Problem 1 (Optimizing a Kernel, 15pts)

Kernel-based regression techniques are similar to nearest-neighbor regressors: rather than fit a parametric model, they predict values for new data points by interpolating values from existing points in the training set. In this problem, we will consider a kernel-based regressor of the form:

$$f(x^*) = \frac{\sum_n K(x_n, x^*) y_n}{\sum_n K(x_n, x^*)}$$

where (x_n, y_n) are the training data points, and $K(x, x')$ is a kernel function that defines the similarity between two inputs x and x'. Assume that each x_i is represented as a column vector, i.e. a D by 1 vector where D is the number of features for each data point. A popular choice of kernel is a function that decays as the distance between the two points increases, such as

$$K(x, x') = \exp(-||x - x'||_2^2) = \exp(-(x - x')^T (x - x'))$$

However, the squared Euclidean distance $||x - x'||_2^2$ may not always be the right choice. In this problem, we will consider optimizing over squared Mahalanobis distances

$$K(x, x') = \exp(-(x - x')^T W (x - x'))$$

where W is a symmetric D by D matrix. Intuitively, introducing the weight matrix W allows for different dimensions to matter differently when defining similarity.

1. Let $\{(x_n, y_n)\}_{n=1}^N$ be our training data set. Suppose we are interested in minimizing the residual sum of squares. Write down this loss over the training data $\mathcal{L}(W)$ as a function of W.

   Important: When computing the prediction $f(x_i)$ for a point x_i in the training set, carefully consider for which points x' you should be including the term $K(x_i, x')$ in the sum.

2. In the following, let us assume that $D = 2$. That means that W has three parameters: W_{11}, W_{22}, and $W_{12} = W_{21}$. Expand the formula for the loss function to be a function of these three parameters.

3. Derive the gradients of the loss function with respect to each of the parameters of W for the $D = 2$ case. (This will look a bit messy!)

Problem 1 (cont.)

4. Consider the following data set:

```
x1 , x2 , y
  0 , 0 , 0
  0 , .5 , 0
  0 , 1 , 0
  .5 , 0 , .5
  .5 , .5 , .5
  .5 , 1 , .5
  1 , 0 , 1
  1 , .5 , 1
  1 , 1 , 1
```

   And the following kernels:

$$W_1 = \alpha \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad W_2 = \alpha \begin{bmatrix} 0.1 & 0 \\ 0 & 1 \end{bmatrix} \qquad W_3 = \alpha \begin{bmatrix} 1 & 0 \\ 0 & 0.1 \end{bmatrix}$$

   with $\alpha = 10$. Write some Python code to compute the loss with respect to each kernel for the dataset provided above. Which kernel does best? Why? How does the choice of α affect the loss?

   For this problem, you can use our staff **script to compare your code to a set of staff-written test cases.** This requires, however, that you use the structure of the starter code provided in `T1_P1.py`. More specific instructions can be found at the top of the file `T1_P1_Testcases.py`. You may run the test cases in the command-line using `python T1_P1_TestCases.py`. **Note that our set of test cases is not comprehensive: just because you pass does not mean your solution is correct! We strongly encourage you to write your own test cases and read more about ours in the comments of the Python script.**

5. Bonus: Code up a gradient descent to optimize the kernel for the data set above. Start your gradient descent from W_1. Report on what you find.
   Gradient descent is discussed in Section 3.4 of the cs181-textbook notes and Section 5.2.4 of Bishop, and will be covered later in the course!

