LINEARE ABBILDUNGEN und MATRIZENRECHNUNGEN (as seen by HGFei)

Betrachten wir nun die Tatsache, dass Matrix Multiplikation eine elegante Form der Bildung von Linear-Kombinationen ist. Im Fall der Matrix-Multiplikation von links wendet man eine Matrix $A = [\overrightarrow{a_1}, \overrightarrow{a_2}, \ldots, \overrightarrow{a_n}]$ auf einen Spaltenvektor $\overrightarrow{x}$, um die Linearkombination $\overrightarrow{b} = \sum_{k=1}^{n} x_k \overrightarrow{a_k}$ zu bilden, umgekehrt bildet die Matrix Multiplikation von rechts, angewendet auf einen Zeilenvektor der Länge m eine Linearkombination von Zeilen. Die lineare Abbildung, die von einer $m \times n$ Matrix induziert wird, ist also ein "Bilden von Linearkombination". Wenn diese Abbildung *injektiv* ist (!das ist genau die Definition des Begriffes der linearen Unabhängigkeit!), dann muss es auch möglich sein, die (eindeutigen) Koeffizienten zu bestimmen. Bekanntlich ist das gleichbedeutend mit der Frage, ob der *Kern* von A nichttrivial (d.h. von $\{0\}$ verschieden) ist. Andererseits bedeutet *Surjektivität* genau, dass die Spalten der Matrix A ein Erzeugendensystem sind. Der Bildraum ist aber genau der Spaltenraum, sodass die Surjektivität dadurch charakterisiert werden kann, dass das orthog. Komplement des Spaltenraums, also der Nullraum von A' trivial ist. Kurz, $x \mapsto A * x$ ist surjektiv genau dann wenn $y \mapsto A' * y$ injektiv ist.

Im Idealfall ($m = n$), ist die Abbildung *bijektiv*, also injektiv und surjektiv, was aus der Sicht der Matrizenrechnung genau auf die Invertierbarkeit der Matrix A hinausläuft. Wir sagen, die Spalten (oder im Endeffekt auch die Zeilen, weil mit A auch A' invertierbar ist, weil ja $inv(A') = inv(A)'$ gilt, in MATLAB Schreibweise) von A bilden eine Basis von $\mathbb{C}^m$ (oder $\mathbb{R}^m$). Geometrisch bedeutet es ganz einfach, dass wir ein *Koordinatensystem* für den ganzen $\mathbb{C}^m$ haben, jeder (!) Vektor $b \in \mathbb{C}^m$ läßt sich auf eindeutige Weise als Linearkombination der Spalten darstellen (in Worten), oder in Matrix-Symbolen dargestellt, für *jede rechte Seite* $\overrightarrow{b}$ ist die lineare Gleichung

$$A * \overrightarrow{x} = \overrightarrow{b}$$

eindeutig lösbar, d.h. es gibt *genau ein* $\overrightarrow{x} \in \mathbb{C}^m$ sodass $\overrightarrow{b} = \sum_{k=1}^{n} x_k \overrightarrow{a_k}$ gilt.

Da nun das Anwenden der Matrix (Matrixmultiplikation von links) den Übergang von Koeffizienten- zu Linearkombinationen (gegeben die Spaltenvektoren von A) reprästentiert ist klar, dass der Übergang, d.h. die Bestimmung der Koeffizientenaus der Linearkombination, durch Anwendung der *inversen Matrix* möglich ist. Wir können also $\overrightarrow{x}$ einfach bestimmen, indem wir in MATLAB den folgenden Befehl eingeben:

```
x = inv(A) *  b;
```

Als Alternative (zunächst einmal schreibtechnisch) bietet sich das Kommado

```
x =  A\b;   % identisch mit:  x=pinv(A) * b;  auch f\"ur rechteckige Matrizen
```

an (da ja mit A von links multipliziert wird, muss nun "von links" durch A dividiert werden, daher anstatt des gewöhnlichen Bruchstrichst `/` (eines slash) ein backslash `\`.

Wir halten fest: Betrachten wir die Spalten einer (invertierbaren) Matrix $A = [\overrightarrow{a_1}, \overrightarrow{a_2}, \ldots, \overrightarrow{a_n}]$ als Kollektion von Spaltenmatrizen, so ist für den Vektor $\overrightarrow{b} = \sum_{k=1}^{n} x_k \overrightarrow{a_k}$ die Folge $\overrightarrow{x}$ die Folge der (eindeutigen) *Koeffizienten* von $\overrightarrow{x}$ der Basis $[\overrightarrow{a_1}, \overrightarrow{a_2}, \ldots, \overrightarrow{a_n}]$. Natürlich gilt:

$$\overrightarrow{x} = A^{-1} * \overrightarrow{b}.$$

Manchmal ist es auch sinnvoll (um die Vorstellung von einer Matrix zu verallgemeinern) das Symbol $\mathscr{A}$ für das geordnete n-Tupel von Basisvektoren $[\overrightarrow{a_1}, \overrightarrow{a_2}, \ldots, \overrightarrow{a_n}]$ zu verwenden. Dann ist es naheliegend (ohne auf die konkrete Berechnungsmethode einzugehen), die (nach Definition einer Basis) eindeutige Koeffizienten-Folge $\overrightarrow{x}$ durch ihre funktionale Rolle, es *sind die Koeffizientenzur Darstellung von* $\overrightarrow{b}$ bezüglich der Basis $\mathscr{A}$, mit dem Symbol $[\overrightarrow{b}]_{\mathscr{A}}$. [1]

Bevor wir auf den Begriff des *Basiswechsel* eingehen, sei kurz auf den Begriff der Biorthogonalen Systeme hingewiesen, die man als Verallgemeinerung von Orthonormalsystemen angesehen werden können.

