Review

Pulkit Tandon

ENGR108
Stanford University

slides adopted from Prof. Stephen Boyd

August 11, 2022
Announcements
Outline

Final Exam

This lecture
Final Exam

- Take-home, open class notes but not open resource
- Covers everything up to Least Squares Model Fitting (except Validation and Feature Engineering)
- Released Thu midnight (canvas), due Sun midnight (gradescope)
- ED posts will be private. No clarifications during exam duration
- Submit screenshots of code and relevant results/plots for specific questions requiring Julia
Outline

Final Exam

This lecture
Since the textbook and slides are very comprehensive, this review guide is designed to highlight some of the most important concepts of the course in a concise manner.

Each chapter is summarized and many non-essential concepts were omitted. We hope this will help you prepare for the final.
Review
Outline

Vectors

Linear functions
Norm and Distance
Clustering
Linear independence
Matrices
Linear Equations
Matrix multiplication
Matrix inverses
Least Squares
Least squares data fitting
We can write a vector in one of three ways

\[ a = (1, 7, 3) = \begin{bmatrix} 1 \\ 7 \\ 3 \end{bmatrix} = \begin{pmatrix} 1 \\ 7 \\ 3 \end{pmatrix} \]

An \( n \)-vector is a vector with \( n \) entries (sometimes, we say this vector has length \( n \), but in some contexts length can mean other things). The \( i^{th} \) entry of the \( n \)-vector \( x \) is denoted by \( x_i \).

**Caution:** Sometimes \( x_i \) refers to the \( i^{th} \) vector in a list of vectors, instead of the \( i^{th} \) entry of the vector \( x \). You will always be able to determine this from the context of the question.

If \( a \) is a \( k \)-vector and \( b \) is a \( p \)-vector, we can define a block vector \( x = (a, b) \) of length \( k + p \).

If \( x \) is an \( n \)-vector, then \( x_{i:j} = (x_i, \ldots, x_j) \) is a vector of length \( j - i + 1 \).
Vectors

- A vector is called *sparse* if it has very few non-zero entries. The function $\text{nnz}(x)$ returns the number of non-zero entries in $x$. If $\text{nnz}(x)$ is much smaller than the number of entries in $x$, then $x$ is sparse.
- Both $\mathbf{0}$ and $\mathbf{1}$ are vectors of undefined length, and we can generally infer their length from the context of the question.
- A *unit vector* $\mathbf{e}_i$ is a vector with one in its $i$th entry and zero in every other entry.
The inner product $a^T b$ between two $n$-vectors $a$ and $b$ is

$$a^T b = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n$$

Some important properties of the inner product include

- **Commutativity:** $a^T b = b^T a$
- **Linearity:** $a^T (\alpha b + \beta c) = \alpha (a^T b) + \beta (a^T c)$
  
  (where $\alpha$, $\beta$ are scalars and $a, b, c$ are $n$-vectors)

A linear combination of $n$-vectors $x_1, \ldots, x_k$ is a vector $y$ defined by

$$y = \alpha_1 x_1 + \cdots + \alpha_k x_k$$

where $\alpha_1, \ldots, \alpha_k$ are scalars.
Outline

Vectors

**Linear functions**

Norm and Distance

Clustering

Linear independence

Matrices

Linear Equations

Matrix multiplication

Matrix inverses

Least Squares

Least squares data fitting
Linear functions

When we define a function $f : \mathbb{R}^n \to \mathbb{R}$, we mean that the function $f$ has $n$-vectors as inputs, and scalars as outputs. This means that when we write $f(x)$, $x$ must be an $n$-vector, and $f(x)$ is a scalar.

A function $f : \mathbb{R}^n \to \mathbb{R}$ is a linear function if and only if it satisfies

$$f(\alpha x + \beta y) = \alpha f(x) + \beta f(y)$$

for all $n$-vectors $x, y$ and all scalars $\alpha, \beta$.

Equivalently, $f$ is a linear function if and only if it $f(x) = c^T x$ for some $n$-vector $c$. The entries of $c$ cannot depend on $x$.

