Basic Methods of Data Analysis

Philipp Renz

2021/11/26 12:31 Noon

Chapter 1

Linear algebra

In data analysis we will often represent our data as matrices. Consequently many algorithms are described using the concept of matrices and their properties which could be broadly called the study of linear algebra. Here I don't want to give an axiomatic introduction but rather provide some basic results with the prerequisite that the reader already knows some basics about vectors and matrices.

We will denote vectors as bold letters $\boldsymbol{a}$,

$$\boldsymbol{a} = \begin{pmatrix} a_1 \\ \vdots \\ a_m \end{pmatrix} \tag{1.1}$$

Often we will have a set of vectors $\{\boldsymbol{a}_1, \dots, \boldsymbol{a}_n\}$. We denote the elements of these vectors with two indices:

$$\boldsymbol{a}_j = \begin{pmatrix} a_{j1} \\ \vdots \\ a_{jm} \end{pmatrix} \tag{1.2}$$

We can also write this as a matrix containing set of vectors as rows

Or would this be better as columns as vectors are column vectors

$$\boldsymbol{A} = \begin{pmatrix} \text{---} & \boldsymbol{a_1} & \text{---} \\ & \vdots & \\ \text{---} & \boldsymbol{a_n} & \text{---} \end{pmatrix} = \begin{pmatrix} a_{11} & \dots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{nm} \end{pmatrix} \tag{1.3}$$

1.1 Matrix multiplication

We are given two matrices,

$$\boldsymbol{A} = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \dots & a_{mn} \end{pmatrix}, \tag{1.4}$$

and

$$\boldsymbol{B} = \begin{pmatrix} b_{11} & \dots & b_{1m} \\ \vdots & \ddots & \vdots \\ b_{n1} & \dots & b_{nm} \end{pmatrix}, \tag{1.5}$$

where $\boldsymbol{A} \in R^{m \times n}$ and $\boldsymbol{B} \in R^{n \times p}$. The entries of their product $\boldsymbol{C} = \boldsymbol{AB}$ is defined by

$$c_{ik} = \sum_{i=1}^{n} a_{ij} b_{jk}. \tag{1.6}$$

The resulting matrix $\boldsymbol{C}$ has a shape of $(m \times p) \leftarrow (m \times n)(n \times p)$ as the inner dimension n is eliminated by summing over it. One can also see that the sum moves horizontally/rowwise over the entries of $\boldsymbol{A}$ and down the columns of $\boldsymbol{B}$

Often we are interested in a special case in which one of the matrices reduces to a vector, that means it has only one column or one row. We define a vector as a shorthand notation for a matrix with only one column, where we drop the column index:

$$\boldsymbol{B} = \begin{pmatrix} b_{11} \\ \vdots \\ b_{n1} \end{pmatrix} \widehat{=} \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix} = \boldsymbol{b}. \tag{1.7}$$

Implicitly when we talk of a vector $\boldsymbol{b}$ with n elements we mean a matrix $\boldsymbol{B} \in R^{n \times 1}$. When we want to talk about a vector with only one row we write this a transposed vector

$$\boldsymbol{b}^T = \begin{pmatrix} b_1 & \dots & b_n \end{pmatrix} = \begin{pmatrix} b_{11} & \dots & b_{1n} \end{pmatrix} \tag{1.8}$$

A matrix vector product $\boldsymbol{Ab}$ again results in a vector $(m \times 1) \leftarrow (m \times n)(n \times 1)$.

The inner or scalar product between to vectors is defined as

$$\boldsymbol{a} \cdot \boldsymbol{b} = \boldsymbol{a}^T \boldsymbol{b} = \sum_{i=1}^{n} a_i b_i. \tag{1.9}$$

The outer product between vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ is defined as

$$\boldsymbol{C} = \boldsymbol{a}\boldsymbol{b}^{\boldsymbol{T}} = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{pmatrix} \begin{pmatrix} b_1 & b_2 & \dots & b_n \end{pmatrix} = \begin{pmatrix} a_1 b_1 & a_1 b_2 & \dots & a_1 b_n \\ a_2 b_1 & a_2 b_2 & \dots & a_2 b_n \\ \vdots & \vdots & \ddots & \vdots \\ a_m b_1 & a_m b_2 & \dots & a_m b_n \end{pmatrix}. \tag{1.10}$$

This means that every column is a multiple of $\boldsymbol{a}$ and every row a multiple of $\boldsymbol{b}$. We see that the entry $c_{ij} = a_i b_j$.