Erinnern wir uns kurz an die Lineare Algebra: Dort wurden wohl Orthogonalsystem definiert, als Vektoren in einem Raum mit Skalarprodukt (das muss nicht nur der $\mathbb{R}^d$ oder der $\mathbb{C}^k$ mit dem Standard Skalarproduct sein, das kann auch allgemeiner sein!), die die Eigenschaft haben, paarweise. Die Spalten einer Matrix O bilden genau dann ein Orthogonalsystem, wenn die Matrix $O' * O$ eine Diagonalmatrix ist. In der Diagonale dieser Matrix stehen natürlich die Quadrate der Längen der Spaltenvektoren von O. Es ist klar, dass ein Orthogonalsystem (soferne kein Nullvektor vorkommt) linear unabhängig sind. In Matrix-Schreibweise ausgedrückt: OWenn $O * \overrightarrow{x} = \overrightarrow{0}$ gilt, dann gilt natürlich auch $(O' * O) * \overrightarrow{x} = \overrightarrow{0}$. Das Linksmultiplizieren eines Vektors mit einer Diagonalmatrix multipliziert aber nur die Koordinaten mit den Werten in der Diagonale (die hier alle ungleich Null sind!), also muss $\overrightarrow{x} = \overrightarrow{0}$ gelten.

Nun nennt man eine $n \times n$ Matrix *orthogonal*[2] wenn ihre Spalten eine *Orthonormalbasis* bilden, d.h. wenn es ein Orthogonalsystem ist, dessen Vektoren alle normiert sind (d.h. Länge 1 haben). Das läßt sich wieder in Matrix-Schreibweise einfach ausdrücken:

$$O' * O == eye(n); \quad \text{oder äquivalent} \quad O * O' = eye(n)' = eye(n);$$

Schreiben wir wie üblich $O = [\overrightarrow{o_1}, \overrightarrow{o_2}, \ldots, \overrightarrow{o_n}]$, so kann die letze Identität auch in folgender Form geschrieben werden (das ist de facto eine andere Beschreibung *derselben* Identität!):

$$\overrightarrow{x} = \sum_{k=1}^{n} \langle \overrightarrow{x}, \overrightarrow{o_k} \rangle \overrightarrow{o_k} \quad \forall \overrightarrow{x} \in \mathbb{C}^n. \tag{1}$$

Das "schöne" an Orthonormalbasen ist also die Tatsache, dass einerseits die Koffizienten auf ganz einfache Weise (und mit geringem Rechenaufwand) bestimmt werden können [3]

---

[1] Die Verwendung der eckigen Klammern hat hier weniger Bedeutung, ausser dass sie daran erinnert, dass man in MATLAB einen echten (Spalten-) Vektor schreiben würde, und natürlich deutet der Index darauf hin, dass man eine *bestimmte* Basis im Sinne hat, und beim Übergang zu einer anderen Matrix hat man mit anderen Koeffizientenzu rechnen!

[2] Es sei hier darauf hingewiesen, dass dieser Begriff leider ziemlich irreführend ist, eine Erklärung folgt weiter unten!

[3] Im Gegensatz dazu muss im allgemeinen Fall die Matrix A invertiert werden, was einen wesentlich höheren Rechenaufwand bedeutet. Man bedenke nur, dass die Bestimmung von n reellen Skalarprodukten im $\mathbb{R}^n$ insgesamt n^2 Multiplikationen benötigen, während alleine schon die Matrixmultiplikation zweier $n \times n$-Matrizen insgesamt n^3 Multikplikationen erfordern, und die Kosten der Inversion sind in einer vergleichbaren Größenordnung. Man beachte dabei die "Fluch der Dimension". Eine Verzehnfachung der Dimension verursacht schon den tausendfachen Rechenaufwand für die Matrizmultiplikation, im Gegensatz zu einer "Verhundertfachung" im Falle von ONBs.

Möchte man eine analoge Aussage für eine allgemeine Basen gewinnen, d.h. möchte man auch in diesem Falle eine Familie von Vektoren bestimmen, sodass deren Skalarprodukte die "richtigen Koeffizienten" liefern, so ist klar, dass das auf die Suche nach einer Matrix B hinausläuft, für die gilt $A * B' = eye(n)$, bzw.

$$\overrightarrow{x} = \sum_{k=1}^{n} \langle \overrightarrow{x}, \overrightarrow{b_k} \rangle \overrightarrow{a_k} \quad \forall \overrightarrow{x} \in \mathbb{C}^n. \tag{2}$$

Ist ist sofort klar, dass dann gelten muss $B = (A^{-1})' = A'^{-1}$. Da diese Identität wiederum durch eine nun *BIorthogonlitätsrelation* beschrieben werden können, nämlich

$$\langle \overrightarrow{b_j}, \overrightarrow{a_k} \rangle = \delta_{j,k} = 0^4 \quad \text{für} \quad j \neq k, \quad \text{und} \quad 1 \quad \text{sonst} \tag{3}$$

nennt man das System der Spalten von B ein Biorthogonalsystem zum System der Spaltenvektoren von A (beide als geordnetes n-Tupel von Vektoren in $\mathbb{C}^n$ (oder wie immer $\mathbb{R}^n$) betrachtet.