Solution

1. Recall from Lecture 2 that the loss function for the sum of the squared residual is:

$$\mathcal{L}(W) = \frac{1}{2}\sum_n (y_n - \hat{y}_n)^2$$

Our prediction for $\hat{y}$ is given by:

$$\hat{y} = f(x^*) = \frac{\sum_n K(x_n, x^*) y_n}{\sum_n K(x_n, x^*)}$$

Therefore, the loss function over the training data is:

$$\mathcal{L}(W) = \frac{1}{2}\sum_{i=1}^{N}\left(y_i - \frac{\sum_{n\in N:n\neq i} K(x_n, x_i) y_n}{\sum_{n\in N:n\neq i} K(x_n, x_i)}\right)^2$$

We are given the Mahalanobis distance as:

$$K(x, x') = \exp(-(x - x')^T W (x - x'))$$

so we may now rewrite our loss function as a function of W:

$$\mathcal{L}(W) = \frac{1}{2}\sum_{i=1}^{N}\left(y_i - \frac{\sum_{n\in N:n\neq i} \exp(-(x_n - x_i)^T W (x_n - x_i)) y_n}{\sum_{n\in N:n\neq i} \exp(-(x_n - x_i)^T W (x_n - x_i))}\right)^2$$

2. To expand our loss function, we first need to rewrite $-(x_n - x_i)^T W (x_n - x_i)$ in terms of our three desired parameters. When $D = 2$ we can write the matrix W and vectors x_n and x_i as:

$$W = \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix} \quad x_n = \begin{bmatrix} x_{n1} \\ x_{n2} \end{bmatrix} \quad x_i = \begin{bmatrix} x_{i1} \\ x_{i2} \end{bmatrix}$$

So we rewrite:

$$-(x_n - x_i)^T W (x_n - x_i) = -\left(\begin{bmatrix} x_{n1} \\ x_{n2} \end{bmatrix} - \begin{bmatrix} x_{i1} \\ x_{i2} \end{bmatrix}\right)^T \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix} \left(\begin{bmatrix} x_{n1} \\ x_{n2} \end{bmatrix} - \begin{bmatrix} x_{i1} \\ x_{i2} \end{bmatrix}\right)$$

$$- \begin{bmatrix} x_{n1} - x_{i1} & x_{n2} - x_{i2} \end{bmatrix} \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix} \begin{bmatrix} x_{n1} - x_{i1} \\ x_{n2} - x_{i2} \end{bmatrix}$$

$$= - \begin{bmatrix} W_{11}(x_{n1} - x_{i1}) + W_{21}(x_{n2} - x_{i2}) & W_{12}(x_{n1} - x_{i1}) + W_{22}(x_{n2} - x_{i2}) \end{bmatrix} \begin{bmatrix} x_{n1} - x_{i1} \\ x_{n2} - x_{i2} \end{bmatrix}$$

$$= -(W_{11}(x_{n1} - x_{i1})^2 + W_{21}(x_{n1} - x_{i1})(x_{n2} - x_{i2}) + W_{12}(x_{n1} - x_{i1})(x_{n2} - x_{i2}) + W_{22}(x_{n2} - x_{i2})^2)$$

Because $W_{12} = W_{21}$, we can rewrite this as:

$$-W_{11}(x_{n1} - x_{i1})^2 - 2W_{12}(x_{n1} - x_{i1})(x_{n2} - x_{i2}) - W_{22}(x_{n2} - x_{i2})^2$$

To simplify our expression, we define two new scalars as the Euclidean distances in the above expression:

$$x_{ni1} = x_{n1} - x_{i1} \quad \text{and} \quad x_{ni2} = x_{n2} - x_{i2}$$

So we rewrite our expression as:

$$-W_{11}x_{ni1}^2 - 2W_{12}x_{ni1}x_{ni2} - W_{22}x_{ni2}^2$$

Thus, substituting into our loss function, we have rewritten our loss function in terms of our three desired parameters W_{11}, W_{22}, and W_{12}:

$$\mathcal{L}(W) = \frac{1}{2}\sum_{i=1}^{N}\left(y_i - \frac{\sum_{n\in N:n\neq i} \exp(-W_{11}x_{ni1}^2 - 2W_{12}x_{ni1}x_{ni2} - W_{22}x_{ni2}^2) y_n}{\sum_{n\in N:n\neq i} \exp(-W_{11}x_{ni1}^2 - 2W_{12}x_{ni1}x_{ni2} - W_{22}x_{ni2}^2)}\right)^2$$