If we know $f$ is linear, the entries of $c$ must be

$$c = \begin{bmatrix} f(e_1) \\ \vdots \\ f(e_n) \end{bmatrix}$$

be sure to understand why.
Linear functions

A function $f : \mathbb{R}^n \to \mathbb{R}$ is a affine function if and only if it satisfies

$$f(\alpha x + \beta y) = \alpha f(x) + \beta f(y)$$

for all $n$-vectors $x, y$ and all scalars $\alpha, \beta$ such that $\alpha + \beta = 1$.

All linear functions are affine, but not all affine functions are linear.

Equivalently, $f$ is a affine function if and only if it $f(x) = c^T x + b$ for some $n$-vector $c$ and some scalar $b$. The entries of $c, b$ cannot depend on $x$.

If we know $f$ is affine, we know $b = f(0)$ and the entries of $c$ must be

$$c = \begin{bmatrix} f(e_1) - f(0) \\ \vdots \\ f(e_n) - f(0) \end{bmatrix}$$

be sure to understand why.
Outline

Vectors
Linear functions

Norm and Distance

Clustering
Linear independence

Matrices
Linear Equations
Matrix multiplication
Matrix inverses

Least Squares
Least squares data fitting
The norm of an $n$-vector $x$ is

$$||x|| = \sqrt{x^T x} = \sqrt{x_1^2 + \cdots + x_n^2}$$

Some important properties of the norm include:

- **Homogeneity**: $||\alpha x|| = |\alpha||x||$ where $\alpha$ is a scalar.
- **Cauchy-Schwarz**: $|x^T y| \leq ||x|| ||y||$ (see the book for a proof)
- **Triangle inequality**: $||x + y|| \leq ||x|| + ||y||$

The distance between two $n$-vectors $x$ and $y$ is $||x - y||$. 
Some definitions you should know and understand

- **Root mean square:**

  \[
  \text{rms}(x) = \frac{\|x\|}{\sqrt{n}}
  \]

  where \( n \) is the number of entries in \( x \).

- **Average:**

  \[
  \text{avg}(x) = \frac{\mathbf{1}^T x}{n}
  \]

  where \( n \) is the number of entries in \( x \).

- **Angle:**

  \[
  \theta = \cos^{-1}\left(\frac{x^T y}{\|x\| \|y\|}\right)
  \]
For the following definitions, let $\tilde{x} = x - \text{avg}(x) \mathbf{1}$. The vector $\tilde{x}$ is often called the *demeaned* version of $x$.

- **Standard deviation:**

  $$\text{std}(x) = \text{rms}(\tilde{x})$$

- **Correlation:**

  $$\text{corr}(x, y) = \frac{\tilde{x}^T \tilde{y}}{\|\tilde{x}\| \|\tilde{y}\|}$$

  Using Cauchy-Schwarz, we know $-1 \leq \text{corr}(x, y) \leq 1$. 


Outline

Vectors
Linear functions
Norm and Distance

Clustering
Linear independence

Matrices
Linear Equations
Matrix multiplication
Matrix inverses
Least Squares
Least squares data fitting
Some rough pseudocode for the $k$-means algorithm:

1. Choose $k$ initial centroids $z_1, \ldots, z_k$.
   Repeat until (2) and (3) until convergence
2. For each datapoint $x_j$, compute distances $\|x_j - z_1\|, \ldots, \|x_j - z_k\|$. Find the closest centroid, i.e. choose $i$ to minimize $\|x_j - z_i\|$. Assign $x_j$ to the group $G_i$.
3. For each group $G_i$, let $x_{j1}, \ldots, x_{j_{ni}}$ be the vectors assigned to group $G_i$. Set $z_i$ to be the mean of these vectors, i.e.

$$z_i = \frac{x_{j1} + \cdots + x_{j_{ni}}}{n_i}$$
A set of vectors $a_1, \ldots, a_k$ are called **linearly dependent** if

$$\beta_1 a_1 + \cdots + \beta_k a_k = 0$$

for some $\beta_1, \ldots, \beta_k$ that are not all zero.