Es ist eine einfache Übungsaufgabe festzustellen, dass einerseits diese Biorthogonalität-Relation symmetrisch in Bezug auf A und B ist, und dass andererseits nur genau *ein* biorthogonales System von Vektoren existiert, denn das orthogonale Komplement eines jeden Teilraumes, der von allen bis auf einen (der n) Vektoren des einen Systems aufgespannt wird ist logischerweise 1-dimensional (denn diese $n-1$ Vektoren spannen ja einen $n-1$-dimensionalen Teilraum auf!). Es ist also legitim und üblich, von *DEM* Biorthogonalsystem zu A bzw. zu sprechen. [5]

Nun kommen wir aber wieder zur Frage des Basiswechels. Es seien $[\overrightarrow{x}]_{\mathscr{A}}$ bzw. $[\overrightarrow{x}]_{\mathscr{B}}$ die Koeffizientenvektoren zu $\overrightarrow{x}$ in der Basis $\mathscr{A}$ bzw. $\mathscr{B}$. Da die EINDEUTIGKEIT der Koeffizientenzur Folge hat, dass der Zusammenhang zwischen den Elementen eines (beliebigen, endlichdimensionalen) Vektorraumes und seiner Koeffizientenfolge in $\mathbb{C}^n$ bzw. $\mathbb{R}^n$ LINEAR und bijektiv ist, [6] muss es also eine MATRIX geben, sodass der Übergang von der einen zu der anderen Basis (d.h. genauer gesagt den bijektiven und linearen Zusammenhang zwischen den entsprechenden Koeffizientenfolgen) realisiert. Wir nennen diese Matrix die Übergangsmatrix.

Wir betrachten das Problem einmal für den "Standardfall" des Vektorraumes $\mathbb{R}^n$, dann gleich in allgemeiner Form (Matrix-Darstellung beliebiger linearer Abbildungen, ohne auf alle technischen Details in dieser Vorlesung einzugehen).

Sind also die Koeffizenten eines Vektor $\overrightarrow{x} \in \mathbb{R}^n$ bzgl. $\mathscr{A}$ gegeben, wir schreiben $[x]_{\mathscr{A}}$, so erhalten wir daraus den Vektor, in dem wir $x = A * [x]_c A$ bilden. Die $\mathscr{B}$-Koeffizientenfolge diese Vektors ergeben sich somit als

$$[x]_{\mathscr{B}} = B^{-1} * x = (B^{-1} * A) * [x]_{\mathscr{A}} = C * [x]_{\mathscr{A}}, \tag{4}$$

und üblicherweise wird $C = B^{-1} * A$ als Übergangsmatrix (von den $\mathscr{A}$-Koeffizeneten zu den $\mathscr{B}$-Koeffizienten) bezeichnet. Es ist nun klar, dass einerseits aus symmetrischen Überlegungen, bzw. auch weil es die Umkehrabbildung beschreibt, deren Matrix natürlich die inverse Matrix ist, dass der Übergang von den $\mathscr{B}$-Koordinaten zu den $\mathscr{A}$-Koordinaten durch die Matrix $A^{-1} * B$ gegeben ist.

[4] Das ist einfach Kronecker's Delta-Funktion

[5] Später werden wir noch sehen, wohl als Übungsaufgabe, dass das Ergebnis des Gram-Schmidt Prozesses jeweils von der Anordnung der Vektoren abhängt. Unabhängig davon, wie die Anordnung der Vektoren davor ist, ist das letzte Element im GS-Verfahren immer der "biorthogonale Partner" zum letzen Vektor das Ausgangsbasis, mit der das GS-Verfahren gestartet wird. Man kann also (im Prinzip) das Biorthogonalsystem dadurch gewinnen, dass man mehrfach das Gram-Schmidt Verfahren laufen lässt, und jeweils *einen anderen* Basisvektor an letzter Stelle, durch geeignete Umnumerierung setzt.

[6] Wir erinnern uns, dass im Falle des *konkreten* Vektorraumes $\mathbb{R}^n$ und einer durch eine Matrix A gegebenen Basis $\mathscr{A} = [\overrightarrow{a_1}, \overrightarrow{a_2}, \ldots, \overrightarrow{a_n}]$ ist, so ist die lineare Abbildung zwischen Koeffzieten $[x]_{\mathscr{A}}$ und Vektoren x durch Matrix-Multiplikation mit A gegeben, die Umkehroperation also durch Matrix-Multiplikation (wieder von links) mit der Matrix A^{-1}.

Man kann sich die Reihenfolge leicht der Operationen (die Reihenfolge, in der die Matrizen angeschrieben wird, ist bei Links-Matrixmultiplikation - leider - genau umgekehrt!) dadurch merken, dass man feststellt, dass zuerst aus den Koeffizienten(oder Koordinaten) in der gegebenen Basis der Vektor "gebildet werden muss" (also Matrixmultiplikation mit der die Basis beschreibenden Matrix), und *dann* müssen für den so gebildeten Vektor die Koeffizientenin der *anderen* Basis bestimmt werden, und dafür ist (im Falle des Vektorraumes $\mathbb{R}^n$ und einer durch eine Matrix gegebenen Basis) die Multiplikation mit der inversen Matrix (als zweiter Schritt, und daher von links) nötig. Wir beschreiben den Sachverhalt in Vorwegnahme einer allgemeineren Terminologie mit der Gleichung: $[Id]_{\mathscr{B}\leftarrow\mathscr{A}} = B^{-1} * A$.