3. As expressed above, the kernel function is:

$$K(x_n, x_i) = \exp(-W_{11}x_{ni1}^2 - 2W_{12}x_{ni1}x_{ni2} - W_{22}x_{ni2}^2)$$

We may derive the above with respect to our three desired parameters:

$$\frac{\partial K(x_n, x_i)}{\partial W_{11}} = -x_{ni1}^2 \exp(-W_{11}x_{ni1}^2 - 2W_{12}x_{ni1}x_{ni2} - W_{22}x_{ni2}^2)$$

$$\frac{\partial K(x_n, x_i)}{\partial W_{12}} = -2x_{ni1}x_{ni2} \exp(-W_{11}x_{ni1}^2 - 2W_{12}x_{ni1}x_{ni2} - W_{22}x_{ni2}^2)$$

$$\frac{\partial K(x_n, x_i)}{\partial W_{22}} = -x_{ni2}^2 \exp(-W_{11}x_{ni1}^2 - 2W_{12}x_{ni1}x_{ni2} - W_{22}x_{ni2}^2)$$

Recall our loss function from above:

$$\mathcal{L}(W) = \frac{1}{2}\sum_{i=1}^{N}\left(y_i - \frac{\sum_{n\in N:n\neq i} K(x_n, x_i)y_n}{\sum_{n\in N:n\neq i} K(x_n, x_i)}\right)^2$$

First, deriving the gradient of the loss function with respect to W_{11}:

$$\frac{\partial \mathcal{L}(W)}{\partial W_{11}} = \frac{\partial}{\partial W_{11}}\left[\frac{1}{2}\sum_{i=1}^{N}\left(y_i - \frac{\sum_{n\in N:n\neq i} K(x_n, x_i)y_n}{\sum_{n\in N:n\neq i} K(x_n, x_i)}\right)^2\right]$$

Applying the chain rule:

$$\frac{\partial \mathcal{L}(W)}{\partial W_{11}} = \sum_{i=1}^{N}\left[\left(y_i - \frac{\sum_{n\in N:n\neq i} K(x_n, x_i)y_n}{\sum_{n\in N:n\neq i} K(x_n, x_i)}\right)\frac{\partial}{\partial W_{11}}\left[y_i - \frac{\sum_{n\in N:n\neq i} K(x_n, x_i)y_n}{\sum_{n\in N:n\neq i} K(x_n, x_i)}\right]\right]$$

$$\frac{\partial \mathcal{L}(W)}{\partial W_{11}} = -\sum_{i=1}^{N}\left[\left(y_i - \frac{\sum_{n\in N:n\neq i} K(x_n, x_i)y_n}{\sum_{n\in N:n\neq i} K(x_n, x_i)}\right)\frac{\partial}{\partial W_{11}}\left[\frac{\sum_{n\in N:n\neq i} K(x_n, x_i)y_n}{\sum_{n\in N:n\neq i} K(x_n, x_i)}\right]\right]$$

Then we may apply the quotient rule to the gradient term:

$$\frac{\partial}{\partial W_{11}}\left[\frac{\sum_{n\in N:n\neq i} K(x_n, x_i)y_n}{\sum_{n\in N:n\neq i} K(x_n, x_i)}\right] =$$

$$\frac{(\sum_{n\in N:n\neq i} K(x_n, x_i))(\sum_{n\in N:n\neq i} \frac{\partial}{\partial W_{11}}K(x_n, x_i)y_n) - (\sum_{n\in N:n\neq i} K(x_n, x_i)y_n)(\sum_{n\in N:n\neq i} \frac{\partial}{\partial W_{11}}K(x_n, x_i))}{(\sum_{n\in N:n\neq i} K(x_n, x_i))^2}$$

