ECBM E6040 Neural Networks and Deep Learning

Lecture #2: Elements of Linear Algebra

Instructor Zoran Kostic

Columbia University
Department of Electrical Engineering

September 6, 2016

Outline of Part I

2. Summary of the Previous Lecture
   - Topics Covered
   - Learning Objectives

Outline of Part II

3. Elements of Linear Algebra
   - Finite Dimensional Vector Spaces
   - Eigendecomposition and SVD
   - Principal Components Analysis

Part I

Review of Previous Lecture

Topics Covered

- Logistics
- Introduction to Neural Networks and Deep Learning
- Programming Tools and Computing Resources

Learning Objectives

- Neural Networks and Deep Learning: History, Role of GPUs, Expected Impact, Power and Limitations of Deep Learning
- Understanding how to use the Amazon Elastic Computing Cloud, Jupyter Notebooks and Git Repositories

Part II

Today's Lecture

Finite Dimensional Vector Spaces

Definition

A set E of elements is called a **vector space** (or a linear space, or a linear vector space) over $\mathbb{C}$ if we have a function $+$ on $E \times E$ to E and a function $\cdot$ on $\mathbb{C} \times E$ to E such that for all $x, y \in E$:

$$
\begin{aligned}
& x + y = y + x \\
& (x + y) + z = x + (y + z) \\
& x + 0 = x \\
& \alpha(x + y) = \alpha x + \alpha y \\
& (\alpha + \beta)x = \alpha x + \beta x \\
& \alpha(\beta x) = (\alpha\beta)x \\
& 0 \cdot x = 0 \; and \; 1 \cdot x = x
\end{aligned}
$$

We call $+$ the addition and $\cdot$ the multiplication by scalars.

Subspaces of a Vector Spaces

Definition

A nonempty subset S of the vector space E is a **subspace** or a **linear manifold** if $\alpha_1 x_1 + \alpha_2 x_2$ belongs to S whenever x_1 & x_2 do.

In what follows we shall assume for simplicity that $dim(E) = n$.

Definition

The span of $S \subset E$ is the subspace of all linear combinations of vectors in S, i.e.,

$$span(S) = \{\sum_{i=1}^{n} \alpha_i x_i | \alpha_i \in \mathbb{C}, x_i \in S\}.$$

Basis

Definition

A sequence $\{e_k\}_{k=1}^n$ in E is a basis for E if the following two conditions are satisfied:

(i) $E = span\{e_k\}_{k=1}^n$;

(ii) $\{e_k\}_{k=1}^n$ is linearly independent, i.e., if $\sum_{k=1}^n c_k e_k = 0$ for some scalar coefficients $\{c_k\}_{k=1}^n$, then $c_k = 0$ for all $k, k = 1, ..., n$.

Vectors in $E = \mathbb{R}^n$

Consider the vectors $\mathbf{x}, \mathbf{x} \in \mathbb{R}^n$, and $\mathbf{y}, \mathbf{y} \in \mathbb{R}^n$, and let $c, d \in \mathbb{R}$. Then any linear combination $c\mathbf{x} + d\mathbf{y} \in \mathbb{R}^n$.

Definition (Dot Product)

The dot product or inner product of $\mathbf{x} = (x_1, x_2, ..., x_n)$ and $\mathbf{y} = (y_1, y_2, ..., y_n)$ is given by

$$< \mathbf{x}, \mathbf{y} >= \mathbf{x}^T \cdot \mathbf{y} = \sum_{k=1}^{n} x_k y_k.$$

Definition (Length or Norm)

The length or norm of a vector $\mathbf{x}$ is given by

$$||\mathbf{x}|| = \sqrt{\mathbf{x}^T \cdot \mathbf{x}}.$$

Eigenvalues and Eigenvectors

Let $\mathbf{A}$ be an (n, n) square matrix. If

$$\mathbf{Ax} = \lambda \mathbf{x}$$

$\mathbf{x}$ is said to be an eigenvector of $\mathbf{A}$ and λ an eigenvalue.

