IFT 6760A - Lecture 3
Linear regression, inverse and pseudo inverse, eigenvalues and eigenvectors

Scribe(s): Sebastien Henwood, Amir Zakeri
(adapted from Tayssir Doghri, Bogdan Mazoure last year's notes) **Instructor:** Guillaume Rabusseau

1 Summary

In the previous lecture, we introduced one of the matrix decomposition methods called the Singular Value Decomposition(SVD). Then, we introduced some definitions related to orthogonality and projections such as orthonormal basis and orthogonal matrix.
In this lecture, we will continue to introduce some notions related to orthogonality and projections which are orthogonal complement and orthogonal projection. Then we will present another matrix decomposition called the QR decomposition along with an application in linear regression. In addition, we will present matrix inverse, eigenvalues and eigenvectors.

2 The QR decomposition

In order to solve some matrix problems, we use matrix decompositions (factorizations). In this section, we present the QR decomposition which can be used to solve the linear least squares problem.

Theorem 1. *Any matrix* $\mathbf{A} \in \mathbb{R}^{m\times n}$ *can be written as* $\mathbf{A} = \mathbf{QR}$ *where* $\mathbf{Q} \in \mathbb{R}^{m\times m}$ *is orthogonal and* $\mathbf{R} \in \mathbb{R}^{m\times n}$ *is upper triangular. This decomposition of* $\mathbf{A}$ *is called the QR decomposition.*

If $m > n$ *then the* reduced (thin) QR decomposition *of* $\mathbf{A}$ *is defined as:*

$$\mathbf{A} = \begin{bmatrix}\mathbf{Q}_1 & \mathbf{Q}_2\end{bmatrix}\begin{bmatrix}\mathbf{R}_1 \\ \\ 0\end{bmatrix} = \mathbf{Q}_1\mathbf{R}_1$$

where $\mathbf{Q}_1 \in \mathbb{R}^{m\times n}$ *is orthogonal and* $\mathbf{R}_1 \in \mathbb{R}^{n\times n}$ *is upper triangular.*

Remark 2. *If* $\mathbf{U} \in \mathbb{R}^{n\times k}$ *and* $\operatorname{rank}(\mathbf{U}) = k$ *then its thin QR decomposition* $\mathbf{U} = \mathbf{QR}$ *is such that:*

- $\mathcal{R}(\mathbf{Q}) = \mathcal{R}(\mathbf{U})$
- $\mathbf{R}$ *is invertible*

where $\mathbf{Q} \in \mathbb{R}^{n\times k}$ *and* $\mathbf{R} \in \mathbb{R}^{k\times k}$

If $\mathbf{u}_1, \cdots \mathbf{u}_k \in \mathbb{R}^n$ is a basis of $\mathcal{U}$, which is **not necessarily orthonormal**, and $\mathbf{U} \in \mathbb{R}^{n\times k}$ such that

$$\mathbf{U} = \begin{bmatrix} | & & | \\ \mathbf{u}_1 & \cdots & \mathbf{u}_k \\ | & & | \end{bmatrix}$$

then we have the following property: $\Pi_u(\mathbf{x}) = \mathbf{U}(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T\mathbf{x}$, with $\mathbf{U}(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T$ the matrix of the orthogonal projection onto $\mathcal{R}(u)$.

Proof. In order to show the previous property, let's consider the thin QR decomposition of $\mathbf{U}$, i.e, $\mathbf{U} = \mathbf{QR}$ where $\mathbf{Q} \in \mathbb{R}^{n\times k}$ is orthogonal and $\mathbf{R} \in \mathbb{R}^{k\times k}$ is upper triangular and invertible. We have

$$\begin{aligned}\mathbf{U}(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T\mathbf{x} &= \mathbf{Q}\underbrace{\mathbf{R}(\mathbf{R}^T\mathbf{Q}^T\mathbf{Q}\mathbf{R})^{-1}\mathbf{R}^T}_{\mathbf{I}}\mathbf{Q}^T\mathbf{x} \\ &= \mathbf{Q}\mathbf{Q}^T\mathbf{x} \\ &= \Pi_u(\mathbf{x})\end{aligned} \tag{1}$$

