EE511 Machine Learning, Winter 2018: Homework 1

Due: Wednesday, Janunary 17^{th}, beginning of class

1 MLE [25 points]

This question uses a probability distribution called the Poisson distribution. A discrete random variable X follows a Poisson distribution with parameter λ if

$$\Pr(X = k) = \frac{\lambda^k}{k!} e^{-\lambda}, k \in \{0, 1, 2, \dots\}$$

Suppose the number of eggs laid by a turtle follows the Poisson distribution with parameter λ. Also suppose that all turtles are mutually independent. Once laid, every egg hatches after a while. When ready, hatchlings tear their shells apart and dig through the sand. When they reach the surface, they head towards the sea. Due to opportunist predators, the probability that a hatchling will reach the sea is p. Assume that each egg hatches at a different time, so that the survival of each hatchling is independent from the others. A biologist studies 10 of these turtles, observing the number of eggs laid by the turtle and the number of hatchlings that reach the sea. She records her observations in the following table.

Turtle	1	2	3	4	5	6	7	8	9	10
Eggs Laid	8	9	6	4	1	5	2	12	9	7
Hatchlings that Reached the Sea	5	6	4	3	0	5	2	9	8	6

We will establish some notation for the general case. Let $E = (E_1, \dots, E_n)$ be a random vector corresponding the number of eggs laid by each of n turtles. Let $H = (H_1, \dots, H_n)$ be a random vector corresponding to the number of hatchlings that reached the sea for each turtle. (The above table gives realizations of E and H.) Compute

1. *(8 points)* the joint log-likelihood function of E and H given λ and p.
2. *(12 points)* the MLE for λ and p in the general case.
3. *(5 points)* the MLE for λ and p using the observed values of E and H.

2 Regularization Constants [15 points]

We have discussed the importance of regularization as a technique to avoid overfitting our models. For linear regression, we have mentioned both LASSO (which uses the L_1 norm as a penalty), and ridge regression (which uses the squared L_2 norm as a penalty). In practice, the scaling factor of these penalties has a significant impact on the behavior of these methods, and must often be chosen empirically for a particular dataset. In this problem, we look at what happens when we choose our regularization factor poorly.

2.1 LASSO Regression

Recall that the loss function to be optimized under LASSO regression is:

$$E_L = \sum_{i=1}^{n} (y_i - (\hat{w}_0 + x^{(i)}\hat{w}))^2 + \lambda \|\hat{w}\|_1$$

where

$$\lambda\|\hat{w}\|_1 = \lambda\sum_{i=1}^{d}|\hat{w}_i| \tag{1}$$

and λ is our regularization constant.

1. Suppose our λ is much too small; that is,

   $$\sum_{i=1}^{n}(y_i - X\hat{w})^2 + \lambda\|w\|_1 \approx \sum_{i=1}^{n}(y_i - X\hat{w})^2$$

   How will this affect the magnitude of:

   (a) *(2 point)* The error on the training set?

   (b) *(2 point)* The error on the testing set?

   (c) *(2 point)* $\hat{w}$?

   (d) *(2 point)* The number of nonzero elements of $\hat{w}$?

2. Suppose instead that we overestimated on our selection of λ. What do we expect to be the magnitude of:

   (a) *(2 point)* The error on the training set?

   (b) *(2 point)* The error on the testing set?

   (c) *(2 point)* $\hat{w}$?

   (d) *(2 point)* The number of nonzero elements of $\hat{w}$?

2.2 Ridge Regression

Recall that the loss function to be optimized under ridge regression is now:

$$E_R = \sum_{i=1}^{n}(y_i - (\hat{w}_0 + x^{(i)}\hat{w}))^2 + \lambda\|\hat{w}\|_2^2$$

where

$$\lambda\|\hat{w}\|_2^2 = \lambda\sum_{i=1}^{d}(\hat{w}_i)^2 \tag{2}$$

If λ is too small, then the misbehavior of ridge regression is similar to that of LASSO in the previous section. Let's suppose λ has been set too high, and compare the behavior of ridge regression to LASSO.

To make this comparison, let's look at what happens to a particular feature weight $\hat{w}_i$. Since $\hat{w}$ is the solution that minimizes our error function E, it must be the case that $\frac{\partial E}{\partial \hat{w}_i} = 0$.

1. *(2 points)* Suppose we use the LASSO loss function E_L as defined in the previous section. Let's ignore the first term (which corresponds to the Sum Squared Error of our prediction), and calculate the partial derivative of the penalty term in Equation 1 with respect to $\hat{w}_i$. This can be thought of as the direction that LASSO "pushes" us away from the SSE solution. *Note that the absolute value is not differentiable at $\hat{w}_i = 0$, so you only need to answer for the case that $\hat{w}_i \neq 0$.*

2. *(2 point)* Instead, suppose we use the ridge regression loss function E_R from above. Again, ignore the SSE term and calculate the partial derivative with respect to $\hat{w}_i$ of Equation 2 to find how ridge regression "pushes" us away from the SSE solution.