Theorem

The eigenvalues are the solution of the equation

$$det(\mathbf{A} - \lambda \mathbf{I}) = 0$$

and the eigenvectors are in the nullspace of $(\mathbf{A} - \lambda \mathbf{I})$.

Diagonalizing a Matrix

Assume that the matrix $\mathbf{A}$ has n linearly independent eigenvectors $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n$. Let $\mathbf{S}$ be the matrix defined by $\mathbf{S} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n]$.

Theorem

$$\mathbf{A} = \mathbf{S}\boldsymbol{\Lambda}\mathbf{S}^{-1},$$

where

$$\boldsymbol{\Lambda} = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_n \end{bmatrix} = \mathbf{S}^{-1}\mathbf{A}\mathbf{S}.$$

Remark

Invertibility is concerned with eigenvalues. Diagonalizability is concerned with eigenvectors.

Symmetric Matrices

Assume that $\mathbf{A} = \mathbf{A^T}$, where T denotes the transpose. Then

- $\mathbf{A}$ has only real eigenvalues;
- The eigenvectors can be chosen to be orthonormal.

Theorem (Spectral Theorem)

Every symmetric matrix $\mathbf{A}$ can be written as

$$\mathbf{A} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T$$

with $\mathbf{\Lambda}$ having real eigenvalues, $\mathbf{Q}$ orthonormal eigenvectors and $\mathbf{Q}^{-1} = \mathbf{Q}^T$.

Bases and Singular Value Decomposition (SVD)

Let $\mathbf{A}$ be an (m, n) matrix, square or rectangle. Recall that if $\mathbf{A}$ is a diagonalizable square matrix the input and output bases are eigenvectors of $\mathbf{A}$ and

$$S^{-1}AS = \Lambda.$$

However, this factorization will not work if the matrix is non-diagonalizable (eigenvectors are dependent) or $m \neq n$.

We will use a different method of diagonalization. Assume that the row space of A is r-dimensional in $\mathbb{R}^n$. Its column space is also r-dimensional in $\mathbb{R}^m$. For SVD, the input and output bases are eigenvectors of the symmetric matrices AA^T and A^TA (both of rank r) and

$$U^{-1}AV = \Sigma.$$

AA^T and A^TA are (m, m) and (n, n), respectively.

Singular Value Decomposition

We consider the orthonormal basis $\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_n$ for the row space (in $\mathbb{R}^n$), and the orthonormal basis $\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_m$ for the column space (in $\mathbb{R}^m$). We require non-negative numbers σ_i such that

$$\mathbf{A}\mathbf{v}_i = \sigma_i \mathbf{u}_i, \text{ and } \mathbf{A}^T \mathbf{u}_i = \sigma_i \mathbf{v}_i, i = 1, 2, \ldots, r \ ,$$

that is, $\mathbf{A}\mathbf{v}_i$ is in the direction of $\mathbf{u}_i$. In matrix form this becomes

$$\mathbf{A}\mathbf{V} = \mathbf{U}\mathbf{\Sigma} \quad \text{or} \quad \mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T,$$

since by orthonormality $\mathbf{V}^{-1} = \mathbf{V}^T$. Here $\mathbf{\Sigma}$ is an (m, n) diagonal matrix with elements σ_i on the diagonal, $\mathbf{U}$ is an (m, m) matrix and $\mathbf{V}$ is an (n, n) matrix.

Singular Value Decomposition (cont'd)

Now note that

$$\mathbf{A}^T\mathbf{A} = (\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T)^T(\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T) = \mathbf{V}\boldsymbol{\Sigma}^T\boldsymbol{\Sigma}\mathbf{V}^T$$

or

$$\mathbf{A}^T\mathbf{A} = \mathbf{V}\begin{bmatrix} \sigma_1^2 & 0 & \dots & 0 & \dots & 0 \\ 0 & \sigma_2^2 & \dots & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma_r^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & \dots & 0 \end{bmatrix}\mathbf{V}^T.$$

Similarly

$$\mathbf{A}\mathbf{A}^T = \mathbf{U}\boldsymbol{\Sigma}\boldsymbol{\Sigma}^T\mathbf{U}^T.$$

Singular Value Decomposition (cont'd)

Note that the diagonal matrix $\mathbf{\Sigma}\mathbf{\Sigma}^T$ is an (m, m) matrix and $\mathbf{\Sigma}^T\mathbf{\Sigma}$ is an (n, n) matrix.

