Homework 1

1. Nearest neighbor predictor. We have a collection of $n$ observations, $x^i \in \mathbb{R}^d$, $y^i \in \mathbb{R}$, $i = 1, \ldots, n$. Based on these observations, the nearest neighbor predictor is defined as $g^\text{nn}(x) = y^k$, where $x^k$ is a nearest neighbor of $x$ among the data points. (We can break ties arbitrarily. Recall that $x^k$ is a nearest neighbor of $x$ means that $\|x - x^k\| \leq \|x - x^i\|$ for $i = 1, \ldots, n$.)

(a) Write a Julia function $y\_hat = \text{nn\_predictor}(x, X, y)$ that implements $g^\text{nn}$, given the argument $x$. The second and third arguments give the data on which the predictor is based: $X$ is an $n \times d$ matrix whose $i$th row is $(x^i)^T$, and $y$ is an $n$-vector with $i$th entry $y^i$.

(b) Report the train and test RMSE of the predictor in part (a) on the data provided in nearest\_neighbor\_data.json, training on the first 1500 data points and testing on the rest. Briefly interpret your results.

(c) Approximately how many flops (floating point operations) are required to evaluate $g^\text{nn}(x)$, in terms of $n$ and $d$?

2. Soft nearest neighbor. In this exercise we examine an extension of the nearest neighbor predictor that can occasionally perform better. For data set $x^i \in \mathbb{R}^d$, $y^i \in \mathbb{R}$, $i = 1, \ldots, n$, the soft nearest neighbor predictor is defined as

$$g^\text{snn}(x) = \frac{\sum_{i=1}^n y^i e^{-\|x - x^i\|^2/\sigma^2}}{\sum_{i=1}^n e^{-\|x - x^i\|^2/\sigma^2}},$$

where $\sigma \geq 0$ is a parameter. Note that $\sigma$ can be thought of as a distance; the function $e^{-\|x - x^i\|^2/\sigma^2}$ is near one when the distance between $x$ and $x^i$ is much less than $\sigma$, and it is very small when the distance is much more than $\sigma$.

(a) What does $g^\text{snn}(x)$ converge to as $\sigma \to 0$? (Explain briefly.)

(b) What does $g^\text{snn}(x)$ converge to as $\sigma \to \infty$? (Explain briefly.)

(c) Implement the soft nearest neighbor predictor in Julia as $y\_hat = \text{snn\_predictor}(x,X,y,\sigma)$.

(d) Using the data in nearest\_neighbor\_data.json, plot train and test RMSEs for $g^\text{snn}$ as a function of $\sigma$ over the range $[10^{-1}, 10^1]$. What value of $\sigma$ would you choose? How does the test RMSE of predictor compare to the nearest neighbor predictor and the constant predictor?
3. Selecting regressors to use. When considering different models, we want to consider which features give us the smallest test error. In selecting_regressors.json, you will find an $n \times d$ matrix $U$ of raw data, with rows $(u^i)^T$, with $u^i \in \mathbb{R}^3$.

(a) Define $S = \{s_1, \ldots, s_k\} \subseteq \{1, 2, 3\}$, and let $\phi_S(u) = (1, u_{s_1}, \ldots u_{s_k})$. In words, $\phi_S$ is the embedding that has a constant first feature, and the other features are the components of $u$ with indices in $S$. For example, if $S = \{1, 3\}$, then $\phi_S(u) = (1, u_1, u_3)$. Implement the following function,

$$X = \text{embed}(S, U),$$

which returns a data matrix $X$, with rows $(x^i)^T = (\phi_S(u^i))^T$. The size of $X$ is $n \times (|S| + 1)$, where $|S|$ denotes the number of elements in $S$.

(b) With the data selecting_regressors.json and the code in selecting_regressors.jl, use the embedding function you implemented in part (a) to find parameter value $\theta_S$ which minimizes the training set RMSE, for each $S \subseteq \{1, 2, 3\}$. For each $S$, compute and report the test RMSE for the associated predictor. Which are the two best embeddings? *Hint:* Use the provided function `powerset()` to compute all the subsets of $\{1, 2, 3\}$. 
