# ECON 570 Problem Set 1

Due: September 25, 2020

## 1 Cholesky Factorization

A. Recall that if $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ is symmetric and positive definite, then it can be factored as

$$\boldsymbol{A} = \boldsymbol{L}\boldsymbol{L}', \tag{1}$$

where $\boldsymbol{L}$ is lower triangular and nonsingular.

Assume that $\boldsymbol{A}$ is $2 \times 2$. Write down the system of equations implied by (1) and calculate the number of flops required to perform the Cholesky factorization in this case.

B. Extend the previous result and show that the complexity of the Cholesky factorization for the general case of a $n \times n$ matrix is $O(n^3)$.

## 2 Computational Complexity

A. Write custom functions for

1. Calculating the inner product of two vectors.
2. Calcuatling the product of two matrices.

B. Generate random $m \times n$ matrix $X$ and $n \times p$ matrix $Y$, for $m = 50, n = 100, p = 200$. Compute

1. The amount of time it takes to multiply $X$ and $Y$ using the custom code you wrote.
2. The amount of time it takes to multiply $X$ and $Y$ using built-in NumPy methods.

C. Now increase $m, n, p$ each by a factor of 10.

   1. How long do you expect it would take to multiply $X$ and $Y$ using your custom code? How long does it actually take?
   2. How long do you expect it would take to multiply $X$ and $Y$ using built-in NumPy methods? How long does it actually take?

D. Increase $m, n, p$ each by a factor of 10 again, and repeat the above but using NumPy's built-in methods only. How long does it take to multiply $X$ and $Y$? Did it increase by the same factor as it did before when all the dimensions were increased by a factor of 10? Why or why not?

E. Generate a $n \times p$ matrix and a $n$-vector $y$.

   1. Set $n = 5000, p = 200$. How long does it take to regress $y$ on $X$?
   2. Set $n = 50000, p = 200$. How long do you expect the same regression would take? How long does it actually take?
   3. Set $n = 5000, p = 2000$. How long do you expect the same regression would take? How long does it actually take?

# 3 Breast Cancer Data

Use the breast cancer data from `sklearn` to perform the following exercises.

A. Load the breast cancer data with the `load_breast_cancer` method from the module `sklearn.datasets`.

B. Standardize each feature in the data set.

C. Perform PCA on the standardized features. How many principle components must we keep to explain 90% of the total variance? How much variance is explained if we keep 2?

D. Perform $k$-means with $k = 2$ on the full set of features, and on the first 2 principle components only. Compare how well the clusters found by $k$-means in each of these cases compare to the true targets of the data set.

# 4 Olivetti Faces

Use the Olivetti faces data set available through `sklearn` to do the following.

A. Fetch and load the data with the `fetch_olivetti_faces` method from the module `sklearn.datasets`.

B. Demean each face in the data set (no need to divide by standard deviation as every dimension is a number between a fixed range representing a pixel).

C. Compute and display the first 9 eigenfaces.

D. In class we showed that any given face in the data set can be represented as a linear combination of the eigenfaces. For any face in the data set, show how it progresses as we combine $1, 51, 101, \ldots$ eigenfaces, until the full image is recovered.