- Another way of phrasing this is saying a set of vectors are linearly dependent if the zero vector can be formed as a linear combination of these vectors.
- **Caution:** Linear dependence is a property of a set or collection of vectors, not of an individual vector.

A set of vectors $a_1, \ldots, a_k$ are called **linearly independent** if they are not linearly dependent. Equivalently, the expression

$$\beta_1 a_1 + \cdots + \beta_k a_k = 0$$

only holds when $\beta_1 = \cdots = \beta_k = 0$.

- If $x = \beta_1 a_1 + \cdots + \beta_k a_k$ and $a_1, \ldots, a_k$ are linearly independent, then the coefficients $\beta_1, \ldots, \beta_k$ used to form $x$ are unique.
Linear Independence

▶ A basis is a set of $n$ linearly independent $n$-vectors.
  - If $a_1, \ldots, a_n$ are a basis, then any $n$-vector can be written as a linear combination of $a_1, \ldots, a_n$.

▶ The independence-dimension inequality states that any collection of $n + 1$ or more $n$-vectors is linearly dependent.

▶ A collection of vectors $a_1, \ldots, a_k$ is orthogonal if $a_i^T a_j = 0$ whenever $i \neq j$.
  - Orthogonal vectors are always linearly independent.

▶ A collection of vectors $a_1, \ldots, a_k$ is orthonormal if it is orthogonal and $\|a_i\| = 1$ for $i = 1, \ldots, k$.
  - If $x$ is a linear combination of orthonormal vectors $a_1, \ldots, a_k$ defined by $x = \beta_1 a_1 + \cdots + \beta_k a_k$, then the coefficients are given by $\beta_i = a_i^T x$. 

Linear independence
Given a set of vectors $a_1, \ldots, a_k$, the Gram-Schmidt algorithm produces a set of orthonormal vectors $q_1, \ldots, q_k$ or terminates early if $q_1, \ldots, q_k$ are linearly dependent. The algorithm is roughly given by for $i = 1, \ldots, k$

1. Orthogonalize: $\tilde{q}_i = a_i - (q_1^T a_i)q_1 - \cdots - (q_{i-1}^T a_i)q_{i-1}$
2. Test for dependence: if $\tilde{q}_i = 0$, quit
3. Normalize: $q_i = \tilde{q}_i / \|\tilde{q}_i\|$
Outline

Vectors
Linear functions
Norm and Distance
Clustering
Linear independence

Matrices
Linear Equations
Matrix multiplication
Matrix inverses
Least Squares
Least squares data fitting
An $m \times n$ matrix is a rectangular array of numbers with $m$ rows and $n$ columns. For example,

$$A = \begin{bmatrix} 1 & -1 & 6 \\ 2 & 2 & 9 \end{bmatrix}$$

To access the entry in $i^{th}$ row and $j^{th}$ column of a matrix $A$, we write $A_{ij}$. You can also access several entries at once using slices, $A_{i:k,j:p}$. For example, $A_{1,2:3} = [-1 \ 6]$. This is similar to slices of vectors.

Matrices can be defined in terms of other matrices. Given matrices $A, B, C, D$, we can define a block matrix

$$M = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$$

be sure the dimensions match.
A vector can be thought of as an $n \times 1$ matrix. We sometimes call vectors \textit{column vectors}, to distinguish them from $1 \times n$ matrices known as \textit{row vectors}. It is often useful to define matrices as blocks of column or row vectors. Let $a_1, \ldots, a_n$ be column vectors and $b_1, \ldots, b_m$ be row vectors. We could define the matrices $A$ and $B$ by

$$A = \begin{bmatrix} a_1 & \cdots & a_n \end{bmatrix} \quad \quad \quad B = \begin{bmatrix} b_1 \\ \vdots \\ b_m \end{bmatrix}$$

The \textit{transpose} $A^T$ of a matrix $A$ is defined by $(A^T)_{ij} = A_{ji}$. The columns of $A^T$ are the rows of $A$, and vice versa. The transpose of a black matrix is

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}^T = \begin{bmatrix} A^T & C^T \\ B^T & D^T \end{bmatrix}$$

- We often define vectors $a_1, \ldots, a_m$ and let the row vectors $a_1^T, \ldots, a_m^T$ be the rows of a matrix $A$.
- A matrix is \textit{symmetric} if $A^T = A$.
- Many properties of the transpose are not too complicated to derive, such as $(A + B)^T = A^T + B^T$ and $(\alpha A)^T = \alpha A^T$. 