Observe that we can apply the same procedure to the two other gradients with respect to W_{12} and W_{22}. That is, we can also apply the chain rule and the quotient rule on the other two gradients to write them as:

$$\frac{\partial \mathcal{L}(W)}{\partial W_{12}} = -\sum_{i=1}^{N}\left[\left(y_i - \frac{\sum_{n\in N:n\neq i} K(x_n, x_i)y_n}{\sum_{n\in N:n\neq i} K(x_n, x_i)}\right)\frac{\partial}{\partial W_{12}}\left[\frac{\sum_{n\in N:n\neq i} K(x_n, x_i)y_n}{\sum_{n\in N:n\neq i} K(x_n, x_i)}\right]\right]$$

$$\frac{\partial}{\partial W_{12}}\left[\frac{\sum_{n\in N:n\neq i} K(x_n, x_i)y_n}{\sum_{n\in N:n\neq i} K(x_n, x_i)}\right] =$$

$$\frac{(\sum_{n\in N:n\neq i} K(x_n, x_i))(\sum_{n\in N:n\neq i} \frac{\partial}{\partial W_{12}}K(x_n, x_i)y_n) - (\sum_{n\in N:n\neq i} K(x_n, x_i)y_n)(\sum_{n\in N:n\neq i} \frac{\partial}{\partial W_{12}}K(x_n, x_i))}{(\sum_{n\in N:n\neq i} K(x_n, x_i))^2}$$

$$\frac{\partial \mathcal{L}(W)}{\partial W_{22}} = -\sum_{i=1}^{N}\left[\Big(y_i - \frac{\sum_{n\in N:n\neq i} K(x_n,x_i)y_n}{\sum_{n\in N:n\neq i} K(x_n,x_i)}\Big)\frac{\partial}{\partial W_{22}}\Big[\frac{\sum_{n\in N:n\neq i} K(x_n,x_i)y_n}{\sum_{n\in N:n\neq i} K(x_n,x_i)}\Big]\right]$$

$$\frac{\partial}{\partial W_{22}}\Big[\frac{\sum_{n\in N:n\neq i} K(x_n,x_i)y_n}{\sum_{n\in N:n\neq i} K(x_n,x_i)}\Big] =$$

$$\frac{(\sum_{n\in N:n\neq i} K(x_n,x_i))(\sum_{n\in N:n\neq i} \frac{\partial}{\partial W_{22}}K(x_n,x_i)y_n) - (\sum_{n\in N:n\neq i} K(x_n,x_i)y_n)(\sum_{n\in N:n\neq i} \frac{\partial}{\partial W_{22}}K(x_n,x_i))}{(\sum_{n\in N:n\neq i} K(x_n,x_i))^2}$$

Thus, in order to find our three gradients, we need to find:

$$\frac{\partial}{\partial W_{11}}\Big[\frac{\sum_{n\in N:n\neq i} K(x_n,x_i)y_n}{\sum_{n\in N:n\neq i} K(x_n,x_i)}\Big], \quad \frac{\partial}{\partial W_{12}}\Big[\frac{\sum_{n\in N:n\neq i} K(x_n,x_i)y_n}{\sum_{n\in N:n\neq i} K(x_n,x_i)}\Big], \quad \text{and} \quad \frac{\partial}{\partial W_{22}}\Big[\frac{\sum_{n\in N:n\neq i} K(x_n,x_i)y_n}{\sum_{n\in N:n\neq i} K(x_n,x_i)}\Big]$$

* *Please note, for ease of notation, we'll define*:

$$\sum \equiv \sum_{n\in N:n\neq i} \quad \text{and} \quad \gamma_{n,i} \equiv K(x_n,x_i) = \exp(-W_{11}x_{ni1}^2 - 2W_{12}x_{ni1}x_{ni2} - W_{22}x_{ni2}^2)$$

For W_{11}:

$$\frac{\partial}{\partial W_{11}}\Big[\frac{\sum K(x_n,x_i)y_n}{\sum K(x_n,x_i)}\Big] = \frac{\big(\sum \gamma_{n,i}\big)\frac{\partial}{\partial W_{11}}\big(\sum \gamma_{n,i}y_n\big) - \big(\sum \gamma_{n,i}y_n\big)\frac{\partial}{\partial W_{11}}\big(\sum \gamma_{n,i}\big)}{\big(\sum \gamma_{n,i}\big)^2}$$