A matrix vector product can be view as computing a linear combination of the column vectors of $\boldsymbol{A}$:

$$\boldsymbol{Ab} = \begin{pmatrix} \sum_{j=1}^{n} a_{1j} b_j \\ \sum_{j=1}^{n} a_{2j} b_j \\ \vdots \\ \sum_{j=1}^{n} a_{mj} b_j \end{pmatrix} = \begin{pmatrix} a_{11} b_1 + a_{12} b_2 + \ldots + a_{1n} b_n \\ a_{21} b_1 + a_{22} b_2 + \ldots + a_{2n} b_n \\ \vdots \quad\quad \vdots \quad\quad \ddots \quad\quad \vdots \\ a_{m1} b_1 + a_{m2} b_2 + \ldots + a_{mn} b_n \end{pmatrix} = \begin{pmatrix} & | & & | & & & | \\ b_1 & \boldsymbol{a}_1 + & b_2 & \boldsymbol{a}_2 + & \ldots + & b_n & \boldsymbol{a}_n \\ & | & & | & & & | \end{pmatrix} = b_1 \boldsymbol{a}_1 + \ldots b_n \boldsymbol{a}_n = \sum_{i=1}^{n} b_i \boldsymbol{a_i} \tag{1.11}$$

Since $c_i = \sum_{j=1}^{n} a_{ij} b_j$ we can interpret each entry in the result as a dot product between i^{th} row-vector of $\boldsymbol{A}$ and $\boldsymbol{b}$ as the for each entry of $\boldsymbol{c}$ the row-index stays constant while we iterate over the columns:

$$\boldsymbol{c} = \begin{pmatrix} c_1 \\ \vdots \\ c_m \end{pmatrix} = \begin{pmatrix} — & \boldsymbol{a}_1 & — \\ & \vdots & \\ — & \boldsymbol{a}_m & — \end{pmatrix} \begin{pmatrix} | \\ \boldsymbol{b} \\ | \end{pmatrix} = \begin{pmatrix} \boldsymbol{a}_1 \cdot \boldsymbol{b} \\ \vdots \\ \boldsymbol{a}_m \cdot \boldsymbol{b} \end{pmatrix} \tag{1.12}$$

Thus we can always interpret a matrix-vector multiplication as either computing a linear combination of columns of $\boldsymbol{A}$ where the weights for each column are given by the entries of $\boldsymbol{b}$

Of course the same patterns can be found when considering multiplying a transposed vector times a matrix. In the derivations above one can just look at $\boldsymbol{Ab}$ and transpose it to yield $\boldsymbol{c} = \boldsymbol{b}^T \boldsymbol{A}^T$. Thus all the things apply when we just exchange the notion of columns and vectors. For notational simplicity we choose a new matrix $A \leftarrow A^T$ so we can drop the transposed sign.

make same color stuff as above

$$\boldsymbol{c}^T = \boldsymbol{b}^T\boldsymbol{A} = \begin{pmatrix}\sum_{j=1}^n b_j a_{j1} & \sum_{j=1}^n b_j a_{j2} & \dots & \sum_{j=1}^n b_j a_{jm}\end{pmatrix} = \tag{1.13}$$

$$= \begin{pmatrix} b_1 a_{11} & b_1 a_{12} & \dots & b_1 a_{1m} \\ + & + & & + \\ b_2 a_{21} & b_2 a_{22} & \dots & b_2 a_{2m} \\ + & + & & + \\ \vdots & \vdots & \ddots & \vdots \\ b_n a_{n1} & b_n a_{n2} & \dots & b_n a_{nm} \end{pmatrix} = \begin{pmatrix} — & b_1 \boldsymbol{a}_1 & — \\ & + & \\ — & b_2 \boldsymbol{a}_2 & — \\ & + & \\ & \vdots & \\ — & b_n \boldsymbol{a}_n & — \end{pmatrix} = b_1 \boldsymbol{a}_1^T + \dots + b_n \boldsymbol{a}_n^T = \sum_{j=1}^n b_j \boldsymbol{a}_j^T \tag{1.14}$$

.