The singular values $\sigma_1^2, \sigma_2^2, ..., \sigma_r^2$ are the eigenvalues and the columns of $\mathbf{V}$ are the eigenvectors of $\mathbf{A}^T\mathbf{A}$.

The singular values $\sigma_1^2, \sigma_2^2, ..., \sigma_r^2$ are the eigenvalues and the columns of $\mathbf{U}$ are the eigenvectors of $\mathbf{A}\mathbf{A}^T$.

Note that r is the rank of the matrix $\mathbf{A}$.

Finally, note that all this works because $\mathbf{A}\mathbf{A}^T$ and $\mathbf{A}^T\mathbf{A}$ are (m, m) and (n, n) symmetric matrices, respectively.

Change of Basis

Recall that by choosing good bases

$$\mathbf{A}\mathbf{v}_i = \sigma_i \mathbf{u}_i,$$

where $\mathbf{v}_i \in \mathbb{R}^n$ and $\mathbf{u}_i \in \mathbb{R}^m$. Thus, $\mathbf{A}$ takes $\mathbf{v}_i$ in the row space and maps into $\sigma_i \mathbf{u}_i$ in the column space. We are interested in doing the opposite now, i.e.,

$$\mathbf{A}^{-1}\mathbf{u}_i = \mathbf{v}_i / \sigma_i.$$

However, $\mathbf{A}$ is an (m, n) matrix and therefore it does not have a proper inverse. We answer this question by essentially constructing an inverse on a subset of vectors.

Pseudo-Inverse

The pseudo-inverse is given by

$$\mathbf{A}^{+} = \mathbf{V}\mathbf{\Sigma}^{+}\mathbf{U}^{T}$$

or

$$\mathbf{A}^{+} = [\mathbf{v}_1 \dots \mathbf{v}_r \dots \mathbf{v}_n] \begin{bmatrix} \sigma_1^{-1} & 0 & \dots & 0 & \dots & 0 \\ 0 & \sigma_2^{-1} & \dots & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma_r^{-1} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & \dots & 0 \end{bmatrix} [\mathbf{u}_1 \dots \mathbf{u}_r \dots \mathbf{u}_m]^T$$

Pseudo-Inverse (cont'd)

The pseudo-inverse $\mathbf{A}^+$ is an (n, m) matrix. If $\mathbf{A}^{-1}$ exists, then

$$\mathbf{A}^+ = \mathbf{A}^{-1} = (\mathbf{U\Sigma V}^T)^{-1} = (\mathbf{V\Sigma}^{-1}\mathbf{U}^T).$$

Notice also that,

$$\mathbf{A}^+\mathbf{u}_i = \mathbf{v}_i/\sigma_i \quad \text{for} \quad i \leq r \quad \text{and} \quad \mathbf{A}^+\mathbf{u}_i = 0 \quad \text{for} \quad i > r.$$

Lemma

$\mathbf{AA}^+$ is the projection matrix onto the column space of $\mathbf{A}$. $\mathbf{A}^+\mathbf{A}$ is the projection matrix onto the row space of $\mathbf{A}$ and

$$\mathbf{AA}^+ = \mathbf{U\Sigma\Sigma}^+\mathbf{U}^T, \quad \mathbf{A}^+\mathbf{A} = \mathbf{V\Sigma}^+\mathbf{\Sigma V}^T,$$

where $\mathbf{u}, \mathbf{v}$ are the matrices $\mathbf{U}, \mathbf{V}$ restricted to their first r columns.