Der geschilderte Sachverhalt ist aber in noch viel größerer Allgemeinheit gültig. Da jede lineare Abbildung $T : v \mapsto T(v) \in W, v \in V$, zwischen zwei endlichdimensionalen Räumen V und W durch die Wahl einer Basis $\mathscr{A}$ in V und einer Basis $\mathscr{B}$ in W gleichwertig durch den (linearen) Übergang vom Koeffizenten $[v]_{\mathscr{A}}$ zu $[Tv]_{\mathscr{B}}$, also eine Matrix vom Format $m \times n$ beschrieben werden kann (wobei wir annehmen, dass V n-dimensional ist, und W m-dimensional. Die natürliche Bezeichnungsweise für diese Matrix erscheint mit das Symbol $[T]_{\mathscr{B}\leftarrow\mathscr{A}}$ zu sein, welche durch die Identät

$$[Tv]_{\mathscr{B}} = [T]_{\mathscr{B}\leftarrow\mathscr{A}} * [v]_{\mathscr{A}}. \tag{5}$$

Wir überlassen es den LeserInnen als Übungsaufgabe (! Achtung, mögliche Prüfungsfrage) nachzuweisen, dass man diese Matrix ganz einfach dadurch bestimmen kann, dass man die Koeffizenten der Bilder der Basisvektoren aus der Basis $\mathscr{A}$ im Raum W, in Bezug auf die dort vorhandene Basis $\mathscr{B}$ (was sonst? [7]) bestimmt, und die so gewonnen m-Tupel als Spalten der Matrix einträgt. [8]

Man könnte sagen (und auch genau mathematisch beschreiben), dass es sich bei dem Übergang (zu irgendwelchen) Koordinaten naürlich um eine reine *Umformung* des Problems kommt, sozusagen von einer Übersetzung von einer Sprache in eine andere, von einer *begrifflichen* Sprache in eine numerisch umsetzbare Sprache. Wir werden das zunächst anhand von dem was ich als "halb-abstrkate" Problemstellungen (mit Vektorräumen von Polynomen) beschreiben. Weiters ist anzumerken, dass es einigen Fällen so ist, dass das Problem in bestimmten Koordinaten gegeben ist (z.B. in den natürlichen Koordinaten, was immer das im jeweiligen Kontext ist, etwa in der Basis er Monome $Montr$ für den Raum $\mathcal{P}_3(\mathbb{R})$ der kubischen Polynomfunktionen auf $\mathbb{R}$, um die MATLAB Ordnung zu berücksichtigen, oder in der Basis der Einheitsvektoren $(\overrightarrow{e_k})$ im $\mathbb{R}^m$).

Um die zuletzt gemachte Aussage zu verdeutlichen, eine kleine Liste von entsprechenden Aussagen (die so selbstverständlich erscheinen, dass man sich fragt, warum man diese überhaupt formulieren muss). Man bedenke aber, dass es keine "logischen Fakten" sind, sondern dass es sich dabei im Prinzip um eine Erläuterung des Prinzips handelt, das uns nur bestätigt, dass die Matrix-Multiplikation eine wichtige begriffliche Umsetzung, sozusagen vom Alltagsproblem ein eine mathematische Sprache ist,. Weiters erlaubt uns MATLAB (oder OCTAVE oder andere math. Softwareprogramme) die in Matrix-Schreibweise beschriebenen Fragestellungen mit ein paar Zeilen von Code umzusetzen.[9]

1. Das Problem $Tv = w$ hat eine Lösung genau dann, wenn die Matrixgleichung $[T]_{\mathscr{A}\leftarrow\mathscr{B}} * [v]_{\mathscr{A}} = [w]_{\mathscr{B}}$ ein Lösung hat;

2. Eine Familie von Vektoren ist genau dann linear unabhängig, wenn die Familie ihrer Koeffizientenvektoren linear unabhängig ist. Schreiben wir diese in eine Matrix, so ist das genau dann der Fall, wenn diese Matrix einen nicht-trivialen Kern (Nullraum) hat (> ergibt konkrete Testmöglichkeit.

[7] Die übliche *Domino-Regel*, wonach Matrizen nur dann komponierbar sind, wenn sie dimensionsweise zusammenpassen, also z.B. eine $m \times n$ Matrix kann nur mit einer $n \times k$ Matrix komponiert werden, gilt auch hier.

[8] Diese Konventionen beruhen also auf dem Prinzip, dass es ausreicht, eine lineare Abbildung auf der Basis zu kennen, weil man die restlichen Vektoren durch Linearkombinationen (eindeutig) bestimmen kann.

[9] Hier gibt es einen klaren Unterschied zu Zeiten von BASIC oder PASCAL, wo die BenutzerIn mit der "Programmierung" trivialer Routinen (sagen wir Gauss-Elemination oder Fast Fourier Transform) schon die eigene Zeit vergeudette, noch bevor sie/er in die Nähe einer echten Anwendung kommen konnte.

3. die Komposition von linearen Abbildungen entspricht genau dem Matrixprodukt der entsprechenden Matrizen. Man kann sagen, dass die Matrizenrechnung *genau so gemacht ist*, dass diese Aussage gilt. Das ist also keine Geschenk des Himmels, sondern eine der grundlegenden Einsichten, die der Abstraktion zu verdanken ist, und die diverse Rechnungen in und zwischen verschiedenen konkreten endlichdimensionalen Vektorräumen auf eine einheitliche Basis stellt. MATLAB hilft dann bei der konkreten Umsetzung und erspart sozusagen das Rechnen von Hand. Der Benutzer muss nur (nun aber viel bewusster) die Umsetzung vom Anwendungskontext in die MATLAB (bzw. einfach Matrix) Beschreibung durchführen, und dann auch die Ergebnisse wieder Rückinterpretieren. Auch die Assoziativität der Matrix Multiplikation (normalerweise durch umständliches Index-Jonglieren gewonnen, ist auf diese Weise leicht einzusehen und vollkommen plausible, und nicht überraschend). Die entsprechende Formel (Kurzfassung) wäre in unsere Notation

$$[S \circ T]_{\mathscr{A} \leftarrow \mathscr{C}} \tag{6}$$

In Worten: Wendet man T ausgehend vom Vektorraum V mit der Basis $\mathscr{A}$ ..... und dann S an, so ist die Matrix der *zusammengesetzen* Abbildung ..... genau das *Matrixprodukt* der einzelnen Matrizen, wobei klarerweise die verwendeten Basen zusammenpassen müssen [10]

