell
/*save as bsk
/*BSk dry-cold steppe climate
if ((latitude >= 0 & tjan2down <= 1 & tjul2down > 10 & tjul2down < 24 ~
 & pjan <= 2000 & pjul <= 2000) ~
 or (latitude < 0 & tjul2down <= 1 & tjan2down > 10 & tjan2down < 24 ~
 & pjul <= 2000 & pjul <= 2000)) bsk = 2
end
&return
/******************************************************************
&routine bwk
/*Delete grid if it already exists.
&if [exists bwk -grid] &then
kill bwk

/*Reclassify the grids with the DOCELL command.
docell
/*save as bwk
/*BWk dry-cold desert climate
if ((latitude >= 0 & tjul2down >= 18 & ctmp2down > -150 & tjan2down < 5 ~
 & tjan2down > -20 & cpre < 4000 & pjul < 1000 & pjan < 1000) ~
 or (latitude < 0 & tjan2down >= 18 & ctmp2down > -150 & tjul2down < 5 ~
 & tjul2down > -20 & cpre < 4000 & pjan < 1000 & pjan < 1000)) bwk = 2
end
&return
/******************************************************************
&routine dwbc
/*Delete grid if it already exists.
&if [exists dwbc -grid] &then
kill dwbc

/*Reclassify the grids with the DOCELL command.
docell
/*save as dwbc
/*Rainy climates with severe winters; dry winters; warmest month < 22
if ((latitude >= 0 & tjul2down > 10 & tjul2down < 22 & tjan2down < -3 ~
 & (4 * pjan) < pjul) ~
 or (latitude < 0 & tjan2down > 10 & tjan2down < 22 & tjul2down < -3 ~
 & (4 * pjul) < pjan)) dwbc = 1
end
&return
/******************************************************************
&routine dwa
/*Delete grid if it already exists.
&if [exists dwa -grid] &then
kill dwa

/*Reclassify the grids with the DOCELL command.
docell
/*save as dwa
/*Rainy climates with severe winters; dry winters; warmest month > 22
if ((latitude >= 0 & pjan < 500 & pjul > 2000 & tjan2down < -3 & ~
 tjul2down > 22)~
 or (latitude < 0 & pjul < 500 & pjan > 2000 & tjul2down < -3 & ~
 tjan2down > 22)) dwa = 1
end
&return
/******************************************************************
&routine et
/*Delete grid if it already exists.
&if [exists et -grid] &then
kill et
```

```
/*Reclassify the grids with the DOCELL command.
docell
/*save as et
/*Polar climates with tundra
if ((latitude >= 0 & tjul2down > 0 & tjul2down < 10 ) ~
 or (latitude < 0 & tjan2down > 0 & tjan2down < 10 )) et = 0
end
&return
/******************************************************************
&routine ef
/*Delete grid if it already exists.
&if [exists ef -grid] &then
kill ef

/*Reclassify the grids with the DOCELL command.
docell
/*save as ef
/*Polar climates with perpetual frost
if (tjan2down < 0 & tjul2down < 0 & ctmp2down < 0) ef = 0
end
&return
/******************************************************************
&routine topo13
/*Delete grid if it already exists.
&if [exists topo13 -grid] &then
kill topo13

/*Reclassify the grids with the DOCELL command.
docell
/*save as ef
/*Polar climates with perpetual frost
if (grid13f == 1) topo13 = 9
end
&return
/******************************************************************
&routine merge
/*Delete grid if it already exists.
&if [exists cat13k -grid] &then
kill cat13k

/*Routine to combine all the preference grids together.  Order of
/*grids matters for this operation.  Preceding grid values will
/*be preserved over succeeding grid values that overlap.
cat13k = merge(snow,afr_rain,h0a,h0b,h0c,am,af,aw,bsh,bwh,cfb,csb,dfa,cfa,~
cwa,cfc,csa,dfd,dwd,dwa,dfbc,bsk,bwk,dwbc,et,ef,h2a,h2b,h3a,h3b,~
h4a,h4b,nodata,topo13)
&return
/******************************************************************
```

19.4 <u>The AML used to Produce the 18k Outcome</u>

```
/*****************************************************************
/*This code is used to create the Rule Based coverage of wildcat
/*presence as defined by expert knowledge from literature and
/*Dr. Andrew Kitchener for 18,000 years ago.
/*Wildcat preference is ranked from 0 to 3.
/*Rules about highland specifics and the limit of the distribution
/*relative to African precipitation have been added.
/*
/*The AML code to run this code is: &r cat18k
/*
/*Author: Erin E. Rees; August 13, 2002.
/*****************************************************************
/*MAIN METHOD
```