And again the dot product interpretation

$$\boldsymbol{c}^T = \boldsymbol{b}^T\boldsymbol{A} = \begin{pmatrix}\sum_{j=1}^n b_j a_{j1} & \sum_{j=1}^n b_j a_{j2} & \dots & \sum_{j=1}^n b_j a_{jm} =\end{pmatrix} \tag{1.15}$$

$$= \begin{pmatrix}\boldsymbol{b^T a_1} & \boldsymbol{b}^T\boldsymbol{a}_2 & \dots & \boldsymbol{b^T a_m}\end{pmatrix} = \tag{1.16}$$

$$= \begin{pmatrix} — & \boldsymbol{b} & — \end{pmatrix} \begin{pmatrix} | & \dots & | \\ \boldsymbol{a}_1 & \dots & \boldsymbol{a}_m \\ | & \dots & | \end{pmatrix} \tag{1.17}$$

The last line shows us again how matrix vector products always operate in a "crossed" way.

1.2 Matrix matrix

1.2.1 multiple matrix vector operations

$$\boldsymbol{AB} = \boldsymbol{A} \begin{pmatrix} | & \dots & | \\ \boldsymbol{b}_1 & \dots & \boldsymbol{b}_m \\ | & \dots & | \end{pmatrix} = \begin{pmatrix} | & \dots & | \\ \boldsymbol{Ab}_1 & \dots & \boldsymbol{Ab}_m \\ | & \dots & | \end{pmatrix} \tag{1.18}$$

This of course means that we can interpret this as multiple linear combination of the columns of $\boldsymbol{A}$ which are stored in the columns of $\boldsymbol{C}$. Each column is weighted by the respective column of $\boldsymbol{B}$.

In the dot product view we can write this as.

$$\boldsymbol{C} = \begin{pmatrix} — & \boldsymbol{a}_1 & — \\ & \vdots & \\ — & \boldsymbol{a}_m & — \end{pmatrix} \begin{pmatrix} | & & | \\ \boldsymbol{b}_1 & \dots & \boldsymbol{b}_p \\ | & & | \end{pmatrix} = \begin{pmatrix} \boldsymbol{a}_1 \cdot \boldsymbol{b}_1 & \boldsymbol{a}_1 \cdot \boldsymbol{b}_2 & \dots & \boldsymbol{a}_1 \cdot \boldsymbol{b}_p \\ \boldsymbol{a}_2 \cdot \boldsymbol{b}_1 & \boldsymbol{a}_2 \cdot \boldsymbol{b}_2 & \dots & \boldsymbol{a}_2 \cdot \boldsymbol{b}_p \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{a}_m \cdot \boldsymbol{b}_1 & \boldsymbol{a}_m \cdot \boldsymbol{b}_2 & \dots & \boldsymbol{a}_m \cdot \boldsymbol{b}_p \end{pmatrix} \tag{1.19}$$

This means that $c_{ij} = \boldsymbol{a}_i \cdot \boldsymbol{b_j}$, as can also be seen from the definition of matrix multiplication.

$$c_{ik} = \sum_j a_{ij} b_{jk} \tag{1.20}$$

If we concentrate on the column vectors of $\boldsymbol{A}$ and rows vectors of $\boldsymbol{B}$ instead we can view a matrix matrix multiplication as a sum of outer products.

$$c_{ik} = \sum_j a_{ij} b_{jk} = a_{i1} b_{1k} + a_{i2} b_{2k} + \dots \tag{1.21}$$

We can interpret the summands in this equation as outer products. E.g. $a_{i1} b_{1k}$ is the ik^{th} entry of the outer product of the first column vector of $\boldsymbol{A}$, $\boldsymbol{a_1}$ and the first row vector of $\boldsymbol{B}$, $\boldsymbol{b_1}$.

$$\boldsymbol{C} = \boldsymbol{AB} = \begin{pmatrix} | & & | \\ \boldsymbol{a}_1 & \dots & \boldsymbol{a}_n \\ | & & | \end{pmatrix} \begin{pmatrix} — & \boldsymbol{b}_1 & — \\ & \vdots & \\ — & \boldsymbol{b}_n & — \end{pmatrix} = \sum_{i=1}^n \boldsymbol{a}_i \boldsymbol{b}_i^T \tag{1.22}$$

The trace of a square matrix $\boldsymbol{A}$ is defined as $\sum_i a_{ii}$. The trace of the product $\boldsymbol{AB}$ is

$$\operatorname{Tr}(\boldsymbol{AB}) = \sum_i \sum_j a_{ij} b_{ji} = \sum_j \sum_i b_{ji} a_{ij} = \operatorname{Tr}(\boldsymbol{BA}). \tag{1.23}$$

In both of the middle terms the sum over the inner indices can be interpreted as belonging to the matrix multiplication and the outer to the trace. Since the summation and multiplication are commutative we can exchange the order.