4. Die Abbildung T von V nach V ist invertierbar, genau dann wenn die entsprechende (quadratische) Matrix $[T]_{\mathscr{A} \leftarrow \mathscr{A}}$ $[T]_{\mathscr{A} \leftarrow \mathscr{A}}$ invertierbar ist. Natürlich wird die inverse Abbildung von der inversen Matrix bestimmt. Wieder kurz in Form einer prägnanten Formel:

$$[T \in]_{\mathscr{A} \leftarrow \mathscr{A}} = [T]^{-1}_{\mathscr{A} \leftarrow \mathscr{A}} \tag{7}$$

5. Wenn wir eine Abbildung in einer anderen Basis beschreiben wollen, müssen wir die eine Matrix mit einer Basiswechsel-Matrix "konjugieren"

$$[T]_{\mathscr{B} \leftarrow \mathscr{B}} = [Id]_{\mathscr{B} \leftarrow \mathscr{A}} * [T]_{\mathscr{A} \leftarrow \mathscr{A}} * [Id]_{\mathscr{A} \leftarrow \mathscr{B}} = C * [T]_{\mathscr{A} \leftarrow \mathscr{A}} * C^{-1}, \quad C = [Id]_{\mathscr{B} \leftarrow \mathscr{A}}. \tag{8}$$

6. Eine Abbildung ist surjektiv genau dann wenn die Spalten der zugehörigen Matrix ein Erzeugendensystem sind, also genau dann wenn $([T]_{\mathscr{A} \leftarrow \mathscr{A}})'$ trivialen Nullraum hat.

7. Man könnte die Liste noch weiter ausführen (wird wohl später noch gemacht).

Aber wie ist das mit der Berechnung von Winkeln? Gibt es auch in abstrakten (endlichdim.) Vektorräumen Winkel bzw. Skalarprodukte. Können wir im $\mathbb{R}^n$ oder $\mathbb{C}^m$ beliege Koordinaten nehmen, um Skalarprodukte zu berechnen?? Das kan ja nicht sein, den die meisten invertierbaren, linearen Abbildungen, wie z.B. die Scherungen in der Ebene $\mathbb{R}^2$ deformieren Winkel und lassen selbst orthogonale Winkel nicht intakt (sind als keine *orthogonalen* Transformationen)[11]

Um dies zu ergründen und auch gleich ein sehr wichtige! praktische Rechenregel (die eigentlich auch erklärt, warum in MATLAB und auch in der theoretischen Literatur die Matrix A' (transponiert konjugierte von A) eine viele wichtigere Rolle spielt als die transponierte Matrix, in MATLAB als $A.'$ realisiert):

$$\langle A * x, y \rangle = \langle x, A' * y \rangle \quad \forall x, y \in \mathbb{C}^n. \tag{9}$$

[10] Andernfalls wäre noch eine Basiswechselmatrix erforderlich, dann hätten wir z.B.

$$[S \circ T]_{\mathscr{A} \leftarrow \mathscr{A}} = [T]_{\mathscr{A} \leftarrow \mathscr{B}} * [Id]_{\mathscr{B} \leftarrow \mathscr{A}} * [S]_{\mathscr{A} \leftarrow \mathscr{A}}$$

[11] Hier hat der Begriff der orthogonalen Matrizen seinen Ursprung, sie sollten eigentlich "orthogonalitätserhaltende Matrizen" heissen!

Zum Beweis realisieren wir einfach beide Seiten der Gleichung in MATLAB Notation und benutzen die Assoziativität der Matrixmultiplikation:

$$y' * (A * x) = y' * (A')' * x = (A * y)' * x.$$

Als nächstes erinnern wir, dass die Matrizen U mit der Eigenschaft, dass $U^{-1} = U'$, also die mit der Eigenschaft, dass $U' * U = Id = U * U'$ erfüllen die orthogonalen, im Kontext von $\mathbb{C}^m$ treffender die "unitären" Matrizen genannt werden. Diese trifft auch besser die Analogie zum Körper der komplexen Zahlen, d.h. der 1×1-Matrizen. Dort sind die vom Absolutbetrag gleich 1 natürlich eine ausgezeichnete Teilmenge (sogar ein Untergruppe bzgl. Multiplikation!, und es gilt, dass das inverse Element $1/u = conj(u)$ ist, weil ja $conj(u) * u = abs(u)^2 = 1^2 = 1$ gilt. Ausserdem ändert sich der Absolutbetrag einer komplezen Zahl nicht, wenn sie mit einer Zahl $u \in \mathbb{U} := \{u \in \mathbb{C} \,|\, |u| = 1\}$ multipliziert wird, weil ja $|u \cdot z| = |u| \cdot |z|$ gilt.

Daraus ergibt sich die Erwartung, dass die Anwendung einer unitären $n \times n$-Matrix die Längen von Vektore bewahrt. Wir betrachten es gleich allgemeiner, und zeigen dass Skalarprodukte bewahrt bleiben.