Pseudo-Inverse (cont'd)

A projection matrix $\mathbf{P}$ has the property that $\mathbf{P}^2 = \mathbf{P}$. Clearly

$$(\mathbf{A}\mathbf{A}^+)^2 = \mathbf{U}\mathbf{\Sigma}\mathbf{\Sigma}^+\mathbf{U}^T\mathbf{U}\mathbf{\Sigma}\mathbf{\Sigma}^+\mathbf{U}^T = \mathbf{U}(\mathbf{\Sigma}\mathbf{\Sigma}^+)^2\mathbf{U}^T = \mathbf{U}\mathbf{\Sigma}\mathbf{\Sigma}^+\mathbf{U}^T$$

and

$$(\mathbf{A}^+\mathbf{A})^2 = \mathbf{V}\mathbf{\Sigma}^+\mathbf{\Sigma}\mathbf{V}^T\mathbf{V}\mathbf{\Sigma}^+\mathbf{\Sigma}\mathbf{V}^T = \mathbf{V}(\mathbf{\Sigma}^+\mathbf{\Sigma})^2\mathbf{V}^T = \mathbf{V}\mathbf{\Sigma}^+\mathbf{\Sigma}\mathbf{V}^T.$$

Furthermore,

(i) $\mathbf{A}\mathbf{A}^+\mathbf{A} = \mathbf{A}$;
(ii) $\mathbf{A}^+\mathbf{A}\mathbf{A}^+ = \mathbf{A}^+$;
(iii) $(\mathbf{A}\mathbf{A}^+)^T = \mathbf{A}\mathbf{A}^+$;
(iv) $(\mathbf{A}^+\mathbf{A})^T = \mathbf{A}^+\mathbf{A}$;

Example

Let $\mathbf{A} = [1, 1]$. Then

- $\mathbf{A}^T\mathbf{A} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$.
- $\mathbf{A}^T\mathbf{A}$ has eigenvalues $\sigma_1^2 = 2$ and $\sigma_2^2 = 0$ with corresponding eigenvectors $\mathbf{v}_1 = [\sqrt{2}/2, \sqrt{2}/2]^T$ and $\mathbf{v}_2 = [-\sqrt{2}/2, \sqrt{2}/2]^T$, respectively.
- $\mathbf{A}\mathbf{A}^T = 2$ with eigenvalue $\sigma_1^2 = 2$ and eigenvector $\mathbf{u}_1 = 1$.
- The SVD and pseudoinverse of $\mathbf{A}$ are given by

$$\mathbf{A} = 1[\sqrt{2}, 0] \begin{bmatrix} \sqrt{2}/2 & -\sqrt{2}/2 \\ \sqrt{2}/2 & \sqrt{2}/2 \end{bmatrix}^T$$

$$\mathbf{A}^+ = \begin{bmatrix} \sqrt{2}/2 & -\sqrt{2}/2 \\ \sqrt{2}/2 & \sqrt{2}/2 \end{bmatrix} \begin{bmatrix} \sqrt{2}/2 \\ 0 \end{bmatrix} 1 = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}$$

Principal Component Analysis

A Simple Machine Learning Algorithm

Assume we have a collection of m points

$$\{\mathbf{x}^1, \mathbf{x}^2, ..., \mathbf{x}^m\} \in \mathbb{R}^n.$$

We would like to reduce the storage requirements without loosing too much precision.

Approach: Each point $\mathbf{x}^i \in \mathbb{R}^n$ is mapped into a code vector $\mathbf{c}^i \in \mathbb{R}^l$ with $l < n$. To make the decoder simple, a matrix $\mathbf{D} \in \mathbb{R}^{n \times l}$ is chosen and $\mathbf{Dc}$ is used to map back the code into $\mathbb{R}^n$.

The Optimal Code

To keep the encoding problem tractable, PCA constrains the columns of $\mathbf{D}$ to be orthogonal to each other. In addition, we shall assume that the columns of $\mathbf{D}$ have unit norm.