To find a closed form analytical solution we need to solve for the quantity:

$$\frac{\partial}{\partial W_{11}}\gamma_{n,i} = \frac{\partial}{\partial W_{11}}\exp(-W_{11}x_{ni1}^2 - 2W_{12}x_{ni1}x_{ni2} - W_{22}x_{ni2}^2)$$

$$\frac{\partial}{\partial W_{11}}\gamma_{n,i} = -2x_{ni1}^2\exp(-W_{11}x_{ni1}^2 - 2W_{12}x_{ni1}x_{ni2} - W_{22}x_{ni2}^2) = -2x_{ni1}^2\gamma_{n,i}$$

So we have:

$$\frac{\partial}{\partial W_{11}}\Big[\frac{\sum K(x_n,x_i)y_n}{\sum K(x_n,x_i)}\Big] = \frac{\big(\sum \gamma_{n,i}\big)\big(\sum(-2x_{ni1}^2\gamma_{n,i})y_n\big) - \big(\sum \gamma_{n,i}y_n\big)\big(\sum(-2x_{ni1}^2\gamma_{n,i})\big)}{\big(\sum \gamma_{n,i}\big)^2}$$

So our gradient for W_{11} is:

$$\frac{\partial \mathcal{L}(W)}{\partial W_{11}} = -\sum_{i=1}^{N}\left[\Big(y_i - \frac{\sum \gamma_{n,i}y_n}{\sum \gamma_{n,i}}\Big)\frac{\partial}{\partial W_{11}}\Big[\frac{\sum_{n\in N:n\neq i} K(x_n,x_i)y_n}{\sum_{n\in N:n\neq i} K(x_n,x_i)}\Big]\right]$$

$$\frac{\partial \mathcal{L}(W)}{\partial W_{11}} = -\sum_{i=1}^{N}\left[\Big(y_i - \frac{\sum \gamma_{n,i}y_n}{\sum \gamma_{n,i}}\Big)\Big(\frac{\big(\sum \gamma_{n,i}\big)\big(\sum(-2x_{ni1}^2\gamma_{n,i})y_n\big) - \big(\sum \gamma_{n,i}y_n\big)\big(\sum(-2x_{ni1}^2\gamma)\big)}{\big(\sum \gamma_{n,i}\big)^2}\Big)\right]$$

$$\frac{\partial \mathcal{L}(W)}{\partial W_{11}} = \sum_{i=1}^{N}\left[\Big(y_i - \frac{\sum \gamma_{n,i}y_n}{\sum \gamma_{n,i}}\Big)\Big(\frac{\big(\sum \gamma_{n,i}\big)\big(\sum 2x_{ni1}^2\gamma_{n,i}y_n\big) - \big(\sum \gamma_{n,i}y_n\big)\big(\sum 2x_{ni1}^2\gamma_{n,i}\big)}{\big(\sum \gamma_{n,i}\big)^2}\Big)\right]$$

Similarly, for W_{12} we have:

$$\frac{\partial}{\partial W_{12}}\Big[\frac{\sum K(x_n,x_i)y_n}{\sum K(x_n,x_i)}\Big] = \frac{\big(\sum \gamma_{n,i}\big)\frac{\partial}{\partial W_{12}}\big(\sum \gamma_{n,i}y_n\big) - \big(\sum \gamma_{n,i}y_n\big)\frac{\partial}{\partial W_{12}}\big(\sum \gamma_{n,i}\big)}{\big(\sum \gamma_{n,i}\big)^2}$$