```
/*****************************************************************
/*Command to run the bailout routine if an error occurs.
&severity &error &routine bailout
/*The call order is also the order the grids should be layered.
&call snow
&call afr_rain
&call h0a
&call h0b
&call h0c
&call h2a
&call h2b
&call h3a
&call h3b
&call h4a
&call h4b
&call am
&call af
&call aw
&call bwh
&call bsh
&call cfb
&call csb
&call dfa
&call cfa
&call cwa
&call cfc
&call csa
&call dfd
&call dwd
&call dwa
&call dfbc
&call bsk
&call bwk
&call dwbc
&call et
&call ef
&call topo18
&call merge
&type MAIN METHOD COMPLETED
&return
/*****************************************************************
/*ROUTINES
/*****************************************************************
/*Routine to bailout of the AML if an error occurs.
&routine bailout

/*Error message to be displayed if an error occurs:
&setvar errormessage = ERROR in AML - program closing.

/*Print error message to the screen.
&type %errormessage%

/*Close all files.
&call closeFiles

/*Stop the 'cat' AML and return control to the command line.
&stop
/******************************************************************
/*Routine to close all files that are open.
&routine closeFiles
&setvar close = [close -all]
&return
/******************************************************************
/******************************************************************
```

```
/****************************************************************
/*Routines to reclassify the environmental grids according to the
/*Koppen Climate Classification scheme.
/****************************************************************
&routine snow
/*Delete grid if it already exists.
&if [exists snow -grid] &then
kill snow

/*Reclassify the grids with the DOCELL command.
docell
/*Snow depth beyond preferred cat levels (i.e. > 30 cm)
 if (tjan10down < -5 & tjul10down <= 20) snow = 0
 end
&return
/****************************************************************
&routine afr_rain
/*Delete grid if it already exists.
&if [exists afr_rain -grid] &then
kill afr_rain

/*Reclassify the grids with the DOCELL command.
docell
/*Highland beyond preferred cat levels
 if (africa == 1 & cpre < 365 & pjan < 100 & pjul < 1000) ~
   afr_rain = 0
 end
&return
/****************************************************************
&routine h0a
/*Delete grid if it already exists.
&if [exists h0a -grid] &then
kill h0a

/*Reclassify the grids with the DOCELL command.
docell
/*Highland beyond preferred cat levels
if (europe == 1 & dem > 3000) h0a = 0
end
&return
/****************************************************************
&routine h0b
/*Delete grid if it already exists.
&if [exists h0b -grid] &then
kill h0b

/*Reclassify the grids with the DOCELL command.
docell
/*Highland beyond preferred cat levels
if (africa == 1 & dem > 3500) h0b = 0
end
&return
/****************************************************************
&routine h0c
/*Delete grid if it already exists.
&if [exists h0c -grid] &then
kill h0c

/*Reclassify the grids with the DOCELL command.
docell
/*Highland beyond preferred cat levels
if (asia == 1 & dem > 3000) h0c = 0
end
&return
```

```
/************************************************************
&routine h2a
/*Delete grid if it already exists.
&if [exists h2a -grid] &then
kill h2a

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (africa == 1 & dem < 2000) h2a = 3
 end
&return
/************************************************************
&routine h2b
/*Delete grid if it already exists.
&if [exists h2b -grid] &then
kill h2b

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (asia == 1 & dem < 2000) h2b = 3
 end
&return
/************************************************************
&routine h3a
/*Delete grid if it already exists.
&if [exists h3a -grid] &then
kill h3a

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (africa == 1 & dem >= 2000 & dem < 2500) h3a = 2
 end
&return
/************************************************************
&routine h3b
/*Delete grid if it already exists.
&if [exists h3b -grid] &then
kill h3b

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (asia == 1 & dem >= 2000 & dem < 2500) h3b = 2
 end
&return
/************************************************************
&routine h4a
/*Delete grid if it already exists.
&if [exists h4a -grid] &then
kill h4a

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (africa == 1 & dem >= 2500 & dem < 3500) h4a = 1
 end
&return
/************************************************************
&routine h4b
/*Delete grid if it already exists.
&if [exists h4b -grid] &then
```

```
kill h4b

/*Reclassify the grids with the DOCELL command.
docell
/*Highland within preferred cat levels
 if (asia == 1 & dem >= 2500 & dem < 3000) h4b = 1
 end
&return
/*****************************************************************
&routine am
/*Delete grid if it already exists.
&if [exists am -grid] &then
kill am