Matrices
Some important matrices include the zero matrix (usually denoted just as $0$) and the identity matrix $I$. Also be familiar with the notion of an upper triangular matrix, a lower triangular matrix, and a diagonal matrix.

Be sure to understand matrix-matrix addition $A + B$ and scalar-matrix multiplication $\beta A$, as well as relevant properties (e.g. $\beta(A + B) = \beta A + \beta B$).
Matrix-vector multiplication between an $m \times n$ matrix $A$ and an $n$-vector $x$ is defined as

$$Ax = x_1a_1 + \cdots + x_na_n = \begin{bmatrix} \tilde{a}_1^T x \\ \vdots \\ \tilde{a}_m^T x \end{bmatrix}$$

where $a_1, \ldots, a_n$ are the columns of $A$ and $\tilde{a}_1^T, \ldots, \tilde{a}_m^T$ are the rows of $A$.

The columns $a_1, \ldots, a_n$ of a matrix $A$ are linearly dependent if $Ax = 0$ for some $x \neq 0$. 

Matrices
Outline

Vectors
Linear functions
Norm and Distance
Clustering
Linear independence
Matrices

**Linear Equations**

Matrix multiplication
Matrix inverses
Least Squares
Least squares data fitting
Linear Equations

▶ When we denote a function \( f : \mathbb{R}^n \to \mathbb{R}^m \), we mean the function \( f(x) \) has \( n \)-vectors as inputs and \( m \)-vectors as outputs. We often refer to the \( i \)th output \( (f(x))_i \) as \( f_i(x) \).

▶ A function \( f : \mathbb{R}^n \to \mathbb{R}^m \) is a linear function if and only if it satisfies

\[
f(\alpha x + \beta y) = \alpha f(x) + \beta f(y)
\]

for all \( n \)-vectors \( x, y \) and all scalars \( \alpha, \beta \).

▶ Equivalently, \( f \) is a linear function if and only if it \( f(x) = Ax \) for some \( m \times n \) matrix \( A \). The entries of \( A \) cannot depend on \( x \).

▶ If we know \( f \) is linear, the columns of \( A \) must be

\[
A = \begin{bmatrix} f(e_1) & \cdots & f(e_n) \end{bmatrix}
\]

be sure to understand why.

▶ Note how the linear functions in chapter 2 are just the special case when \( m = 1 \).
A function \( f : \mathbb{R}^n \rightarrow \mathbb{R}^m \) is a *affine function* if and only if it satisfies
\[
f(\alpha x + \beta y) = \alpha f(x) + \beta f(y)
\]
for all \( n \)-vectors \( x, y \) and all scalars \( \alpha, \beta \) such that \( \alpha + \beta = 1 \).

All linear functions are affine, but not all affine functions are linear.

Equivalently, \( f \) is a *affine function* if and only if it \( f(x) = Ax + b \) for some \( m \times n \) matrix \( A \) and some \( m \)-vector \( b \). The entries of \( A, b \) **cannot** depend on \( x \).

If we know \( f \) is affine, we know \( b = f(0) \) and the columns of \( A \) must be
\[
A = [f(e_1) - f(0) \quad \ldots \quad f(e_n) - f(0)]
\]
be sure to understand why.
A system $m$ linear equations with $n$ variables $x_1, \ldots, x_n$ has the form

$$A_{11}x_1 + \cdots + A_{1n}x_n = b_1$$

$$\vdots$$

$$A_{m1}x_1 + \cdots + A_{mn}x_n = b_m$$

which can be written succinctly as $Ax = b$.