Lemma

Let $\mathbf{c}^*$ denote the optimal code for each input point $\mathbf{x}$, i.e.,

$$\mathbf{c}^* = \underset{\mathbf{c}}{\textit{arg min}} \, ||\mathbf{x} - \mathbf{D}\mathbf{c}||_2.$$

We have

$$\mathbf{c}^* = \mathbf{D}^T \mathbf{x}.$$

The Optimal Code (cont'd)

Proof

Note that

$$\mathbf{c}^* = \arg\min_{\mathbf{c}} ||\mathbf{x} - \mathbf{D}\mathbf{c}||_2^2.$$

Now since

$$||\mathbf{x}-\mathbf{D}\mathbf{c}||_2^2 = (\mathbf{x}-\mathbf{D}\mathbf{c})^T(\mathbf{x}-\mathbf{D}\mathbf{c}) = \mathbf{x}^T\mathbf{x}-\mathbf{c}^T\mathbf{D}^T\mathbf{x}-\mathbf{x}^T\mathbf{D}\mathbf{c}+\mathbf{c}^T\mathbf{D}^T\mathbf{D}\mathbf{c},$$

we obtain

$$||\mathbf{x} - \mathbf{D}\mathbf{c}||_2^2 = \mathbf{x}^T\mathbf{x} - 2\mathbf{x}^T\mathbf{D}\mathbf{c} + \mathbf{c}^T\mathbf{I}_l\mathbf{c},$$

and, therefore,

$$\mathbf{c}^* = \arg\min_{\mathbf{c}} [-2\mathbf{x}^T\mathbf{D}\mathbf{c} + \mathbf{c}^T\mathbf{c}].$$

The Optimal Code (cont'd)

Proof (cont'd)

This is a simple optimization problem that can be solved by computing the solution of the gradient equation:

$$\nabla \mathbf{c}[-2\mathbf{x}^T\mathbf{D}\mathbf{c} + \mathbf{c}^T\mathbf{c}] = \mathbf{0},$$

i.e.,

$$-2\mathbf{D}^T\mathbf{x} + 2\mathbf{c} = \mathbf{0}$$

or

$$\mathbf{c} = \mathbf{D}^T\mathbf{x}.$$

PCA Reconstruction

Finding the $\mathbf{D}$ Matrix

The PCA reconstruction operation amounts to computing $\mathbf{D}\mathbf{D}^T\mathbf{x}$. We now need to find an optimal encoding matrix $\mathbf{D}$.

Lemma

The decoding matrix $\mathbf{D}$ minimizes the Frobenius norm

$$\mathbf{D}^* = \arg\min_{\mathbf{D}} \sqrt{\sum_{i,j} [x_j^i - (\mathbf{D}\mathbf{D}^T\mathbf{x}^i)_j]^2}$$

subject to $\mathbf{D}^T\mathbf{D} = \mathbf{I}_l$ is given by the l eigenvectors corresponding to the largest eigenvalues of $\mathbf{X}^T\mathbf{X}$. Here, $\mathbf{X} \in \mathbb{R}^{m \times n}$ is the matrix defined by stacking all the vectors describing the m points such that $\mathbf{X}_{i:} = (\mathbf{x}^i)^T$.

Deriving the $\mathbf{D}$ Matrix

Proof

We derive the algorithm for finding $\mathbf{D}^*$ for the case $l = 1$. The case $l > 1$ can be obtained via induction. Here $\mathbf{D} = \mathbf{d}$, where $\mathbf{d}$ is a single vector. The minimization problem becomes

$$\mathbf{d}^* = \arg\min_{\mathbf{d}} \sum_i ||\mathbf{x}^i - \mathbf{d}\mathbf{d}^T\mathbf{x}^i||_2^2 = \arg\min_{\mathbf{d}} \sum_i ||\mathbf{x}^i - \mathbf{d}^T\mathbf{x}^i\mathbf{d}||_2^2$$

or

$$\mathbf{d}^* = \arg\min_{\mathbf{d}} \sum_i ||\mathbf{x}^i - (\mathbf{x}^i)^T\mathbf{d}\mathbf{d}||_2^2 = \arg\min_{\mathbf{d}} \sum_i ||(\mathbf{x}^i)^T - (\mathbf{x}^i)^T\mathbf{d}\mathbf{d}^T||_2^2,$$

subject to $||\mathbf{d}||_2 = 1$. With $\mathbf{X}_{i:} = (\mathbf{x}^i)^T$ the above minimization problem can be written in compact form as

$$\mathbf{d}^* = \arg\min_{\mathbf{d}} ||\mathbf{X} - \mathbf{X}\mathbf{d}\mathbf{d}^T||_F^2$$

subject to $\mathbf{d}^T\mathbf{d} = 1$.