□

3 Linear regression

In the context of statistical learning theory, we are often interested in fitting the best model to a training set (i.e. perform regression).
Formally, we aim to learn a function $f : \mathbb{R}^d \to \mathbb{R}$ from a training set of examples which has the following form:

$$\mathcal{D} = \{(\mathbf{x}_1, y_1), .., (\mathbf{x}_N, y_N)\} \subseteq \mathbb{R}^d \times \mathbb{R} \tag{2}$$

where $y_i \approx f(\mathbf{x}_i)$ for each $i = 1, 2, .., N$.
Suppose the function f is linear, meaning $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$, for all $\mathbf{x} \in \mathbb{R}^d$, and some weight vector $\mathbf{w} \in \mathbb{R}^d$. One plausible approach to learning this function is by minimizing the Squared Error (SE) loss on an observed dataset $\mathcal{D}$:

$$\mathbf{w}^* = \arg\min_{w\in\mathbb{R}^d} \sum_{i=1}^{N} (\mathbf{w}^T\mathbf{x}_i - y_i)^2 \tag{3}$$

If we take

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1^T \\ \vdots \\ \mathbf{x}_N^T \end{pmatrix} \in \mathbb{R}^{N\times d}, \mathbf{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_N \end{pmatrix} \in \mathbb{R}^N, \tag{4}$$

then (3) can be written in matrix form as

$$\mathbf{w}^* = \arg\min_{\mathbf{w}\in\mathbb{R}^d} ||\mathbf{X}\mathbf{w} - \mathbf{y}||^2, \tag{5}$$

in which case $\mathbf{X}\mathbf{w} \in \mathcal{R}(\mathbf{X})$.

To solve for the Eq. 5, which is convex, we can take the gradient and solve it for 0 :

$$\begin{aligned}\nabla_{\mathbf{w}}||\mathbf{X}\mathbf{w} - \mathbf{y}||^2 = 2(\mathbf{X}^T\mathbf{X}\mathbf{w} - \mathbf{X}^T y) &= 0 \\ \Rightarrow \mathbf{X}^T\mathbf{X}\mathbf{w} &= \mathbf{X}^T y \\ \mathbf{w}^* &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}\end{aligned} \tag{6}$$

In fact, finding a solution to the linear regression problem can be seen as projecting the dataset onto the hyperplane spanned by $\mathbf{X}$. For instance, assuming the rank of $\mathbf{X}$ is d (i.e. full-rank), the predictions $\hat{\mathbf{y}}$ follows

$$\begin{aligned}\hat{\mathbf{y}} &= \arg\min_{\mathbf{v}\in\mathcal{R}(\mathbf{X})} ||\mathbf{v} - \mathbf{y}||^2 \\ &= \Pi_{\mathcal{R}(\mathbf{X})}(y) \\ &= \mathbf{X}\underbrace{(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}}_{\mathbf{w}^*},\end{aligned} \tag{7}$$

3.1 Reduced SVD

Given $\mathbf{A} \in \mathbb{R}^{m \times n}$, with the rank $\mathcal{R}(\mathbf{A}) = r$ and $m \geq n$, the Singular Value Decomposition of $\mathbf{A}$ can be written in the following reduced forms. The idea behind these reduced forms is that, as the size of the matrix decreases, so does the memory footprint. Therefore, the cost of storing the matrix in memory reduces substantially compared to full SVD. This will also increase the speed of computation to some extent.
In **Thin SVD**, only the n column vectors of $\mathbf{U}$ corresponding to the n row vectors of $\mathbf{V}^T$ are kept. Equivalently, we keep only the top square sub-matrix from the diagonal matrix $\mathbf{\Sigma}$. The remaining column vectors of $\mathbf{U}$ are not used and therefore are discarded. This has a positive effect on the memory footprint as there are less parameters to store, especially when $n \ll m$.
In **Compact SVD**, only the r column vectors of $\mathbf{U}$ and r row vectors of $\mathbf{V^T}$ corresponding to the r non-zero singular values $\mathbf{\Sigma_r}$ are kept. Equivalently, the top-left square sub-matrix with non-zero diagonal element is kept from the original $\mathbf{\Sigma}$ and the storage saving is increased. Up to this point, the *Thin* and *Compact* SVD were "free" in the sense that it didn't change the resulting matrix $\mathbf{A}$.
Finally, **Truncated SVD** is available to the user ready to forsake some accuracy in computations by discarding some information from $\mathbf{\Sigma}$. The Truncated SVD is an approximation to the full SVD, where the t column vectors of $\mathbf{U}$ and t row vectors of $\mathbf{V^T}$ corresponding to the t largest singular values $\mathbf{\Sigma}_t$ are kept. This again reduces the memory footprint in proportion of the singular values of $\mathbf{\Sigma}$ that are discarded. When a lot of these singular values are "close enough" to 0 to be deemed useless by the user chosen heuristic, we can eventually reach $t \ll r$.