Some terminology: the system is called *overdetermined* if $m > n$ (the matrix $A$ is tall), *underdetermined* if $m < n$ (the matrix $A$ is wide), and *square* if $m = n$ (the matrix $A$ is square). If $b = 0$, the system is called *homogeneous*.
Linear Equations

Let $x_1, \ldots, x_T$ be a sequence of $n$-vectors and $A_1, \ldots, A_T$ be a sequence of $n \times n$ matrices. A *linear dynamical system* is a relationship of the form

$$x_{t+1} = A_t x_t$$

One variation of the linear dynamical system model is a linear dynamical system *with inputs*:

$$x_{t+1} = A_t x_t + B_t u_t + c_t$$

where $c_1, \ldots, c_T$ is a sequence of $n$-vectors, $u_1, \ldots, u_T$ be a sequence of $k$-vectors, and $B_1, \ldots, B_T$ be a sequence of $n \times k$ matrices. The vectors $u_t$ are often called the *inputs*. The vectors $c_t$ are often called the *offsets*. 
Outline

Vectors
Linear functions
Norm and Distance
Clustering
Linear independence
Matrices
Linear Equations

Matrix multiplication
Matrix inverses
Least Squares
Least squares data fitting
Let $A$ be an $m \times n$ matrix and $B$ be an $n \times k$ matrix. Denote $\tilde{a}_1^T, \ldots, \tilde{a}_m^T$ to be the rows of $A$, and $b_1, \ldots, b_k$ to be the columns of $B$. Then the $m \times k$ matrix-matrix product $AB$ is given by

$$AB = \begin{bmatrix} \tilde{a}_1^T b_1 & \cdots & \tilde{a}_1^T b_k \\ \vdots & \ddots & \vdots \\ \tilde{a}_m^T b_1 & \cdots & \tilde{a}_m^T b_k \end{bmatrix}$$

Two convenient ways to rewrite this matrix-matrix product are

$$AB = [\begin{bmatrix} \tilde{a}_1^T B \\ \vdots \\ \tilde{a}_m^T B \end{bmatrix} ]$$

Matrix multiplication
Some important properties and things to be aware of

- **Associativity**: \( A(BC) = (AB)C \)
- **Distributivity**: \( A(B + C) = AB + AC \)
- **Transpose of a product**: \( (AB)^T = B^T A^T \)
- Matrix multiplication is NOT commutative, i.e. it is not always true \( AB \neq BA \) (sometimes one of these expressions won’t even make sense).

The **Gram matrix** of a set of vectors \( a_1, \ldots, a_n \) is \( G = A^T A \), where \( A \) is an \( m \times n \) matrix whose columns are \( a_1, \ldots, a_n \).

The matrix power \( A^p \) denotes \( A \cdots A \) (with \( p \) copies of \( A \) in the product). For example, \( A^2 = AA \).
Matrix multiplication

Let $Q$ be an $n \times k$ matrix with columns $q_1, \ldots, q_k$. If the vectors $q_1, \ldots, q_k$ are orthonormal, then

$$Q^T Q = I$$

Note that this does NOT necessarily imply $QQ^T = I$ (we will see when this is the case).

If a matrix $Q$ has orthonormal columns, then

- $Q$ preserves inner products, i.e. $(Qx)^T (Qy) = x^T y$.
- $Q$ is preserves norms, i.e. $\|Qx\| = \|x\|$.
- $Q$ preserves angles, i.e $\angle(Qx, Qy) = \angle(x, y)$

A square matrix (i.e $n = k$) with orthornormal columns is called **orthogonal**. Its columns form an orthonormal basis for $m$-vectors. In addition to satisfying $Q^T Q = I$, orthogonal matrices also satisfy $QQ^T = I$ (this comes from the properties of inverses).
The matrix form of Gram-Schmidt is called QR factorization. Let $A$ be an $n \times k$ matrix with linearly independent columns $a_1, \ldots, a_k$. Then the orthonormal vectors $q_1, \ldots, q_k$ output by Gram-Schmidt can define a matrix $Q$ and we can write $A = QR$, where $R$ is an upper triangular matrix defined by

\[
R_{ij} = q_i^T a_j \text{ for } i < j \\
R_{ii} = \|\tilde{q}_i\| \\
R_{ij} = 0 \text{ for } i > j
\]
Outline

Vectors
Linear functions
Norm and Distance
Clustering
Linear independence
Matrices
Linear Equations
Matrix multiplication
Matrix inverses
Least Squares
Least squares data fitting
Consider an $m \times n$ matrix $A$. Then $B$ is a left inverse of $A$ if $BA = I$.

