# Mathematical Foundations of Machine Learning
## Assignment 1

**David Felipe Alvear Goyes**

**1. Validation.**
Five different models are fit using the same training data set, and tested on the same (separate) test set (which has the same size as the training set). The RMS prediction errors for each model, on the training and test sets, are reported below. Comment briefly on the results for each model. You might mention whether the model's predictions are good or bad, whether it is likely to generalize to unseen data, or whether it is over-fit. You are also welcome to say that you don't believe the results, or think the reported numbers are fishy.

| Model | Train RMS | Test RMS |
|---|---|---|
| A | 1.355 | 1.423 |
| B | 9.760 | 9.165 |
| C | 5.033 | 0.889 |
| D | 0.211 | 5.072 |
| E | 0.633 | 0.633 |

## Solution

1. Comment on each model based on the results:
   - → good/bad Predictions
   - → likely to generalize unseen data
   - → overfit
   - → fishy numbers

| Model | Train RMS | Test RMS |
|---|---|---|
| A | 1.355 | 1.423 |

→ Model with good prediction results. Likely to generalize unseen data.

| Model | Train RMS | Test RMS |
|---|---|---|
| B | 9.760 | 9.165 |

→ Model with reasonable results, but with high RMS error

| Model | Train RMS | Test RMS |
|---|---|---|
| C | 5.033 | 0.889 |

→ Not reasonable results, train error greater than test results. Suspicious results.

| Model | Train RMS | Test RMS |
|---|---|---|
| D | 0.211 | 5.072 |

→ Model with bad prediction results also overfitting.

| Model | Train RMS | Test RMS |
|---|---|---|
| E | 0.633 | 0.633 |

→ Model with good results, likely to generalize unseen data, But rare that both errors are equal.

**2. Complexity of cross-validation.**
The cost of fitting a model with $D$ features and $N$ data points using $QR$ factorization is $2ND^2$ flops. In this exercise we explore the complexity of carrying out 10-fold cross-validation on the same data set. We divide the data set into 10 folds, each with $N/10$ data points. The naïve method is to fit 10 different models, each using 9 of the folds, using the $QR$ factorization, which requires $10 \cdot 2(0.9)ND^2 = 18ND^2$ flops. So the naïve method of carrying out 10-fold cross-validation requires, not surprisingly, around $10\times$ the number of flops as fitting a single model.

The method below outlines another method to carry out 10-fold cross-validation. Give the total flop count for each step, keeping only the dominant terms, and compare the total cost of the method to that of the naïve method. Let $A_1, \ldots, A_{10}$ denote the $(N/10) \times D$ blocks of the data matrix associated with the folds, and let $b_1, \ldots, b_{10}$ denote the right-hand sides in the least squares fitting problem.

- form the matrices $G_i = A_i^T A_i$ and the vectors $c_i = A_i^T b_i$.
- form $G = G_1 + \cdots + G_{10}$ and $c = c_1 + \cdots + c_{10}$.
- for $k = 1, \ldots, 10$, compute $\theta_k = (G - G_k)^{-1}(c - c_k)$.

Data: $D$ Features, $N$ data points

10-fold cross validation $\rightarrow$ 1-fold $N/10$ data points

QR-factorization $\rightarrow$ $2ND^2$ flops

9-fold QR fact $\rightarrow$ $10 \cdot 2(0.9)ND^2 = 18ND^2$ flops

$\rightarrow$ Compute total flop count for:

Let $A_1, \ldots A_{10}$ data matrix associated with the folds $\quad A_i \rightarrow (N/10) \times D$

$A_i \rightarrow n \times D$
where $n = N/10$

$\rightarrow$ Least squares fitting Problem

$Ax = b \rightarrow$ A: data
X: Parameters
b: target variable

1. Form $G_i = A_i^T A_i \rightarrow A^T \in \mathbb{R}^{D \times n}, A \in \mathbb{R}^{n \times D} \rightarrow G_i \in \mathbb{R}^{D \times D} \quad O(2nD^2)$