$$\mathbf{A}_{m\times n} = \mathbf{U}_{m\times m}\mathbf{\Sigma}_{m\times n}{\mathbf{V}^T}_{n\times n} \qquad \textbf{(Full)} \qquad \mathbf{\Sigma}_{m\times n} = \begin{pmatrix} \mathbf{\Sigma}_{n\times n} \\ 0 \end{pmatrix},$$

$$= \mathbf{U}_{m\times n}\mathbf{\Sigma}_{n\times n}{\mathbf{V}^T}_{n\times n} \qquad \textbf{(Thin)} \qquad \mathbf{\Sigma}_{n\times n} = \begin{pmatrix} \sigma_1 & & & & & \\ & \ddots & & & & \\ & & \sigma_r & & & \\ & & & 0 & & \\ & & & & \ddots & \\ & & & & & 0 \end{pmatrix} = \begin{pmatrix} \mathbf{\Sigma_{r\times r}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}_{n\times n}$$

where, $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$.

$$= \mathbf{U}_{m\times r}\mathbf{\Sigma}_{r\times r}{\mathbf{V}^T}_{r\times n} \qquad \textbf{(Compact)}$$

$$\approx \mathbf{U}_{m\times t}\mathbf{\Sigma}_{t\times t}{\mathbf{V}^T}_{t\times n} \qquad \textbf{(Truncated)} \qquad \text{for } t \ll r$$

4 Matrix inverses and pseudo-inverses

Definition 3 (Matrix inversion)**.** *A matrix $\mathbf{A} \in \mathbb{R}^{m\times m}$ is invertible if $\exists \mathbf{A}^{-1} \in \mathbb{R}^{m\times m}$ such that $\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$.*

The matrix inverse has the following properties:

- $\mathbf{A}\mathbf{A}^{-1} = \mathbf{I} \Leftrightarrow \mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$
- $(\mathbf{A}\mathbf{B})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}$

The following statements regarding the matrix inverse are equivalent:

- $\det(\mathbf{A}) \neq 0$
- $\mathbf{A}^{-1}$ exists, i.e. $\mathbf{A}$ is invertible
- $\mathbf{A}$ has full rank
- $\mathcal{N}(\mathbf{A}) = \{\mathbf{0}\}$

Definition 4 (Moore-Penrose pseudo-inverse)**.** *Let $\mathbf{A} \in \mathbb{R}^{m\times n}$ and let $\mathbf{A} = \underbrace{\mathbf{U}}_{m\times R}\ \mathbf{D}\ \underbrace{\mathbf{V}^T}_{R\times m}$ be a compact SVD where $R = rank(\mathbf{A})$.*
Then, the Moore-Penrose pseudo-inverse of $\mathbf{A}$ is defined as $\mathbf{A}^\dagger = \mathbf{V}\mathbf{D}^{-1}\mathbf{U}^T \in \mathbb{R}^{m\times m}$.

We note that $\mathbf{D}$ is a diagonal matrix with strictly positive entries. The Moore-Penrose pseudo-inverse has the following properties:

- $\mathbf{A}\mathbf{A}^\dagger\mathbf{A} = \mathbf{A}$
- $\mathbf{A}^\dagger\mathbf{A}\mathbf{A}^\dagger = \mathbf{A}^\dagger$
- $(\mathbf{A}\mathbf{A}^\dagger)^T = \mathbf{A}\mathbf{A}^\dagger$
- $(\mathbf{A}^\dagger\mathbf{A})^T = \mathbf{A}^\dagger\mathbf{A}$

Suppose (as a special case) that $\text{rank}(\mathbf{A}) = m \leq n$ (i.e. $\mathbf{A}$ is full-rank). Then,

$$\begin{aligned}\mathbf{A}\mathbf{A}^\dagger &= (\mathbf{U}\mathbf{D}\underbrace{\mathbf{V}^T)(\mathbf{V}}_{identity}\mathbf{D}^{-1}\mathbf{U}^T) \\ &= \mathbf{U}\mathbf{U}^T \\ &= \underbrace{\mathbf{I}}_{m\times m},\end{aligned} \tag{8}$$

where the last equality holds since $\text{rank}(\mathbf{A}) = m$ hence $\mathbf{U}$ is square and $\mathbf{U}\mathbf{U}^T = \mathbf{U}^T\mathbf{U}$.
However, this simplification does not hold for $\mathbf{A}^\dagger\mathbf{A}$. For instance, if $m < n$, then

$$\begin{aligned}\mathbf{A}^\dagger\mathbf{A} &= (\mathbf{V}\mathbf{D}^{-1}\mathbf{U}^T)(\mathbf{U}\mathbf{D}\mathbf{V}^T) \\ &= \mathbf{V}\mathbf{V}^T \neq \mathbf{I} \\ &= \Pi_{\mathcal{R}(\mathbf{A})}\end{aligned} \tag{9}$$