- The matrix $A$ has a left inverse if and only if it has linearly independent columns (try to show this yourself, or at least review the proof in the book). This means wide matrices cannot have left inverses. **Caution:** This does not imply all tall matrices do have left inverses.
- Given an $m \times n$ matrix $A$ with left inverse $B$ and an $m$-vector $b$, if $Ax = b$ has a solution (i.e. there is some $n$-vector $x$ such that $Ax = b$), then the solution is unique.
The matrix $C$ is a right inverse of $A$ if $AC = I$.

- If $A$ has a left inverse $B$, then $A^T$ has a right inverse $B^T$ (since $(AB)^T = B^T A^T$).
- The matrix $A$ has a right inverse if and only if it has linearly independent rows. This means tall matrices cannot have right inverses.
- Given an $m \times n$ matrix $A$ with right inverse $C$ and an $m$-vector $b$, then one solution to $Ax = b$ must be $Cb$ since $Ax = A(Cb) = (AC)b = Ib = b$.

Note the difference between left and right inverses here: a left inverse implies that a solution is unique (if it exists), whereas a right inverse implies a solution always exists (but it may not be unique).

A matrix may have more than one right or left inverse. In fact, if a matrix has two left or right inverses, it has infinitely many left or right inverses, respectively.
Matrix inverses

- If the $n \times n$ matrix $A$ has both a left inverse $B$ and a right inverse $C$ then

\[ B = B(AC) = (BA)C = C \]

So $B = C$, implying any left inverse is equal to any right inverse. Hence, the left and right inverse are unique and equal, and we define the inverse $A^{-1} = B = C$. Here we say $A$ is invertible.

- If $A$ is invertible, then for any $n$-vector $b$, the solution to $Ax = b$ exists and is uniquely given by $x = A^{-1}b$.

- For a square $n \times n$ matrix $A$, the following five conditions are equivalent
  - $A$ is invertible
  - $A$ has linearly independent columns / $A$ has linearly independent rows
  - $A$ has a right inverse / $A$ has a left inverse

In other words, if you show one of these five properties, the rest automatically follow.
If $A$ is invertible, then $(A^T)^{-1} = (A^{-1})^T$. Since the order of the transpose and the inverse does not matter, we often denote $(A^T)^{-1} = A^{-T}$.

If $A, B$ are both invertible, then $(AB)^{-1} = B^{-1}A^{-1}$. **Caution:** If $A$ or $B$ is not invertible, then $(AB)^{-1}$ might still exist.

Triangular matrices are invertible if all their diagonal elements are non-zero.

If $Q$ is orthogonal, then $Q^{-1} = Q^T$. Hence $Q^TQ = QQ^T = I$.

If we compute the QR-factorization $A = QR$, then $A^{-1} = R^{-1}Q^T$, which provides an algorithmic way for finding $A^{-1}$ (since we can invert triangular and orthogonal matrices).
Matrix inverses

If $A$ has linearly independent columns, then the Gram-matrix $A^T A$ is invertible. Similarly, if $A$ has linearly independent rows, then $AA^T$ is invertible.

The pseudo-inverse for a tall or square matrix is $A^\dagger = (A^T A)^{-1} A^T$. The pseudo-inverse is one possible left inverse for $A$.

The pseudo-inverse for a wide or square matrix is $A^\dagger = A^T (A A^T)^{-1}$. The pseudo-inverse is one possible right inverse for $A$. For square matrices, both pseudo-inverses are the same.
Outline

Vectors
Linear functions
Norm and Distance
Clustering
Linear independence
Matrices
Linear Equations
Matrix multiplication
Matrix inverses

Least Squares
Least squares data fitting
Sometimes we wish to find $x$ such that $Ax = b$, but no such $x$ exists. Instead we settle for $Ax \approx b$. Given an $m \times n$ matrix $A$ with linearly independent columns and an $m$-vector $b$, the least squares problem is

$$\text{minimize } \|Ax - b\|^2$$

where we are minimizing over all choices of $x$.