Deriving the $\mathbf{D}$ Matrix (cont'd)

Proof

Now

$$||\mathbf{X} - \mathbf{X}\mathbf{d}\mathbf{d}^T||_F^2 = Tr[(\mathbf{X} - \mathbf{X}\mathbf{d}\mathbf{d}^T)^T(\mathbf{X} - \mathbf{X}\mathbf{d}\mathbf{d}^T)]$$

by the alternate definition of the Frobenius norm and the RHS can be written as

$$Tr(\mathbf{X}^T\mathbf{X} - \mathbf{X}^T\mathbf{X}\mathbf{d}\mathbf{d}^T - \mathbf{d}\mathbf{d}^T\mathbf{X}^T\mathbf{X} + \mathbf{d}\mathbf{d}^T\mathbf{X}^T\mathbf{X}\mathbf{d}\mathbf{d}^T) =$$

$$= Tr(\mathbf{X}^T\mathbf{X}) - Tr(\mathbf{X}^T\mathbf{X}\mathbf{d}\mathbf{d}^T) - Tr(\mathbf{d}\mathbf{d}^T\mathbf{X}^T\mathbf{X}) + Tr(\mathbf{d}\mathbf{d}^T\mathbf{X}^T\mathbf{X}\mathbf{d}\mathbf{d}^T)$$

$$= Tr(\mathbf{X}^T\mathbf{X}) - 2Tr(\mathbf{X}^T\mathbf{X}\mathbf{d}\mathbf{d}^T) + Tr(\mathbf{d}\mathbf{d}^T\mathbf{X}^T\mathbf{X}\mathbf{d}\mathbf{d}^T),$$

because we can cycle the order of the matrices inside the trace.

Deriving the $\mathbf{D}$ Matrix (cont'd)
Proof

The minimization problem can be now written as

$$\arg\min_{\mathbf{d}} \; -2Tr(\mathbf{X}^T\mathbf{X}\mathbf{d}\mathbf{d}^T) + Tr(\mathbf{d}\mathbf{d}^T\mathbf{X}^T\mathbf{X}\mathbf{d}\mathbf{d}^T) =$$

$$= \arg\min_{\mathbf{d}} \; -2Tr(\mathbf{X}^T\mathbf{X}\mathbf{d}\mathbf{d}^T) + Tr(\mathbf{X}^T\mathbf{X}\mathbf{d}\mathbf{d}^T\mathbf{d}\mathbf{d}^T)$$

because we can cycle the order the matrices inside a trace (again!). With the constraint $\mathbf{d}^T\mathbf{d} = 1$ the minimization os reduced to

$$\arg\min_{\mathbf{d}} \; -Tr(\mathbf{X}^T\mathbf{X}\mathbf{d}\mathbf{d}^T) = \arg\max_{\mathbf{d}} Tr(\mathbf{X}^T\mathbf{X}\mathbf{d}\mathbf{d}^T)$$

or finally

$$\arg\max_{\mathbf{d}} Tr(\mathbf{d}^T\mathbf{X}^T\mathbf{X}\mathbf{d})$$

subject to $\mathbf{d}^T\mathbf{d} = 1$.

Deriving the $\mathbf{D}$ Matrix (cont'd)

Proof

The optimization problem

$$\arg\max_{\mathbf{d}} Tr(\mathbf{d}^T\mathbf{X}^T\mathbf{X}\mathbf{d})$$

subject to $\mathbf{d}^T\mathbf{d} = 1$ can be solved using eigendecomposition. The optimal $\mathbf{d}$ is given by the eigenvector of $\mathbf{X}^T\mathbf{X}$ corresponding to the largest eigenvaue. $\square$