Inverse of a matrix The inverse of a square matrix $\boldsymbol{A}$ is denoted as $\boldsymbol{A}^{-1}$ fulfills the following equation $\boldsymbol{AA}^{-1} = \mathbf{1}$. An inverse only exists if the matrix is non-singular that means that all its columns and rows are linearly independent.

Orthogonal matrices Orthogonal matrices have the special properties that their transpose is their inverse.

$$\boldsymbol{UU}^T = \boldsymbol{U}^T\boldsymbol{U} = \mathbf{1} \tag{1.24}$$

Looking at this matrix multiplication in the dot product interpretation this yields

$$\mathbf{1} = \begin{pmatrix} — & \boldsymbol{u}_1 & — \\ & \vdots & \\ — & \boldsymbol{u}_m & — \end{pmatrix} \begin{pmatrix} | & & | \\ \boldsymbol{u}_1 & \dots & \boldsymbol{u}_m \\ | & & | \end{pmatrix} = \begin{pmatrix} \boldsymbol{u}_1 \cdot \boldsymbol{u}_1 & \boldsymbol{u}_1 \cdot \boldsymbol{u}_2 & \dots & \boldsymbol{u}_1 \cdot \boldsymbol{u}_m \\ \boldsymbol{u}_2 \cdot \boldsymbol{u}_1 & \boldsymbol{u}_2 \cdot \boldsymbol{u}_2 & \dots & \boldsymbol{u}_2 \cdot \boldsymbol{u}_m \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{u}_m \cdot \boldsymbol{u}_1 & \boldsymbol{u}_m \cdot \boldsymbol{u}_2 & \dots & \boldsymbol{u}_m \cdot \boldsymbol{u}_m \end{pmatrix} = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{pmatrix} \tag{1.25}$$

Looking at the above shows that both column and row vectors of $\boldsymbol{U}$ have length 1 and are pairwise orthogonal, that means $\boldsymbol{u}_i \cdot \boldsymbol{u}_j = \delta_{ij}$.

1.3 Eigenvalues and eigenvectors

A square matrix $\boldsymbol{A} \in R^{n\times n}$ has at least one and up to n eigenvectors/eigenvalues, that is a vector which is mapped to a multiple of itself,

$$\boldsymbol{Av} = \lambda \boldsymbol{v}. \tag{1.26}$$

The multiplicative factor λ is called eigenvalue. The eigenvalues are a useful characteristic of a matrix and finding eigenvalues has lots of applications ranging from coupled oscillators, heat and Schrödinger equations in physics to principal component analysis in data analysis.

Given the equation above we still need a way to find the eigenvalues/vectors. To do this we change the equation above

$$\boldsymbol{Av} = \lambda \mathbf{1} \boldsymbol{v} \tag{1.27}$$

$$(\boldsymbol{A} - \lambda\mathbf{1})\boldsymbol{v} = 0 \tag{1.28}$$

This yields a linear system of equations, which only has a solution if the columns of $\boldsymbol{A} - \lambda\mathbf{1}$ are linearly dependent. Thus we can find eigenvalues by setting the determinant to zero

$$\det(\boldsymbol{A} - \lambda\mathbf{1}) = 0 \tag{1.29}$$

This gives a polynomial of degree n, called the characteristic polynomial, which according to the fundamental theorem of calculus has n roots and can be written as

$$\det(\boldsymbol{A} - \lambda\mathbf{1}) = \prod_{i=1}^{n} (\lambda - \lambda_i) \tag{1.30}$$

where the λ_i are solutions to the equation above. The number of times a root occurs in the solution is called its algebraic multiplicity.

After calculating the eigenvalues one can insert them into (1.28) one ofter another and solve for the eigenvectors $\boldsymbol{v}_i$. One interesting question is whether the set of all eigenvectors form a complete basis of R^n. This is not the case if for an eigenvalue with algebraic multiplicity $n_a > 1$ the null space of $\boldsymbol{A} - \lambda_i\mathbf{1}$ has dimension less than n_a.