/*Reclassify the grids with the DOCELL command.
docell
/*saved as am
/*Am monsoon variety of Af
 if (ctmp10down > 180 & cpre < 5840 & pjan < 500 ~
  & pjul > 9000) am = 0
  /*am APPEARS TO BE A GOOD THING (fudge rules)
 end
&return
/*****************************************************************
&routine aw
/*Delete grid if it already exists.
&if [exists aw -grid] &then
kill aw

/*Reclassify the grids with the DOCELL command.
docell
/*saved as aw
/*Aw tropical savanna
if (ctmp10down >= 180 & tjan10down >= 18 & tjul10down >= 18 ~
 & trange < 10 & (pjan > (1.5 * pjul) or pjul > (1.5 * pjan))) aw = 3
end

&return
/*****************************************************************
&routine af
/*Delete grid if it already exists.
&if [exists af -grid] &then
kill af

/*Reclassify the grids with the DOCELL command.
docell
/*saved as af
/*Af tropical rainforest
if (ctmp10down >= 180 & pjan >= 18 & pjul >= 18 & trange <= 7 & ~
 pjul > 400 & pjan > 500 & cpre > 1000) af = 0
end
&return
/*****************************************************************
&routine bsh
/*Delete grid if it already exists.
&if [exists bsh -grid] &then
kill bsh

/*Reclassify the grids with the DOCELL command.
docell
/*save as bsh
/*BSh dry-hot steppe climate
if ((latitude >= 0 & tjul10down >= 22 & tjul10down < 35 & ~
 tjan10down >= 3 & ctmp10down > 35 & pjan < 6000 & pjul < 4500 & ~
```

```
 (pjul + pjan) > 450 & cpre < 5000) ~
 or (latitude < 0 & tjul10down >= 7 & tjan10down >= 22 & ~
 pjul < 1500 & pjan < 4500 & trange <= 17 & (pjul + pjan) > 450)) ~
 bsh = 3
end
&return
/*****************************************************************
&routine bwh
/*Delete grid if it already exists.
&if [exists bwh -grid] &then
kill bwh

/*Reclassify the grids with the DOCELL command.
docell
/*save as bwh
/*BWh dry-hot desert climate
if (tjan10down > 0 & tjul10down > 0 & pjan < 1500 & pjul < 1500 ~
 & cpre < 4000) bwh = 2
end
&return
/*****************************************************************
&routine csb
/*Delete grid if it already exists.
&if [exists csb -grid] &then
kill csb

/*Reclassify the grids with the DOCELL command.
docell
/*save csb
/*Rainy climates with mild winters;dry winters;warmest month < 22
if ((latitude >= 0 & tjan10down > 0 & tjan10down <= 10 & ~
 tjul10down <= 23 & pjan <= 10000 & pjul < 1500 & dem < 1000 & ~
 grid18fff < 1 & trange < 20) ~
 or (latitude < 0 & tjul10down > 0 & tjul10down <= 10 & ~
 tjan10down <= 23 & pjul <= 10000 & pjan < 1500 & dem < 1000 & ~
 grid18fff < 1 & trange < 20)) csb = 2
end
&return
/*****************************************************************
&routine cfb
/*Delete grid if it already exists.
&if [exists cfb -grid] &then
kill cfb

/*Reclassify the grids with the DOCELL command.
docell
/*save as cfb
/*Rainy climates with mild winters;constantly moist;warmest month < 22
if ((latitude >= 0 & tjan10down >= -3 & tjan10down < 18 & ~
 tjul10down > 10 & tjul10down <= 22 &  pjan >= 60 & pjul >= 60) ~
 or (latitude < 0 & tjul10down >= -3 & tjul10down < 18 & ~
 tjan10down > 10 & tjan10down <= 22 & pjul >= 60 & pjan >= 60)) cfb = 3
end
&return
/*****************************************************************
&routine dfa
/*Delete grid if it already exists.
&if [exists dfa -grid] &then
kill dfa

/*Reclassify the grids with the DOCELL command.
docell
/*save as dfa
/*Rainy climates with severe winters; constantly moist; warmest month > 22
```

