Lab 10, Week 13

Sidi Wu

In this lab you will write the predict method for mars objects. Before you start you should take a quick look at the source code of the `predict.lm()` function.

As in labs 8 and 9, you should be working from your `mars` project for this lab. You will need to copy the file `testpredict.RData` from the `Exercises/ProjectTestfiles/testthat` folder of the class GitHub repository to your `tests/testthat` folder.

1. Below is an implementation of the `predict.mars()` function. It is based on `predict.lm()`, but with some extra fiddling to the the model matrix right in the case of new data. (There is probably a better way, but this is what I came up with.) Your task is to write the function `make_B()` that takes a design matrix `X` and a list of basis function specifications `Bfuncs` as input and returns the corresponding matrix of basis functions.

```
predict.mars <- function(object,newdata) {
  if(missing(newdata) || is.null(newdata)) {
    B <- as.matrix(object$B)
  }
  else {
    tt <- terms(object$formula,data=newdata)
    tt <- delete.response(tt)
    mf <- model.frame(tt,newdata)
    mt <- attr(mf, "terms")
    X <- model.matrix(mt, mf)[,-1] # remove intercept
    B <- make_B(X,object$Bfuncs)
  }
  beta <- object$coefficients
  drop(B %*% beta)
}
```

2. Create a test script `testpredict.R` in your `mars/tests/testthat` folder. `testpredict.R` should load `testpredict.RData` and call your `predict.mars()` function with the following two inputs:
    1. `predict.mars(testmars)`: predictions for the same data used to fit the model. Compare the output of this call to the expected output `testpredict`.
    2. `predict(testmars,newdata=marstestdata)`: predictions from the fitted model in `testmars` on "new data" in `marstestdata`. (This is the same data used to fit the model, but by passing it in as newdata we test that part of the code.) Compare the output of this call to the expected output `testpredict`.