Example Given a matrix

$$\boldsymbol{A} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \tag{1.31}$$

We can calculate it's characteristic polynomial

$$\det(A - \lambda \boldsymbol{I}) = \left| \begin{pmatrix} 1-\lambda & 1 \\ 0 & 1-\lambda \end{pmatrix} \right| = (1-\lambda)^2. \tag{1.32}$$

Thus we get an eigenvalue $\lambda = 1$ with algebraic multiplicity of two. However when inserting this eigenvalue to solve for $\boldsymbol{v}$ we get

$$(\boldsymbol{A} - 1\boldsymbol{I})\boldsymbol{v} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \quad \Rightarrow \boldsymbol{v} = s \begin{pmatrix} 1 \\ 0 \end{pmatrix} \tag{1.33}$$

Thus the set of all eigenvectors only spans a space of dimension one.

Example: Full basis of eigenvectors Find example.

1.3.1 Diagonalization of a matrix

If we find an eigenbasis, that means a basis that only consists of eigenvectors, we can write the following:

$$\boldsymbol{A} \begin{pmatrix} | & & | \\ \boldsymbol{v}_1 & \dots & \boldsymbol{v}_n \\ | & & | \end{pmatrix} = \begin{pmatrix} | & & | \\ \lambda_1 \boldsymbol{v}_1 & \dots & \lambda_n \boldsymbol{v}_n \\ | & & | \end{pmatrix} = \begin{pmatrix} | & & | \\ \boldsymbol{v}_1 & \dots & \boldsymbol{v}_n \\ | & & | \end{pmatrix} \begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{pmatrix} \tag{1.34}$$

or

$$\boldsymbol{A}\boldsymbol{V} = \boldsymbol{V}\boldsymbol{\Lambda}. \tag{1.35}$$

Since the columns of bmV form a basis they are linear independent and one can invert $\boldsymbol{V}$. Thus if there is an eigenbasis a matrix can be "diagonalized":

$$\boldsymbol{A} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{-1}. \tag{1.36}$$

This form is often practical, for example it's easy to compute

$$\boldsymbol{A}^p = \underbrace{\boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{-1}\boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{-1}\dots\boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{-1}}_{p\times} = \tag{1.37}$$

$$= \boldsymbol{V}\boldsymbol{\Lambda}^{\boldsymbol{p}}\boldsymbol{V}^{-1} = \boldsymbol{V} \begin{pmatrix} \lambda_1^p & & \\ & \ddots & \\ & & \lambda_n^p \end{pmatrix} \boldsymbol{V}^{-1} \tag{1.38}$$

https://people.math.carleton.ca/~kcheung/math/notes/MATH1107/wk10/10_orthogonal_diagonalization_proof.html

1.4 Singular value decomposition

Update this with new stuff

We will now derive a useful way to write any rectangular matrix $\boldsymbol{M}$ of shape $(m \times n)$. We know already that one can diagonalize $\boldsymbol{M}^T\boldsymbol{M}$.

$$\boldsymbol{V}^T\boldsymbol{M}^T\boldsymbol{M}\boldsymbol{V} = \bar{\boldsymbol{D}} \tag{1.39}$$

Without loss of generality we assume that $m \geq n$.

$$\begin{aligned} \boldsymbol{M} &= (m \times n) && \qquad (1.40) \\ \boldsymbol{M}^T\boldsymbol{M} &= (n \times n) & &= (m \times n)(n \times m) \qquad (1.41) \\ \bar{\boldsymbol{D}} &= (n \times n) && \qquad (1.42) \end{aligned}$$

$\bar{\boldsymbol{D}}$ is a diagonal matrix with $l \leq n$, thus we can write it as

$$\bar{\boldsymbol{D}} = \begin{pmatrix} \boldsymbol{D}_{l\times l} & \mathbf{0}_{l\times(n-l)} \\ \mathbf{0}_{(n-l)\times l} & \mathbf{0}_{(n-l)\times(n-l)} \end{pmatrix} \tag{1.43}$$

We have $\boldsymbol{V}$ the matrix of eigenvectors of $\boldsymbol{M}^T\boldsymbol{M}$.