```
if ((latitude >= 0 & tjan10down >= -5 & tjan10down < 0 & tjul10down >= 22 ~
 & pjan >= 2000 & pjul >= 1900 & trange < 30) ~
 or (latitude < 0 & tjul10down >= -5 & tjul10down < 0 & tjan10down >= 22 ~
 & pjul >= 2000 & pjan >= 1900 & trange < 30)) dfa = 1
end
&return
/*****************************************************************
&routine cfa
/*Delete grid if it already exists.
&if [exists cfa -grid] &then
kill cfa

/*Reclassify the grids with the DOCELL command.
docell
/*save as cfa
/*Rainy climates with mild winters;constantly moist;warmest month > 22;
if ((latitude >= 0 & tjan10down >= -3 & tjul10down >= 22 & ~
 trange >= 18 & trange < 30 & cpre > 1000 & cpre < 13000 & ~
 pjan > 400 & pjul > 1000) ~
 or (latitude < 0 & tjul10down >= -3 & tjan10down >= 22 & ~
 trange >= 5 & cpre > 1000 & cpre < 17000 & pjul > 600 & pjan > ~
 1800 & pjan > (1.5 * pjul))) cfa = 2
end
&return
/*****************************************************************
&routine cwa
/*Delete grid if it already exists.
&if [exists cwa -grid] &then
kill cwa

/*Reclassify the grids with the DOCELL command.
docell
/*save as cwa
/*Rainy climates with mild winters;dry winters;warmest month > 22;
if ((latitude >= 0 & tjan10down >= -3 & tjul10down > 15 & ctmp10down ~
 < 230 & trange > 3 & pjan < 1000 & pjul > 1000 & cpre > 300) ~
 or (latitude < 0 & tjul10down >= -3 & tjul10down <= 19 & tjan10down ~
 > 15 & trange > 2 & trange < 14 & pjan > 3500 & pjul < 500)) cwa = 2
end
&return
/*****************************************************************
&routine cfc
/*Delete grid if it already exists.
&if [exists cfc -grid] &then
kill cfc

/*Reclassify the grids with the DOCELL command.
docell
/*save as cfc
/*Rainy climates with mild winters;constantly moist;<4 months > 10 degrees;
if ((latitude >= 0 & tjan10down >= -3 & tjan10down < 10 & tjul10down ~
 < 10 & pjan >= 60 & pjul >= 60 & dem < 1000) ~
 or (latitude < 0 & tjul10down >= -3 & tjul10down < 10 & tjan10down < 15 & ~
 pjul >= 60 & pjan >= 60 & dem < 1000)) cfc = 2
end
&return
/*****************************************************************
&routine csa
/*Delete grid if it already exists.
&if [exists csa -grid] &then
kill csa

/*Reclassify the grids with the DOCELL command.
docell
```

```
/*save csa
/*Rainy climates with mild winters;dry summers;warmest month > 22
if ((latitude >= 0 & tjan10down > -3 & tjan10down < 18 & tjul10down > 18 & ~
 pjul < 500 & pjan > 1000) ~
 or (latitude < 0 & tjul10down > -3 & tjul10down < 18 & tjan10down > 18 & ~
 pjan < 500 & pjul > 1000)) csa = 2
end
&return
/*****************************************************************
&routine dfd
/*Delete grid if it already exists.
&if [exists dfd -grid] &then
kill dfd

/*Reclassify the grids with the DOCELL command.
docell
/*save as dfd
/*Rainy climates with severe winters; constantly moist;coldest month < -38
if (latitude >= 0 & tjan10down <= -38 & tjul10down > 10 & (4 * pjan) ~
 >= pjul) dfd = 0
end
&return
/*****************************************************************
&routine dwd
/*Delete grid if it already exists.
&if [exists dwd -grid] &then
kill dwd

/*Reclassify the grids with the DOCELL command.
docell
/*save as dwd
/*Rainy climates with severe winters; dry winters; coldest month < -38
if ((latitude >= 0 & (4 * pjan) < pjul & tjan10down <= -38) ~
 or (latitude < 0 & (4 * pjul) < pjan & tjul10down <= -38)) dwd = 0
/*SEEMS GOOD
end
&return
/*****************************************************************
&routine dfbc
/*Delete grid if it already exists.
&if [exists dfbc -grid] &then
kill dfbc

/*Reclassify the grids with the DOCELL command.
docell
/*save as dfbc
/*Rainy climates with severe winters; constantly moist; warmest month < 22 (b)
/*Rainy climates with severe winters; constantly moist;>4 months > 10 degrees
(c)
if ((latitude >= 0 & tjan10down > -38 & tjan10down < -3 & tjul10down < ~
 22 & pjul > 1500 & pjan > 500) ~
 or (latitude < 0 & tjul10down > -38 & tjul10down < -3 & tjan10down < ~
 22 & pjan > 1500 & pjul > 500)) dfbc = 1
end
&return
/*****************************************************************
&routine bsk
/*Delete grid if it already exists.
&if [exists bsk -grid] &then
kill bsk

