- Academic honesty is important. Strict plagiarism policy will be enforced.
- You can work in a group of 2–3 people. (Group with different people in each HW.)
- Attach your R code after your typed-homework.
- Send also your R code to sta208.spring2015.ucd@gmail.com in a zipped file.

1. Download the file "threes.Rdata" from smartsite. Now you should have a matrix `threes` that has dimension 658×256. (This data set was taken from the data page on http://www-stat.stanford.edu/~tibs/ElemStatLearn/.) Each row of the matrix corresponds to an image of a "3" that was written by a different person. Hence each row vector is of length 256, corresponding to a 16 16 pixels image that has been unraveled into a vector, and each pixel takes grayscale values between 1 and 1. You can use the `plot.digit` function from HW2 to plot any of the images, i.e., any row of the matrix `threes`.

   (a) Compute the principal component directions and principal component scores of threes. Plot the first two principal component scores (the x-axis being the first score and the y-axis being the second score). Note that each point in this plot corresponds to an image of a "3".

   (b) For each of the first two principal component scores, compute the following percentiles: 5%, 25%, 50%, 75%, 95%. Draw these values as vertical and horizontal lines on top of your plot (i.e., vertical for the percentiles of the first principal component score, and horizontal for those of the second.) (Hint: use `quantile` for the percentiles, and abline to draw the lines.)

   (c) Now you want to identify a point (i.e., an image of a "3") close to each of the vertices of the grid on your plot. This can be done by using the `identify` function, it will print the index of the point that is closest to your clicks location. (Note: although the function returns a vector of indices, and it claims that this vector is ordered by the order of your clicks, it actually given them in a sorted order. To record the indices in order, you may want to consider `replicate(25, identify(x,y,n=1))` instead.)

   (d) Plot all of the images of "3"s that you picked out in part (c), in an order that corresponds to the vertices of the grid.

   (e) Looking at these digits, what can be said about the nature of the first two principal component scores? (The first principal component score is increasing as you move from left-to- right in any of the rows. The second principal component score is decreasing as you move from top-to-bottom in any of the columns.) In other words, explain what changes with respect to changes in each of the component scores.

(f) Plot the proportion of variance explained by the first k principal component directions, as a function of $k = 1, \ldots, 256$. How many principal component directions would we need to explain 50% of the variance? How many to explain 90% of the variance?

2. In this question you will fit various spline models to the fossil data (in "fossil.csv") of Chaudhuri and Marron (1999). These data consist of 106 measurements of ratios of strontium isotopes found in fossil shells and their ages.

   (a) Fit a polynomial function, using 5 fold cross-validation for choosing the degree of the polynomial. Report the selected model.

   (b) Fit a natural cubic spline, using 5 fold cross-validation for choosing the degrees of freedom. Report the degrees of freedom of the chosen model.

   (c) Fit a smoothing spline, using 5 fold cross-validation for choosing the tuning parameter. Report the degrees of freedom of the chosen model.

   (d) Fit a local linear regression (`loess`), using 5 fold cross-validation for choosing the tuning parameter `span`. Report the corresponding degrees of freedom of the chosen model.

   (e) Make a scatterplot of the data, superimposed with the fitted curves by (a), (b), (c) and (d). Compare and discuss the fitted models.

3. It was mentioned that GAMs are generally fit using a backfitting approach. The idea behind backfitting is actually quite simple. We will now explore backfitting in the context of linear additive model. Download the `income` data from

   ```
   income <- read.csv("http://www-bcf.usc.edu/~gareth/ISL/Income2.csv")
   ```

   Suppose we have the model

   $$y_i = \beta_0 + f_1(x_{i1}) + f_2(x_{i2}) + \varepsilon.$$

   where y_i is `Income`, x_{i1} is `Education` and $x_{i2} =$ `Seniority`.

   (a) Suppose f_1 and f_2 are linear. Fit a multiple linear regression model to the data. Denote the estimate of f_1 as $\hat{f}_1$. (You should center the function $\hat{f}_1$)

   (b) Compute the partial residual $z_{1i} = y_i - \hat{f}_1(x_{1i})$ and plot z_{1i} vs x_{2i}. Then fit a smoothing spline on the scatter plot. Denote the centered estimated smooth function as $\hat{f}_2$.

   (c) Compute the partial residual $z_{2i} = y_i - \hat{f}_2(x_{2i})$ and plot z_{2i} vs x_{1i}. Then fit a smoothing spline on the scatter plot. Denote the centered estimated smooth function as $\hat{f}_1$.

   (d) Repeat (b) and (c) a number of times until convergence. (One way to declare convergence is to look at the changes of $\hat{f}_1(x_1) + \hat{f}_2(x_2)$ in successive iterations.)

(e) On this data set, how many backfitting iterations were required in order to obtain a good result? Plot the function $\hat{f}_1(x_1) + \hat{f}_2(x_2)$ on $\mathbb{R}^2$. How does the result compare to thin-plate spline and local linear regression in lab 10?