To find a closed form analytical solution we need to solve for the quantity:

$$\frac{\partial}{\partial W_{12}}\gamma_{n,i} = \frac{\partial}{\partial W_{12}}\exp(-W_{11}x_{ni1}^2 - 2W_{12}x_{ni1}x_{ni2} - W_{22}x_{ni2}^2)$$

$$\frac{\partial}{\partial W_{12}}\gamma_{n,i} = -2x_{ni1}x_{ni2}\exp(-W_{11}x_{ni1}^2 - 2W_{12}x_{ni1}x_{ni2} - W_{22}x_{ni2}^2) = -2x_{ni1}x_{ni2}\gamma_{n,i}$$

So we have:

$$\frac{\partial}{\partial W_{12}}\Big[\frac{\sum K(x_n,x_i)y_n}{\sum K(x_n,x_i)}\Big] = \frac{\big(\sum\gamma_{n,i}\big)\big(\sum(-2x_{ni1}x_{ni2}\gamma_{n,i})y_n\big) - \big(\sum\gamma_{n,i}y_n\big)\big(\sum(-2x_{ni1}x_{ni2}\gamma_{n,i})\big)}{\big(\sum\gamma_{n,i}\big)^2}$$

So our gradient for W_{12} is:

$$\frac{\partial\mathcal{L}(W)}{\partial W_{12}} = -\sum_{i=1}^{N}\left[\Big(y_i - \frac{\sum\gamma_{n,i}y_n}{\sum\gamma_{n,i}}\Big)\frac{\partial}{\partial W_{12}}\Big[\frac{\sum_{n\in N:n\neq i}K(x_n,x_i)y_n}{\sum_{n\in N:n\neq i}K(x_n,x_i)}\Big]\right]$$

$$\frac{\partial\mathcal{L}(W)}{\partial W_{12}} = -\sum_{i=1}^{N}\left[\Big(y_i - \frac{\sum\gamma_{n,i}y_n}{\sum\gamma_{n,i}}\Big)\Big(\frac{\big(\sum\gamma_{n,i}\big)\big(\sum(-2x_{ni1}x_{ni2}\gamma_{n,i})y_n\big) - \big(\sum\gamma_{n,i}y_n\big)\big(\sum(-2x_{ni1}x_{ni2}\gamma_{n,i})\big)}{\big(\sum\gamma_{n,i}\big)^2}\Big)\right]$$

$$\frac{\partial\mathcal{L}(W)}{\partial W_{12}} = \sum_{i=1}^{N}\left[\Big(y_i - \frac{\sum\gamma_{n,i}y_n}{\sum\gamma_{n,i}}\Big)\Big(\frac{\big(\sum\gamma_{n,i}\big)\big(\sum 2x_{ni1}x_{ni2}\gamma_{n,i}y_n\big) - \big(\sum\gamma_{n,i}y_n\big)\big(\sum 2x_{ni1}x_{ni2}\gamma_{n,i}\big)}{\big(\sum\gamma_{n,i}\big)^2}\Big)\right]$$

Similarly for W_{22} we have:

$$\frac{\partial}{\partial W_{22}}\Big[\frac{\sum K(x_n,x_i)y_n}{\sum K(x_n,x_i)}\Big] = \frac{\big(\sum\gamma_{n,i}\big)\frac{\partial}{\partial W_{22}}\big(\sum\gamma_{n,i}y_n\big) - \big(\sum\gamma_{n,i}y_n\big)\frac{\partial}{\partial W_{22}}\big(\sum\gamma_{n,i}\big)}{\big(\sum\gamma_{n,i}\big)^2}$$

To find a closed form analytical solution we need to solve for the quantity:

$$\frac{\partial}{\partial W_{22}}\gamma_{n,i} = \frac{\partial}{\partial W_{22}}\exp(-W_{11}x_{ni1}^2 - 2W_{12}x_{ni1}x_{ni2} - W_{22}x_{ni2}^2)$$

$$\frac{\partial}{\partial W_{22}}\gamma_{n,i} = -2x_{ni2}^2\exp(-W_{11}x_{ni1}^2 - 2W_{12}x_{ni1}x_{ni2} - W_{22}x_{ni2}^2) = -2x_{ni2}^2\gamma_{n,i}$$