/*Reclassify the grids with the DOCELL command.
docell
/*save as bsk
```

```
/*BSk dry-cold steppe climate
if ((latitude >= 0 & tjan10down <= 1 & tjul10down > 10 & tjul10down < 24 ~
 & pjan <= 2000 & pjul <= 2000) ~
 or (latitude < 0 & tjul10down <= 1 & tjan10down > 10 & tjan10down < 24 ~
 & pjul <= 2000 & pjul <= 2000)) bsk = 2
end
&return
/******************************************************************
&routine bwk
/*Delete grid if it already exists.
&if [exists bwk -grid] &then
kill bwk

/*Reclassify the grids with the DOCELL command.
docell
/*save as bwk
/*BWk dry-cold desert climate
if ((latitude >= 0 & tjul10down >= 18 & ctmp10down > -150 & tjan10down ~
 < 5 & tjan10down > -20 & cpre < 4000 & pjul < 1000 & pjan < 1000) ~
 or (latitude < 0 & tjan10down >= 18 & ctmp10down > -150 & tjul10down ~
 < 5 & tjul10down > -20 & cpre < 4000 & pjan < 1000 & pjan < 1000)) bwk = 2
end
&return
/******************************************************************
&routine dwbc
/*Delete grid if it already exists.
&if [exists dwbc -grid] &then
kill dwbc

/*Reclassify the grids with the DOCELL command.
docell
/*save as dwbc
/*Rainy climates with severe winters; dry winters; warmest month < 22
if ((latitude >= 0 & tjul10down > 10 & tjul10down < 22 & tjan10down < -3 ~
 & (4 * pjan) < pjul) ~
 or (latitude < 0 & tjan10down > 10 & tjan10down < 22 & tjul10down < -3 ~
 & (4 * pjul) < pjan)) dwbc = 1
end
&return
/******************************************************************
&routine dwa
/*Delete grid if it already exists.
&if [exists dwa -grid] &then
kill dwa

/*Reclassify the grids with the DOCELL command.
docell
/*save as dwa
/*Rainy climates with severe winters; dry winters; warmest month > 22
if ((latitude >= 0 & pjan < 500 & pjul > 2000 & tjan10down < -3 & ~
 tjul10down > 22) ~
 or (latitude < 0 & pjul < 500 & pjan > 2000 & tjul10down < -3 & ~
 tjan10down > 22)) dwa = 1
end
&return
/******************************************************************
&routine et
/*Delete grid if it already exists.
&if [exists et -grid] &then
kill et

/*Reclassify the grids with the DOCELL command.
docell
/*save as et
```

```
/*Polar climates with tundra
if ((latitude >= 0 & tjul10down > 0 & tjul10down < 10 ) ~
 or (latitude < 0 & tjan10down > 0 & tjan10down < 10 )) et = 0
end
&return
/*****************************************************************
&routine ef
/*Delete grid if it already exists.
&if [exists ef -grid] &then
kill ef

/*Reclassify the grids with the DOCELL command.
docell
/*save as ef
/*Polar climates with perpetual frost
if (tjan10down < 0 & tjul10down < 0 & ctmp10down < 0) ef = 0
end
&return
/*****************************************************************
&routine topo18
/*Delete grid if it already exists.
&if [exists topo18 -grid] &then
kill topo18

/*Reclassify the grids with the DOCELL command.
docell
/*save as ef
/*Polar climates with perpetual frost
if (grid18f == 1) topo18 = 9
end
&return
/*****************************************************************
&routine merge
/*Delete grid if it already exists.
&if [exists cat18k -grid] &then
kill cat18k

/*Routine to combine all the preference grids together.  Order of
/*grids matters for this operation.  Preceding grid values will
/*be preserved over succeeding grid values that overlap.
cat18k = merge(snow,afr_rain,h0a,h0b,h0c,am,af,aw,bsh,bwh,cfb,~
csb,dfa,cfa,cwa,cfc,csa,dfd,dwd,dwa,dfbc,bsk,bwk,dwbc,et,ef,h2a,~
h2b,h3a,h3b,h4a,h4b,nodata,topo18)
&return
/*****************************************************************
```