For this last equation, we can also note that since $\mathcal{R}(\mathbf{V}) = \mathcal{R}(\mathbf{A})$ and $\mathbf{V}$ is orthogonal the following $\Pi_{\mathcal{R}(\mathbf{A})} = \mathbf{V}\mathbf{V}^T = \Pi_{\mathcal{R}(\mathbf{V})}$ is true.
In other words, we have shown that $\mathbf{A}^\dagger\mathbf{A} = \Pi_{\mathcal{R}(\mathbf{A})}$, i.e. the orthogonal projection onto the range; and $\mathbf{A}\mathbf{A}^\dagger = \Pi_{\mathcal{R}(\mathbf{A}^T)}$, i.e. the orthogonal projection onto the row space.

Example 5 (Under-determined linear system of equations)**.** *We consider the problem of solving under-determined system of equations, and will show how the smallest solution of such a system (in terms of the 2-norm) is related to the pseudo-inverse of the matrix of the system.*

We consider an under-determined system of equations:

$$\textit{Solve} \quad \underbrace{\mathbf{A}}_{m\times n}\ \mathbf{x} = \mathbf{y}, \quad \textit{for } \mathbf{x} \in \mathbb{R}^n, \quad \textit{where } m < n.$$

Since $m < n$, this systems has an infinite number of solutions. Assuming that $\mathbf{A}$ is full-rank, we want to show that $\mathbf{A}^\dagger\mathbf{y}$ is the least norm solution. For this, we make the following claims:

Claim 1 *$\mathbf{x}_{LN} = \mathbf{A}^\dagger\mathbf{y}$ is a solution, i.e. $\mathbf{A}\mathbf{x}_{LN} = \mathbf{y}$*

Proof. We can easily show that $\mathbf{A}\mathbf{x}_{LN} = \underbrace{\mathbf{A}\mathbf{A}^\dagger}_{\mathbf{I}}\mathbf{y} = \mathbf{y}$, because $\mathbf{A}$ is full rank. But there are infinite number of solutions to this equation! $\mathbf{x}_{LN}$ is just one of them. □

Now we make the following interesting claim:

Claim 2 *$\mathbf{x}_{LN}$ is the solution with the smallest 2-norm.*

Proof. To prove that, let $\mathbf{x}$ be another solution, i.e., $\mathbf{A}\mathbf{x} = \mathbf{y}$. We want to show that this second solution has larger norm than the first one, i.e., $||\mathbf{x}|| \geq ||\mathbf{x_{LN}}||$.

The idea of the proof is to first show that $\mathbf{x}$ and $(\mathbf{x} - \mathbf{x}_{LN})$ are orthogonal, and then use the Pythagorean theorem (on the triangle with vertices $\mathbf{0}, \mathbf{x}$ and $\mathbf{x_{LN}}$) to show that $||\mathbf{x}|| \geq ||\mathbf{x_{LN}}||$.

Let $\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$ be the compact SVD of $\mathbf{A}$. The inner product of $\mathbf{x}_{LN}$ and the difference between $\mathbf{x}_{LN}$ and $\mathbf{x}$ is

$$\begin{aligned}\langle \mathbf{x}_{LN}, \mathbf{x} - \mathbf{x}_{LN}\rangle &= \mathbf{x}_{LN}^T(\mathbf{x} - \mathbf{x}_{LN}) \\ &= \mathbf{y}^T\mathbf{A}^{\dagger T}(\mathbf{x} - \mathbf{x}_{LN}) \\ &= \mathbf{y}^T\mathbf{U}\mathbf{D}^{-1}\mathbf{V}^T(\mathbf{x} - \mathbf{x}_{LN})\end{aligned} \tag{10}$$

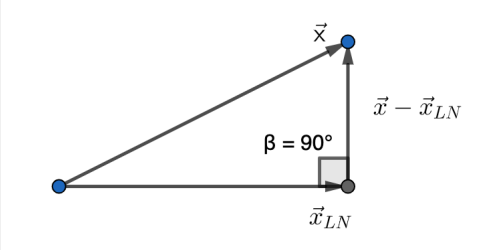
Now observe that

$$\mathbf{A}(\mathbf{x} - \mathbf{x}_{LN}) = \mathbf{y} - \mathbf{y} = 0. \tag{11}$$

Using the fact that $\mathbf{U}$ and $\mathbf{D}$ are both invertible (since $m < n$) this implies that $\mathbf{V}^\top(\mathbf{x} - \mathbf{x}_{LN}) = \mathbf{0}$ (multiply Eq. (11) to the left by $\mathbf{D}^{-1}\mathbf{U}^\top$).