$$\boldsymbol{V} = \begin{pmatrix} \vert & & \vert \\ \boldsymbol{v}_1 & \dots & \boldsymbol{v}_n \\ \vert & & \vert \end{pmatrix}, \tag{1.44}$$

and the matrix of eigenvectors with non-zero eigenvalues.

$$\boldsymbol{V}_1 = \begin{pmatrix} \vert & & \vert \\ \boldsymbol{v}_1 & \dots & \boldsymbol{v}_l \\ \vert & & \vert \end{pmatrix} \qquad \boldsymbol{V}_2 = \begin{pmatrix} \vert & & \vert \\ \boldsymbol{v}_{l+1} & \dots & \boldsymbol{v}_n \\ \vert & & \vert \end{pmatrix}. \tag{1.45}$$

Since these vectors are orthonormal we know that:

$$\boldsymbol{V}_1^T\boldsymbol{V}_1 = \begin{pmatrix} — & \boldsymbol{v}_1 & — \\ & \vdots & \\ — & \boldsymbol{v}_l & — \end{pmatrix} \begin{pmatrix} \vert & & \vert \\ \boldsymbol{v}_1 & \dots & \boldsymbol{v}_l \\ \vert & & \vert \end{pmatrix} = \mathbf{1}_{l\times l} \tag{1.46}$$

$$\boldsymbol{V}_2^T\boldsymbol{V}_2 = \begin{pmatrix} — & \boldsymbol{v}_{l+1} & — \\ & \vdots & \\ — & \boldsymbol{v}_n & — \end{pmatrix} \begin{pmatrix} \vert & & \vert \\ \boldsymbol{v}_{l+1} & \dots & \boldsymbol{v}_n \\ \vert & & \vert \end{pmatrix} = \mathbf{1}_{(n-l)\times(n-l)} \tag{1.47}$$

$$\boldsymbol{V}_1\boldsymbol{V}_1^T + \boldsymbol{V}_2\boldsymbol{V}_2^T = \sum_{i=1}^{l} \boldsymbol{v}_i\boldsymbol{v}_i^T + \sum_{i=l+1}^{n} \boldsymbol{v}_i\boldsymbol{v}_i^T = \boldsymbol{V}\boldsymbol{V}^T = \mathbf{1}_{n\times n} \tag{1.48}$$

$$\boldsymbol{V}_2^T(\boldsymbol{M}^T\boldsymbol{M})\boldsymbol{V}_2 = \boldsymbol{V}_2^T(\mathbf{0}\boldsymbol{V}_2) = \mathbf{0} \Rightarrow \boldsymbol{M}\boldsymbol{V}_2 = \mathbf{0} \tag{1.49}$$

Define $\boldsymbol{U}_1$ as

$$\boldsymbol{U}_1 = \boldsymbol{M}\boldsymbol{V}\boldsymbol{D}^{-\frac{1}{2}} \tag{1.50}$$

$$\begin{aligned} \boldsymbol{U}_{\mathbf{1}}\boldsymbol{D}^{\frac{1}{2}}\boldsymbol{V}_1^T = \boldsymbol{M}\boldsymbol{V}_1\boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{D}^{\frac{1}{2}}\boldsymbol{V}_1^T = \boldsymbol{M}\boldsymbol{V}_1\boldsymbol{V}_1^T = \boldsymbol{M}(\mathbf{1}_{n\times n} - \boldsymbol{V}_2\boldsymbol{V}_2^T) = \quad (1.51) \\ = \boldsymbol{M} - \boldsymbol{M}\boldsymbol{V}_2\boldsymbol{V}_2^T = \boldsymbol{M} \quad (1.52) \end{aligned}$$

Thus we now have a way of factorizing our matrix $\boldsymbol{M} = \boldsymbol{U}_1\boldsymbol{D}^{\frac{1}{2}}\boldsymbol{V}_1^T$. This gives us a first interesting result:

$$\begin{aligned} \boldsymbol{M}^T\boldsymbol{M} &= \boldsymbol{V}_1^T\boldsymbol{D}^{\frac{1}{2}}\boldsymbol{U}_1^T\boldsymbol{U}_1\boldsymbol{D}^{\frac{1}{2}}\boldsymbol{V}_1^T = \boldsymbol{V}_1^T\boldsymbol{D}\boldsymbol{V}_1^T \quad (1.53) \\ \boldsymbol{M}\boldsymbol{M}^T &= \boldsymbol{U}_1^T\boldsymbol{D}^{\frac{1}{2}}\boldsymbol{V}_1^T\boldsymbol{V}_1\boldsymbol{D}^{\frac{1}{2}}\boldsymbol{U}_1^T = \boldsymbol{U}_1^T\boldsymbol{D}\boldsymbol{U}_1^T \quad (1.54) \end{aligned}$$

From this we get that $\boldsymbol{M}^T\boldsymbol{M}$ and $\boldsymbol{M}\boldsymbol{M}^T$ have the same non-zero eigenvalues.