Theorem 1. *Die unitären Matrizen haben die Eigenschaft (und sind dadurch sogar charakterisiert), dass*

$$\langle U * x, U * y \rangle = \quad \forall x, y \in \mathbb{C}^n.$$

Proof.

$$\langle U * x, U * y \rangle = \langle x, U' * U * y \rangle = \langle x, y \rangle.$$

Für die Umkehrung setzen wir die Identität $\langle x, U' * U * y \rangle = \langle x, y \rangle$ für alle $x, y \in \mathbb{C}^n$ als gültig voraus, dann gilt $y = U' * U * y$ für alle $y \in \mathbb{C}^n$, sodass daraus folgt dass $U' * U = Id_n$ gilt. Da die inverse Matrix eindeutig bestimmt ist (die Links-Inverse und die Rechtsinverse müssen gleich sein, das ist eine einfache algebraische Tatsache), muss auch $U * U' = Id_n$ gelten, d.h. $U' = U^{-1}$. □

Als Folge des obigen Argumentes können wir folgende Aussagen verbuchen:

1. Unitäre (also auch orthogonale Matrizen) bewahren Längen und Winkel
2. Unitäre Transformationen sind (genau) diejenigen, die ONBs in ONBs überführen (letztendlich kann man sagen, wenn eine lineare Abbildung die Einheitsbasis in eine ONB überführt, dann führt sie auch *jede andere* ONB in eine solche über.
3. Insbesondere gilt: Kenne ich die Koordinaten von zwei Vektoren bzgl. irgendeiner ONB kann ich deren Skalarprodukt (und daraus Winkel etc.) unmittelbar durch die übliche Form des Skalarproduktes $\langle x, y \rangle = \sum_{k=1}^{n} x_k \overline{y_k}$ bestimmt werden.

1 Pseudo-Inverse & MNLSQ Lösungen lin. Glgsysteme

Nun zur Abwechslung wieder einmal ein paar geometrische Gedanken zum Thema der linearen Algebra und linearen Gleichungssystemen.

Normalerweise werden "vor allem" lineare Gleichungssysteme betrachtet, die mit Hilfe einer invertierbaren Matrix beschrieben werden können. Man findet dann *für beliebiege rechte Seite* **b** die *eindeutig bestimmte* Lösung des Problems $A * x = b$ durch Anwenden der inversen Matrix $x = A^{-1} * b$. Aber was macht man, wenn das Gleichungssystem *inkonsistent* ist, d.h. wenn zuviele bzw. widersprüchliche Gleichungen vorliegen, oder wenn das Gleichungssystem zwar lösbar ist, aber aufgrund freier Parameter unendlich viele Lösungen möglich sind. Welche dieser Lösungen sollte man dann nehmen, gibt es eine ausgezeichnet, sozusagen "natürliche" Lösung?

Nun zur ersten Frage. Wann ist das individuelle Problem (d.h. für konkrete Matrix A und konkrete rechte Seite das Problem $A*x = b$ lösbar. Natürlich genau dann, wenn b zum Spaltenraum (= Bildraum) von A gehört (sozusagen definitionsgemäss. Was aber tun, wenn dies nicht der Fall. Die IDEE dazu ist: Man verändere das Gleichungssystem durch Verändern der rechten Seite, um die Lösbarkeit zu erreichen. Der Vektor $P_A(b)$ (Projektion auf den Spaltenraum von A sei so bezeichnet) findet man sicher denjenigen Vektor im Spaltenraum, für den gilt $\|b - P_A(b)\| \leq \|b - A * x\|$, für alle $x \in \mathbb{R}^n$. Wir schreiben (wenn der Kontext klar ist) auch $\tilde{b} := P_A(b)$. Dann ist also $\sum_{k=1}^{n} |b_k - \widetilde{b_k}|^2 = min!$, dass heisst, eine bestimmte Quadratesumme is minimal. Die *neue Gleichung*, also $A * x = \tilde{b}$, ist somit definitionsgemäß konsistent, d.h. lösbar!

Nun ergibt sich allerdings die Frage, ob es mehrere Lösungen (oder nur genau eine) gibt. Man erinnert sich (! leichte Übungsaufgabe) dass die allgemeine Lösung des inhomogenen Problems aus einer spezielle Lösung des inhomogenen Problems + alle Lösungen des homogenen Gleichungssystems $A * x = 0$ gebildet werden können. In der Tat, falls $A * x_1 = \tilde{b}$ und $A * x_1 = \tilde{b}$, dann gilt klarerweise

$$A * (x_1 - x_2) = A * x_1 - A * x_2 = \tilde{b} - \tilde{b} = 0.$$

Umgekehrt gilt mit $A * z = \tilde{b}$ und $A * n = 0$ auch $A * (z + n) = \tilde{b} + 0 = \tilde{b}$. Damit ist das Argument komplett.

Geometrisch gesprochen gibt es in jeder solchen (affinen) Lösungsmenge zwei Komponenten (gemäß "Four Spaces", nämlich der Anteil *innerhalb* des Zeilenraumes und die (dazu orthogonale) Komponente im Nullraum der Matrix. Aufgrund des Pythagoräischen Satzes ($a^2 + b^2 = c^2$) folgt daraus, dass das Element im Zeilenraum das Element minimaler Norm ist, weil es in die Richtung des Nullraumes die kleinst-mögliche Norm, ist also eine "minimal norm" Lösung.

Deshalb ist diese Lösung die MNLSQ-solution, d.h. die "Minimal Norm Least SQuares Solution". Diese Zuordnung (jeder rechten Seite b wird zuerst $\tilde{b} = P_A b$ und dann der entsprechende Punkt im Zeilenraum von A zugeordnet, wird durch die Matrix $pinv(A)$ realisiert. Zu einer $m \times n$ Matrix A ist $pinv(A)$ vom Format $n \times m$, führt also wieder von $\mathbb{R}^m$ nach $\mathbb{R}^n$ zurück.

Eine weitere Sichtweise, die sich daraus ergibt: *Jede lineare Abbildung besteht im Prinzip aus einem Isomorphismus $\tilde{A}$ zwischen dem Zeilenraum von A (eigentlich Bildraum von A') und dem Spaltenraum von A, ergänzt um die Nullräume von A bzw. A'. Die Pseudo-Inverse (auch Moore Penrose inverse Abb.) invertiert den "invertierbaren Teil" einer Matrix und stellt symmetrische Verhältnisse her.*

`Weiterer Stoff: Operatornorm, Konditionszahl einer Matrix, SVD, PINV`

2 Bestimmung glatter Ausgleichskurven aus Messdaten

In diesem Abschnitt wollen wir einige konkreten Anwendungen betreffend den Umgang mit Polynomfunktionen und Basiswechsel.