So we have:

$$\frac{\partial}{\partial W_{22}}\Big[\frac{\sum K(x_n,x_i)y_n}{\sum K(x_n,x_i)}\Big] = \frac{\big(\sum\gamma_{n,i}\big)\big(\sum(-2x_{ni2}^2\gamma_{n,i})y_n\big) - \big(\sum\gamma_{n,i}y_n\big)\big(\sum(-2x_{ni2}^2\gamma_{n,i})\big)}{\big(\sum\gamma_{n,i}\big)^2}$$

So our gradient for W_{22} is:

$$\frac{\partial\mathcal{L}(W)}{\partial W_{22}} = -\sum_{i=1}^{N}\left[\Big(y_i - \frac{\sum\gamma_{n,i}y_n}{\sum\gamma_{n,i}}\Big)\frac{\partial}{\partial W_{22}}\Big[\frac{\sum_{n\in N:n\neq i}K(x_n,x_i)y_n}{\sum_{n\in N:n\neq i}K(x_n,x_i)}\Big]\right]$$

$$\frac{\partial\mathcal{L}(W)}{\partial W_{22}} = -\sum_{i=1}^{N}\left[\Big(y_i - \frac{\sum\gamma_{n,i}y_n}{\sum\gamma_{n,i}}\Big)\Big(\frac{\big(\sum\gamma_{n,i}\big)\big(\sum(-2x_{ni2}^2\gamma_{n,i})y_n\big) - \big(\sum\gamma_{n,i}y_n\big)\big(\sum(-2x_{ni2}^2\gamma_{n,i})\big)}{\big(\sum\gamma_{n,i}\big)^2}\Big)\right]$$

$$\frac{\partial\mathcal{L}(W)}{\partial W_{22}} = \sum_{i=1}^{N}\left[\Big(y_i - \frac{\sum\gamma_{n,i}y_n}{\sum\gamma_{n,i}}\Big)\Big(\frac{\big(\sum\gamma_{n,i}\big)\big(\sum 2x_{ni2}^2\gamma_{n,i}y_n\big) - \big(\sum\gamma_{n,i}y_n\big)\big(\sum 2x_{ni2}^2\gamma_{n,i}\big)}{\big(\sum\gamma_{n,i}\big)^2}\Big)\right]$$

4. The last kernel $W_3 = \alpha \begin{bmatrix} 1 & 0 \\ 0 & 0.1 \end{bmatrix}$ performs the best in our simulation because it yields the smallest overall loss. This makes intuitive sense because if we observe the data, our $x1$ features are perfectly collinear (identical) to our response variable y, whereas $x2$ is not. With this particular data set, we could perfectly predict y with just $x1$. Therefore, we expect that the kernel W_3 performs the best in terms of our loss function, because it weights deviations in $x1$ features ten times as heavily as it weights deviations in $x2$, and therefore gives a preferences to minimizing deviations in $x1$, thus producing the best predictions and the lowest overall loss. In comparison, $W1$ weights both $x1$ and $x2$ features equally, while $W2$ does the opposite of $W3$ weighting deviations in $x2$ ten times as heavily as deviations in $x1$. The choice of the parameter α affects the magnitude of our losses for the three different kernels. For smaller α, our losses tend towards 1, whereas for larger α our losses will tend to diverge from 1, either towards 0 or towards infinity. In particular, when we reduced our α from 10 to 1, we saw the losses for the $W1$ and $W3$ kernel rise while the loss for the $W2$ kernel decreased. This makes intuitive sense, as a larger α will cause our weight matrix W to affect our loss function more, since the weight matrix is scaled up by a factor of α in our loss functions.

5. We find that our gradient descent will approach the kernel:

$$W = \alpha \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$$

so that all the weighting is put on the first feature and no weighting is put on the second feature. This makes intuitive sense, since gradient descent is numerically optimizing our loss function, and because our first feature is perfectly collinear with our response variable, the gradient descent will tend towards this kernel so that it can perfectly predict the response variable, therefore approaching kernel W given above.