In general we might like to have a similar formula but with complete sets of orthonormal vectors instead of only having l vectors in both $\boldsymbol{U}_1$ with shape $(m \times l)$ and $\boldsymbol{V}_1$ with shape $(n \times l)$.

We will first extend $\boldsymbol{V}_2$ to a complete set by adding zero column vectors to $\boldsymbol{D}$ and $\{\boldsymbol{v}_{l+1}, \dots, \boldsymbol{v}_n\}$ to $\boldsymbol{V}_1$

$$\boldsymbol{D}\boldsymbol{V}_1^T = \begin{pmatrix} \vert & & \vert & \vert & & \vert \\ \boldsymbol{d}_1 & \dots & \boldsymbol{d}_l & \mathbf{0}_{l+1} & \dots & \mathbf{0}_n \\ \vert & & \vert & \vert & & \vert \end{pmatrix} \begin{pmatrix} — & \boldsymbol{v}_1 & — \\ & \vdots & \\ — & \boldsymbol{v}_l & — \\ — & \boldsymbol{v}_{l+1} & — \\ & \vdots & \\ — & \boldsymbol{v}_n & — \end{pmatrix} = \boldsymbol{E}\boldsymbol{V} \tag{1.55}$$

Using the outer product view of matrix multiplication introduced above we see that the added columns and rows do not affect the result.

Chapter 2

Multiple Linear regression

We now want to extend the concept of linear regression to multiple explanatory variables. For example we want to estimate the price of house number i given some of its properties

x_{i1}	$\dots$	area
x_{i2}	$\dots$	garden size
x_{i3}	$\dots$	age
x_{i4}	$\dots$	distance to next pizzeria
$\vdots$		$\vdots$
x_{im}	$\dots$	dummy feature

We want to predict the house price using a linear function

$$\hat{y}_i = w_1 x_{i1} + w_2 x_{i2} + \dots + \underbrace{w_m x_{im}}_{w_m \cdot 1 \mathrel{\widehat{=}} b} = \sum_{j=1}^{m} x_{ij} w_j = \boldsymbol{w}^T \boldsymbol{x}_i \tag{2.1}$$

We can combine this equation for all datapoints using matrix notation:

$$\begin{pmatrix} — & \boldsymbol{x}_1 & — \\ & \vdots & \\ — & \boldsymbol{x}_m & — \end{pmatrix} \begin{pmatrix} | \\ \boldsymbol{w} \\ | \end{pmatrix} = \begin{pmatrix} \boldsymbol{x}_1^T \boldsymbol{w} \\ \vdots \\ \boldsymbol{x}_n^T \boldsymbol{w} \end{pmatrix} = \begin{pmatrix} \hat{y}_1 \\ \vdots \\ \hat{y}_m \end{pmatrix} = \boldsymbol{\hat{y}} \tag{2.2}$$

Our goal is to minimize the mean squared error between the true values $\boldsymbol{y}$ and the predictions $\boldsymbol{\hat{y}}$ by finding a good weight vector $\boldsymbol{w}$ We can write this down as a loss function

$$\begin{aligned} \mathcal{L}(\boldsymbol{w}) =& \frac{1}{n}(\boldsymbol{\hat{y}} - \boldsymbol{y})^2 = \frac{1}{n}(\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y})^2 = & (2.3) \\ =& \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{x}_i \boldsymbol{w} - y_i)^2 = & (2.4) \\ =& \frac{1}{n}\sum_{i=1}^{n}\left(\sum_{j=1}^{m} x_{ij} w_j - y_i\right)^2 & (2.5) \end{aligned}$$