3. *(3 points)* Comparing these two derivatives, for what values of $\hat{w}_i$ will their behaviors differ? What does this mean for our estimate of $\hat{w}_i$ when λ is very large?

3 Programming Question [60 points]

1. Load the data from the AmesHousing.txt file. There should be 2931 rows.

   The file can be downloaded from `https://ww2.amstat.org/publications/jse/v19n3/decock/AmesHousing.txt`. A description of the data is available at `https://ww2.amstat.org/publications/jse/v19n3/Decock/DataDocumentation.txt`.

2. Preprocessing:

   There are some missing values in this data. Replace all the missing values for numerical features with zeros and for categorical features use a special string to indicate a missing value.

   The numerical features are:

```
numerical_variables = ['Lot Area', 'Lot Frontage', 'Year Built',
                        'Mas Vnr Area', 'BsmtFin SF 1', 'BsmtFin SF 2',
                        'Bsmt Unf SF', 'Total Bsmt SF', '1st Flr SF',
                        '2nd Flr SF', 'Low Qual Fin SF', 'Gr Liv Area',
                        'Garage Area', 'Wood Deck SF', 'Open Porch SF',
                        'Enclosed Porch', '3Ssn Porch', 'Screen Porch',
                        'Pool Area']
```

   The categorical features are:

```
discrete_variables = ['MS SubClass', 'MS Zoning', 'Street',
                       'Alley', 'Lot Shape', 'Land Contour',
                       'Utilities', 'Lot Config', 'Land Slope',
                       'Neighborhood', 'Condition 1', 'Condition 2',
                       'Bldg Type', 'House Style', 'Overall Qual',
                       'Overall Cond', 'Roof Style', 'Roof Matl',
                       'Exterior 1st', 'Exterior 2nd', 'Mas Vnr Type',
                       'Exter Qual', 'Exter Cond', 'Foundation',
                       'Bsmt Qual', 'Bsmt Cond', 'Bsmt Exposure',
                       'BsmtFin Type 1', 'Heating', 'Heating QC',
                       'Central Air', 'Electrical', 'Bsmt Full Bath',
                       'Bsmt Half Bath', 'Full Bath', 'Half Bath',
                       'Bedroom AbvGr', 'Kitchen AbvGr', 'Kitchen Qual',
                       'TotRms AbvGrd', 'Functional', 'Fireplaces',
                       'Fireplace Qu', 'Garage Type', 'Garage Cars',
                       'Garage Qual', 'Garage Cond', 'Paved Drive',
                       'Pool QC', 'Fence', 'Sale Type', 'Sale Condition']
```

3. Split the data into train, validation and test sets. We will do this by using the Order column in the data file. For the validation set take the examples where the Order mod 5 is 3 and for the test set use the examples where the Order mod 5 is 4. The rest is for training.

4. Now let's do a simple one variable least squares linear regression as a warm-up. Predict the sale price based on the "Gr Liv Area" feature. Make a scatter plot of this feature vs. the sale price using the training data and overlay the line from your model. What is the equation for the line that you found?

   Apply the model to the data from the validation set and compute the root mean sequared error (RMSE).

5. Now that we have our simple model working, let's add more features.

   First, transform the categorical features to a one-hot encoding so that they can be used in the model. For example, the "Alley" column can take on three possible values "Pave", "Grvl", and "Missing". This will become a 3-dimensional one-hot vector.

Once the categorical features have been transformed, concatenate them with the numerical features and train a new model. Compute the RMSE on the validation set for this model.

6. We can improve the model by using L1 regularization, i.e. penalizing the absolute value of the coefficients. When L1 regularization is used for linear regression it is called Lasso Regression. We need to use cross-validation to select the value of alpha, which is the weight on the L1 regularization term.

   The first thing to do here is to normalize the features by subtracting the mean and dividing by the standard deviation. Be sure to use the mean and variance estimated from the training data when normalizing the validation (and test) data.

   Using cross validation, train models for values of alpha ranging from 50 to 500 in intervals of 50. Make a plot of the RMSE on the train and vaidation sets for each value of alpha. Which setting has the lowest error on the validation set? Briefly explain the concept of over-fitting and how this graph can be used to detect it.

   Plot the number of non-zero coefficients in the model for each of the values of alpha that you tried.

7. Now it's time to use the test data. Take the single variable model you trained, the least squares model that uses all of the variables, and the regularized model that you selected and apply them to the test data. Make a table and report the RMSE for each condition for both the validation set and the test set.