It follows that $\langle \mathbf{x}_{LN}, \mathbf{x} - \mathbf{x}_{LN}\rangle = 0$ and we can use the Pythagorean theorem to obtain

$$\begin{aligned}||\mathbf{x}||^2 &= ||\mathbf{x}_{LN}||^2 + ||\mathbf{x} - \mathbf{x}_{LN}||^2 \\ &\geq ||\mathbf{x}_{LN}||^2.\end{aligned} \tag{12}$$



□

5 Eigenvalues

Definition 6 (Eigenvalue, eigenvector and eigenspace)**.** *Let* $\mathbf{A} \in \mathbb{R}^{m\times m}$*. Any* $\mathbf{v} \in \mathbb{R}^m$ *such that* $\mathbf{v} \neq 0$ *and satisfying*

$$\mathbf{A}\mathbf{v} = \lambda\mathbf{v}$$

for $\lambda \in \mathbb{C}$ *is called an eigenvector of* $\mathbf{A}$ *corresponding to the eigenvalue* λ*. The space* $E_\lambda = \{\mathbf{v} \in \mathbb{R}^m | \mathbf{A}\mathbf{v} = \lambda\mathbf{v}\}$ *is called the eigenspace of* $\mathbf{A}$ *corresponding to* λ*.*

For example, if $\mathbf{A} = \mathbf{I}$ then $\mathbf{A}\mathbf{v} = \mathbf{I}\mathbf{v} = \mathbf{v}$ for all $\mathbf{v}$ and 1 is an eigenvalue with corresponding eigenspace $E_1 = \mathbb{R}^m$.
Eigenvalues can be found by finding the roots of the characteristic polynomial:

$$\begin{aligned} \mathbf{A}\mathbf{v} = \lambda\mathbf{v} &\iff (\mathbf{A} - \lambda\mathbf{I})\mathbf{v} = 0 \\ &\iff \mathbf{v} \in \mathcal{N}(\mathbf{A} - \lambda\mathbf{I}) \\ &\iff \det(\mathbf{A} - \lambda\mathbf{I}) = 0 \end{aligned} \tag{13}$$

As an example, let's find the eigenvalues for a given matrix $\mathbf{A}$.

$$\mathbf{A} = \begin{pmatrix} 1 & 1 \\ 0 & 2 \end{pmatrix}$$

Its characteristic polynomial is

$$\det\left(\begin{pmatrix} 1-\lambda & 1 \\ 0 & 2-\lambda \end{pmatrix}\right) = (1-\lambda)(2-\lambda)$$

which implies that the eigenvalues are $\lambda \in \{1, 2\}$.
As a special case, if $\mathbf{A}$ is triangular, then its determinant is the product of its eigenvalues and the eigenvalues are the diagonal entries of $\mathbf{A}$.

Definition 7 (Diagonalization)**.** *A matrix* $\mathbf{A} \in \mathbb{R}^{n\times m}$ *is diagonalizable iff there exists a basis* $\mathbf{v}_1, .., \mathbf{v}_n$ *of* $\mathbb{R}^n$ *consisting of eigenvectors of* $\mathbf{A}$*.*
In this case, $\mathbf{V}^{-1}\mathbf{A}\mathbf{V} = \mathbf{D}$ *is diagonal.*

An example of a matrix diagonalizable over $\mathbb{C}$ but not over $\mathbb{R}$ is

$$\mathbf{A} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$$

For instance, consider the problem of finding the eigenvectors of $\mathbf{A}$. Simplifying the characteristic polynomial equation implies that we have to solve $\lambda^2 + 1 = 0$. The equation has no real roots but has two complex roots, $\lambda \in \{i, -i\}$, which allows for $\mathbf{A}$ to be diagonalizable over $\mathbb{C}$.
An example of a non-diagonalizable matrix is

$$\mathbf{A} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$$