or

$$\begin{aligned}
\mathcal{L}(\boldsymbol{w}) =& \frac{1}{n}(\hat{\boldsymbol{y}} - \boldsymbol{y})^2 = \frac{1}{n}(\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y})^2 = && (2.6)\\
& \frac{1}{n}(\boldsymbol{w}^T\boldsymbol{X}^T - \boldsymbol{y}^T)(\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}) = && (2.7)\\
& \frac{1}{n}(\boldsymbol{w}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{w} - \boldsymbol{w}^T\boldsymbol{X}^T\boldsymbol{y} - \boldsymbol{y}^T\boldsymbol{X}\boldsymbol{w} + \boldsymbol{y}^T\boldsymbol{y}) = && (2.8)\\
& \frac{1}{n}(\boldsymbol{w}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{w} - 2\boldsymbol{y}^T\boldsymbol{X}\boldsymbol{w} + \boldsymbol{y}^T\boldsymbol{y}) && (2.9)
\end{aligned}$$

To find the minimum of the loss we need to set its gradient equal to zero. The gradient is

$$\boldsymbol{\nabla}\mathcal{L}(\boldsymbol{w}) = \begin{pmatrix} \frac{\partial \mathcal{L}(\boldsymbol{w})}{\partial w_1} \\ \vdots \\ \frac{\partial \mathcal{L}(\boldsymbol{w})}{\partial w_m} \end{pmatrix} = \begin{pmatrix} \partial_1 \mathcal{L}(\boldsymbol{w}) \\ \vdots \\ \partial_m \mathcal{L}(\boldsymbol{w}) \end{pmatrix} \tag{2.10}$$

We'll calculate the gradient in two different ways. In the first we'll start with version of the loss from (2.5). We start of by differentiating the loss with respect to w_a where a is some number in $1, \ldots, m$. I often like to chose an index a here as it doesn't get in the way with the usual summation indices i, j, k etc.

$$\begin{aligned}
\partial_a \mathcal{L}(\boldsymbol{w}) =& \partial_a \frac{1}{n}\sum_{i=1}^{n}\left(\sum_{j=1}^{m} x_{ij}w_j - y_i\right)^2 = \text{(chain rule)} = && (2.11)\\
=& \frac{2}{n}\sum_{i=1}^{n}\left(\sum_{j=1}^{m} x_{ij}w_j - y_i\right)\partial_a\left(\sum_{j=1}^{m} x_{ij}w_j - y_i\right) = && (2.12)\\
=& \frac{2}{n}\sum_{i=1}^{n}\left(\sum_{j=1}^{m} x_{ij}w_j - y_i\right)\left(\sum_{j=1}^{m} x_{ij}\underbrace{\partial_a w_j}_{\delta_{aj}} - \underbrace{\partial_a y_i}_{0}\right) = && (2.13)\\
=& \frac{2}{n}\sum_{i=1}^{n}\left(\sum_{j=1}^{m} x_{ij}w_j - y_i\right)\left(\sum_{j=1}^{m} x_{ij}\delta_{aj}\right) = \text{(only } x_{ia} \text{ survives)} = && (2.14)\\
=& \frac{2}{n}\sum_{i=1}^{n}\left(\sum_{j=1}^{m} x_{ij}w_j - y_i\right) x_{ia} = && (2.15)\\
=& \frac{2}{n}\sum_{i=1}^{n}\sum_{j=1}^{m} x_{ai}^T x_{ij}w_j - x_{ai}^T y_i = && (2.16)\\
=& \frac{2}{n}\left(\sum_{i=1}^{n} x_{ai}^T \sum_{j=1}^{m} x_{ij}w_j - \sum_{i=1}^{n} x_{ai}^T y_i\right) = && (2.17)\\
=& \frac{2}{n}\left(\sum_{i=1}^{n} x_{ai}^T[\boldsymbol{X}\boldsymbol{w}]_i - [\boldsymbol{X}^T\boldsymbol{y}]_a\right) = && (2.18)\\
=& \frac{2}{n}\left([\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{w}]_a - [\boldsymbol{X}^T\boldsymbol{y}]_a\right) = && (2.19)\\
=& \frac{2}{n}\left(\left[\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{w} - \boldsymbol{X}^T\boldsymbol{y}\right]_a\right). && (2.20)
\end{aligned}$$

Thus we have that the derivative of the loss with respect to w_a is the a-th element of the vector above thus

$$\boldsymbol{\nabla}\mathcal{L}(\boldsymbol{w}) = \frac{2}{n}\left(\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{w} - \boldsymbol{X}^T\boldsymbol{y}\right) \tag{2.21}$$