1 Derivation of PCA

Assume we are given \( n \) training data points \((x_i, y_i)\). We collect the target values into \( y \in \mathbb{R}^n \), and the inputs into the matrix \( X \in \mathbb{R}^{n \times d} \) where the rows are the \( d \)-dimensional feature vectors \( x_i^\top \) corresponding to each training point. Furthermore, assume that the data has been centered such that \( \frac{1}{n} \sum_{i=1}^{n} x_i = 0 \), \( n > d \) and \( X \) has rank \( d \). The covariance matrix is given by

\[
\Sigma = \frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x})(x_i - \bar{x})^\top
\]

When \( \bar{x} = 0 \) (i.e., we have subtracted the mean in our samples), we obtain \( \Sigma = \frac{1}{n} X^\top X \).

(a) Maximum Projected Variance: We would like the vector \( w \) such that projecting your data onto \( w \) will retain the maximum amount of information, i.e., variance. We can formulate the optimization problem as

\[
\max_{w, \|w\|_2 = 1} \frac{1}{n} \sum_{i=1}^{n} (x_i^\top w)^2 = \max_{w, \|w\|_2 = 1} \frac{1}{n} w^\top X^\top X w. \tag{1}
\]

Show that the maximizer for this problem is equal to the eigenvector \( v_1 \) that corresponds to the largest eigenvalue \( \lambda_1 \) of \( \Sigma \). Also show that optimal value of this problem is equal to \( \lambda_1 \).

**Hint:** Use the spectral decomposition of \( \Sigma \) and consider reformulating the optimization problem using a new variable.
(b) Let us call the solution of the above part $w_1$. Next, we will use a greedy procedure to find the $i$th component of PCA by doing the following optimization

$$\begin{align*}
\text{maximize} & \quad w_i^\top X^\top Xw_i \\
\text{subject to} & \quad w_i^\top w_i = 1 \\
& \quad w_i^\top w_j = 0 \quad \forall j < i,
\end{align*}$$

(2)

where $w_j$, $j < i$ are defined recursively using the same maximization procedure above. Show that the maximizer for this problem is equal to the eigenvector $v_i$ that corresponds to the $i$th eigenvalue $\lambda_i$ of $\Sigma$. Also show that optimal value of this problem is equal to $\lambda_i$. 
2 Ridge regression vs. PCA

In this problem we want to compare two procedures: The first is ridge regression with hyperparameter $\lambda$, while the second is applying ordinary least squares after using PCA to reduce the feature dimension from $d$ to $k$ (we give this latter approach the short-hand name $k$-PCA-OLS where $k$ is the hyperparameter).

Notation: The singular value decomposition of $X$ reads $X = U \Sigma V^T$ where $U \in \mathbb{R}^{n \times n}$, $\Sigma \in \mathbb{R}^{n \times d}$ and $V \in \mathbb{R}^{d \times d}$. We denote by $u_i$ the $n$-dimensional column vectors of $U$ and by $v_i$ the $d$-dimensional column vectors of $V$. Furthermore the diagonal entries $\sigma_i = \Sigma_{i,i}$ of $\Sigma$ satisfy $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_d > 0$. For notational convenience, assume that $\sigma_i = 0$ for $i > d$.

(a) Consider running ridge regression with $\lambda > 0$ in the $V$-transformed coordinates, i.e.,

$$\hat{w}_{ridge} = \arg \min_w \|XVw - y\|_2^2 + \lambda \|w\|_2^2.$$ 

Note that this does not correspond to any dimensionality reduction, just a change of variables. It turns out that the solution in this case can be written as:

$$\hat{w}_{ridge} = \left[\begin{array}{cccc}
\text{diag} & \left(\begin{array}{c}
\frac{\sigma_1}{\lambda + \sigma_1^2}, & \ldots, & \frac{\sigma_d}{\lambda + \sigma_d^2}
\end{array}\right) & 0
\end{array}\right]U^T y. \quad (3)$$

Use $\hat{y}_{test} = x_{test}^T V \hat{w}_{ridge}$ to denote the resulting prediction for a hypothetical $x_{test}$. Using (3) and the appropriate scalar $\{\beta_i\}$, show that this prediction can be written as:

$$\hat{y}_{test} = x_{test}^T \sum_{i=1}^d v_i \beta_i u_i^T y. \quad (4)$$

(b) Suppose that we do $k$-PCA-OLS — i.e. ordinary least squares on the reduced $k$-dimensional feature space obtained by projecting the raw feature vectors onto the $k < d$ principal components of $\Sigma$. Use $\hat{y}_{test}$ to denote the resulting prediction for a hypothetical $x_{test}$. 

DIS5, ©UCB CS 189/289A, Fall 2021. All Rights Reserved. This may not be publicly shared without explicit permission.
It turns out that the learned k-PCA-OLS predictor can also be written as:

$$\hat{y}_{test} = x_{test}^T \sum_{i=1}^{d} v_i \beta_i \beta_i^T y.$$  \hspace{1cm} (5)

What are the $\beta_i \in \mathbb{R}$ coefficients in this case?

*Hint:* Some of these $\beta_i$ will be zero.

(c) Compare $\hat{y}_{PCA}$ with $\hat{y}_{ridge}$ (at different $\lambda$), how do you find their relationship?