$C_i = A_i^T b_i \rightarrow A^T \in \mathbb{R}^{D \times n}, b_i \in \mathbb{R}^{n \times 1} \rightarrow C_i \in \mathbb{R}^{D \times 1} \quad O(2nD)$

Given that we have 10 $A_i$, the complexity for $G_i, b_i$ are:

$\rightarrow G_i$ for $i = \{1, \ldots 10\}$ $\rightarrow$ flops: $10 \times 2nD^2$

$\rightarrow C_i$ for $i = \{1, \ldots 10\}$ $\rightarrow$ flops: $10 \times 2nD$

2. Form $G = G_1 + G_2 + \cdots + G_{10}$ $\rightarrow$ Addition $G_i$ matrices flops $= 9D^2$

$C = C_1 + C_2 + \cdots + C_{10}$ $\rightarrow$ Addition $C_i$ vectors flops $= 9D$

3. For $K = 1 \ldots 10$ Compute $\theta_K = (G - G_K)^{-1}(C - C_K)$ $\rightarrow$ flops $= D^3 + D^2 +$

- Inverse matrix complexity $O(D^3)$
- $G - G_K$ matrix difference $O(D^2)$
- $C - C_K$ vector difference $O(D)$
- $(G - G_K)^{-1}(C - C_K)$ multiplication $O(2D^2)$

flops $= D^3 + D^2 + D + 2D^2$

It's computed 10 times $\rightarrow$ flops $= 10(D^3 + 3D^2 + D)$

flops $= 10D^3 + 30D^2 + 10D$

Somming all steps:

flops $= 10 \times 2nD^2 + 10 \times 2nD + 9D^2 + 9D + 10D^3 + 30D^2 + 10D$

flops $= 2ND^2 + 2ND + 10D^3 + 39D^2 + 19D$

Since we need to look to Dominant terms

$$\boxed{\text{flops} \approx 2ND^2 + 10D^3}$$

If D is small but N is large, the naive method might be more computationally intensive due to $18ND^2$ term. But if D is large the $10D^3$ term in the alternative method could dominate resulting in high computational complexity.

**3. Interpretation of model parameters.**
Suppose that the $N$ feature vectors $x_1, \dots, x_N$ are word count histograms, and the labels $y_1, ..., y_N$ give the document authors (as one of $1, \dots, K$). A classifier guesses which of the $K$ authors wrote an unseen document, a task called authorship attribution. A least squares classifier using regression is fit to the data, resulting in the classifier

$$\hat{f}(x) = \underset{k=1,\dots,K}{\operatorname{argmax}}(w_k^T x + b_k)$$

For each author (i.e., $k = 1, \dots, K$) we find the ten largest (most positive) entries in the vector $w_k$ and the ten smallest (most negative) entries. These correspond to two sets of ten words in the dictionary, for each author. Interpret these words, briefly, in English.

## Solution

$X = \{x_1, \cdots x_N\}$, $x_i \rightarrow$ Word Count histograms

$Y = \{y_1, \cdots y_N\}$, $y_i \rightarrow$ labels document authors

$x \rightarrow$ Unseen document

$\hat{f}(x) = \underset{K=1,\cdots K}{\operatorname{argmax}}(w_K^T x + b_K) \rightarrow$ Least squares classifier using regression

$W_K$ Represent the sensitivity of the prediction to a determined count histogram $X_i$. If we take the ten largest entries in $W_K$ we will take the most 10 significant or representative words that the author use in the documents. Similarly, taking the 10 smallest entries will give us the less representative words that the author use in the documents. with the most and less representative words for each author we can create a classifier to determine the authorship of a given unseen document.

# 4. Iris classification

Iris Dataset → 50 examples = 150
3 classes

→ Trainning set: 120 (40 Per class)
→ Test set: 30 (10 per class)

1. Generate $\hat{f}_k(x) = \text{sign}(w_k^T x + b_k)$ for each class
   Use the 3 LS boolean classifiers → 3-class classifier
2. generate 3x3 confusion matrix (train, test)
3. generate error rates in training and test data.