The solution to the least squares problem is $\hat{x} = A^+b = (A^TA)^{-1}A^Tb$. Remember that $A^+$ is the left psuedoinverse of $A$.

- If you derive this solution via calculus (see the book), you will find the normal equations

$$A^TA\hat{x} = A^Tb$$

notice that $A^TA$ is the Gram-matrix of $A$, which is invertible because $A$ has linearly independent columns.
The optimal residual $\hat{r} = A\hat{x} - b$ is orthogonal to any linear combination of the columns of $A$. More concisely, for any $n$-vector $z$, we know

$$\hat{r}^T(Az) = 0$$

Using QR-factorization, we can write $A = QR$ and find $A^\dagger = R^{-1}Q^T$. Then $\hat{x} = R^{-1}Q^Tb$. This is an efficient way of computing the least squares solution (more efficient than computing and inverting the Gram-matrix).

In the matrix least squares problem, we replace the $m$-vector $b$ with an $m \times k$ matrix $B$. Our new problem is

$$\text{minimize } \|AX - B\|^2$$

which has solution $X = A^\dagger B$. 
Outline

Vectors
Linear functions
Norm and Distance
Clustering
Linear independence
Matrices
Linear Equations
Matrix multiplication
Matrix inverses
Least Squares

Least squares data fitting
In the data fitting problem, we are given inputs $x^{(1)}, \ldots, x^{(N)}$ and outputs $y^{(1)}, \ldots, y^{(N)}$ where $x^{(i)}$ is an $n$-vector and $y^{(i)}$ is a scalar. We believe the inputs and outputs are related by

$$y \approx \hat{f}(x) = \theta_1 f_1(x) + \cdots + \theta_p f_p(x)$$

where $\theta = (\theta_1, \ldots, \theta_p)$ is a $p$-vector of unknown coefficients or model parameters, and the functions $f_1, \ldots, f_p : \mathbb{R}^n \to \mathbb{R}$ are basis functions or feature mappings.

We can solve this with the least squares problem

$$\text{minimize } \|A\theta - y\|^2$$

where

$$A = \begin{bmatrix}
f_1(x^{(1)}) & \cdots & f_p(x^{(1)}) \\
\vdots & \ddots & \vdots \\
f_1(x^{(N)}) & \cdots & f_p(x^{(N)})
\end{bmatrix} \quad \quad y = \begin{bmatrix}
y^{(1)} \\
\vdots \\
y^{(N)}
\end{bmatrix}$$
Often we let $f_1(x) = 1$, i.e. we add a constant basis function. This corresponds to the first column of $A$ being the ones vector $\mathbf{1}$.

*Example:* In scalar polynomial regression, each $x^{(i)}$ is a scalar and the basis functions are $f_1(x) = 1, f_2(x) = x, \ldots, f_p(x) = x^{p-1}$.

Given a model $f(x) = \theta_1 f_1(x) + \ldots + \theta_p f_p(x)$ and data $x^{(1)}, \ldots, x^{(N)}$ and $y^{(1)}, \ldots, y^{(N)}$, the RMS prediction error is

$$
\frac{1}{\sqrt{N}} \sqrt{\sum_{i=1}^{N} (f(x^{(i)}) - y^{(i)})^2} = \frac{1}{\sqrt{N}} \|A\theta - y\| = \text{rms}(A\theta - y)
$$
To fit the data well, we often fit the least squares model on training data $x^{(1)}, \ldots, x^{(N)}, y^{(1)}, \ldots, y^{(N)}$ then measure its performance on the test data $x^{(N+1)}, \ldots, x^{(M)}, y^{(N+1)}, \ldots, y^{(M)}$. We usually measure performance by computing the RMS prediction error.

If a model fits the training data very well but does not fit the test data well, we say the model is over-fit. This often happens when you add too many features.

If we want to choose a model that makes good predictions from new, unseen data, we should choose the model that minimizes the RMS prediction error on the test data.