Es ist (nun) bekannt, dass man ein Polynom mit n unbekannten Koeffizienten (also vom Grad $n-1$ bzw. von der Ordnung n) aus n Abtastwerten eindeutig bestimmt werden kann.

Beispielsweise sind die folgenden Operationen *Isomorphismen* auf dem Raum $\mathcal{P}_3(\mathbb{R})$ (natürlich auch auf Räumen von Funktionen anderer Ordnung).

Beispielsweise die Translationsoperatoren: $T_z p(x) = p(x-z)$ oder der Dehnungsoperator $D_\rho p(x) = p(\rho z)$, für ein $\rho > 0$.

Dies entsprecht der Verschiebung des Graphen, bzw. der Streckung und Stauchung des Graphen. Bekanntlich ist der Raum der Polynomfunktionen invariant unter diesen Operationen, welche sowohl linear also auch invertierbar war.

Die Möglichkeit besteht unter Zuhilfenahme der Vandermonde Matrix, die den Übergang von den Koeffizienten des Raumes der Polynomfunktionen (mit der *monomialen Basis*, in MATLAB Anordnung) zu bestimmen. Man kann also mit Hilfe der Matrix `A = vander(n);` aus den Daten die Koeffizienten des Polynoms bestimmen.

Bekanntlich sind die Spalten der Invesion Matrix die Lösungen des Gleichtungssystems $A * x = \overrightarrow{e_k}$, $k = 1, \ldots, n$. Das sind aber dann genau die Koeffizienten der sogenannten Lagrange-Intepolation: $L_k(x_j) = \delta_{k,j}$ (Kronecker), d.h. dasjenige Polynom das an genau einem der angegebenen Punkte den Wert 1 annimmt, und an den anderen Null ist. Somit ist auch klar, dass das gesuchte Polynom zu dem Datenvektor $\overrightarrow{d} = (d_1, \ldots d_n)$ von der Form

$$\sum_{k=1}^{n} d_k L_k(x).$$

Wenn man fragt, wie sich kleine Ungenauigkeiten in den Daten auf das so bestimmte Polynom auswirkt, muss man die Konditionszahl der Vandermode-Matrix bestimmen (Definitionen werden noch folgen). Diese wird sehr rasch groß, wenn die Werte (Argumente) gross werden, sodass man dann besser zu einer anderen, lokalen Basis übergeht, z.B. $((x-t)^m, (x-t)^{m-1}, \ldots, (x-t)^2, (x-t), 1)$.

`wird noch wesentlich weiter ausgebaut! hgfei`

3 Affine Teilräume

Der Zusammenhang zwischen den Ebenen im $\mathbb{R}^3$ (korrekterweise den *affinene Teilräumen* von $\mathbb{R}^3$) ist leider nicht so gut verdaut wie man es sich wünschen könnte, daher die folgende kurze Auffrischung:

Erinnern wir uns daran, dass man folgende Sätze über lineare Gleichungssysteme kennt:

1. Die Lösungsmenge eines homogenen Gleichungssytems ist eine Teilraum von $\mathbb{R}^n$.

2. Die Differenz zwischen zwei Lösungen eines inhomogenen Gleichungssystems stellen natürlich eine Lösung des homogenen Systems, und umgekehrt, d.h. die allgemeine Lösung eines inhomogenen Gleichungssystem entsteht aus einer speziellen Lösung des inhomogenenen Systems durch Addition aller Lösungen des homogenen Systems.

In mehr geometrischer Form kann man dies aber auch durch den Begriffe der affinen Teilräume bezeichnen. Man kann also sagen, dass **die Lösungsmengen von linearen Gleichungssystemen genau die affinen Teilräume von $\mathbb{R}^n$ sind**.

Methoden von Projektionen auf affine Teilräume sind daraus leicht ableitbar: SKIZZE: Man mache eine Skizze in $\mathbb{R}^2$: ein affiner Teilraum (ein-dimensional) ist dann einfach eine Gerade, und es geht um die (orthogonale) Projektion eines Punktes auf diese Gerade entlang der Normalen.

Definition 1. Eine Abbildung der Form $\overrightarrow{x} \mapsto A * \overrightarrow{x} + \overrightarrow{b}$ heißt *affine Abbildung.* [12]

Übungsaufgaben:

1. Man zeige, dass jede Projektion auf einen Teilraum des $\mathbb{R}^n$ eine lineare Abbildung ist. Diese Abbildung ist orthogonal genau dann wenn die zugehörige Matrix bzgl. *irgendeiner* ONB des $\mathbb{R}^n$ selbstadjungiert ist (d.h. $A == A'$ erfüllt).

2. Man zeige, dass für je zwei Tripel von (nicht kolinearen) Punkten im $\mathbb{R}^3$ eine affine Abbildung existiert, welche die entsprechenden Punkte ineinander üeberführt. Man bestimme diese Abbildung (Rechengang, der im Prinzip in MATLAB realisierbar sein sollte). Ist diese Abbildung eindeutig gegeben? Überlegungen zur Verallgemeinerung auf allgemeine Dimensionen.

3. Man finde eine Beschreibung (ebenfalls konstruktiv, algorithmisch) für die Spiegelung an einem affinen Teilraum. Dabei wird natürlich die Zerlegung des $\mathbb{R}^n$ nach einem Teilraum und seinem orthonoalen Komplement eine Rolle spielen.

[12] Der Name "affine Abbildung ist durch die Tatsache gerechtfertigt, dass die soeben als affin bezeichneten Abbildungen genau diejenigen Abbildungen sind, die affine Teilräume in affine Teilräume abbilden.

FFT: The Fast Fourier Transform, the FFT algorithms

```
http://www.fftw.org/download.html  :  FFTW download center
http://www.gnu.org/software/octave/  : OCTAVE download center
http://de.wikipedia.org/wiki/Octave   WIKI Information about OCATVE
```

Die Fourier transform kann als lineare Abbildung erkannt werden, weil

```
 fft( A *  x) ==  fft(A)  *  x ,  fuer zufaelligen input;

F = fft(eye(n)) ist die Matrix, mit    F * x == fft(x);
```

Dann ist F eine symmetrische Matrix (d.h. die Matrix stimmt mit ihrer transponierten ueberein!), allerdings auch unitär, bis auf den Normierungsfaktor $\sqrt{n}$, daher ist die inverse (bis auf den Faktor n) gerade die konjugiert Matrix. Somit sind die Eigenschaften von FFT bzw. IFFT sehr ähnlich!

```
ifft(y) =   F' *  y / n;
```

Andererseits ist die Fourier Matrix F im Prinzip eine Vandermonde Matrix. Genauer, man kann sie als die Matrix zur Abbildung $T : p_b(x) \mapsto fft(b)$ verstehen, wobei man die (mathematisch) übliche Sortierung der Monome, als $\mathcal{B} = \{1, z, z^2, \ldots z^{n-1}\}$ als Basis für $\mathcal{P}_n(\mathbb{C})$ verwendet, und die Auswertung dieses Polynoms an den Einheitswurzeln der Ordnung n, beginnend bei $1 = \omega^0$, im Uhrzeigersinn.

In anderen Worten, die Abbildung $x \mapsto fft(x)$ kann auch so interpretiert werden: Mache aus dem Vektor eine Polynom (wobei die *formale* Länge des Vektors entscheidend ist, nicht der effektive Grad), und bilde dann davon die Wertfolge diese (komplex-wertigen) Polynoms aus $\mathcal{P}_n(\mathbb{C})$ Beachte, dass die in MATLAB verwendete Sortierung der Monome die umgekehrte Reihenfolge verwendet. Daher haben wir, dass die FFT den Werten an den $n-$ten Einheitswurzeln entspricht, bei $\omega^0 = 1$ beginnend, im Uhrzeigersinn:

```
disp('n');    urts =  exp(- 2 * pi * i * (0: n-1)/n);  b = flipud(a(:));
norm( fft(a) -   polyval(b,urts)),
bzw/ die FFT-Matrix entspricht einer Vandermondematrix,
wobei zu beachten ist, dass der "fliplr" Befehl die Reihenfolge
umdreht, weil die FFT tranditionell mit der ueblichen Reihenfolge
der Monome, waehrend "polyval" die Monome in fallenden Potenzen ordnet!
% respectively the corresponding matrices are equal:
norm( vander(exp( - 2 * pi * i * (0 : n-1)/n))  - fliplr(fft(eye(7))) ,'fro' )
ans =   3.0760e-015
```

Der Ausdruck `norm(A,'fro')` bedeutet "Frobenius-Norm", was nichts anderes als `norm(A(:))` bedeutet, also die Länge der $m \times n$-Matrix A gesehen als Element von $\mathbb{C}^{n \cdot m}$.

Normale Matrizen

Als Vorbereitung auf die Theorie der SVD (Singular Value Decomposition) und die Diagonalisierung ein paar Ideen zum Thema Blockmatrizen und Reduktion von grossen linearen Gleichungssystemen.

Definition 2. Eine quadratische Matrix A heisst *normal* wenn $A * A' == A' * A$.

BEMERKUNG: Jede symmetrische Matrix (im Sinne von reell-symmetrisch oder hermitsch, mit $A == A'$) und jede unitäre Matrix ist normal (weil $A * A' = Id = A' * A$).

Wir wollen nun zeigen, dass für diese Matrizen einige interessante Eigenschaften gelten.

Lemma 1. *Für jede normale Matrix sind Zeilen- und Spaltenräume identisch.*

Proof. Schreiben wir $Null(A$ für den Nullraum einer Matrix. Dann gilt

$$Null(A') = Null(A * A') = \text{ (by assumption) } = Null(A' * A) = Null(A).$$

Daraus folgt, dass auch deren orthogonalen Komplemente, das sind der Spaltenraum bzw. der Zeilenraum von A (eigentlich der Spaltenraum von A') gleich. □

Lemma 2. *Für eine normale Matrix sind sowohl der Spalten- als auch der Nullraum der Matrix A Teilräume von $\mathbb{R}^n$.*

Proof. Es sei $n \in Null(A)$ gegeben, d.h. $\langle n, A * x\rangle = 0$ sei die Voraussetzung. Dann gilt

$$\langle A * n, A * x\rangle = \langle n, A' * A * x\rangle = \langle n, A' * y\rangle = 0$$

for $y = A' * x$, also gilt $A * n \in Null(A)$. □

Wir werden diese Fakten dazu benutzen, um verständlich zu machen, warum gerade die "normalen Matrizen" diejenigen sind, die diagonalisierbar sind. Dass eine diagonalisierbare Matrix jedenfalls normal sein *muss* ist ja leicht einzusehen, denn wenn $U * D * U' = A$ gilt, ist die Normalität leicht zu verifizieren!

Lemma 3. *Es sei W ein A-invarianter Teilraum, dann ist auch das orthogonale Komplement $W^\perp$ von W invariant unter A, d.h. die Matrix-Darstellung der Abbildung kann auf zwei kleinere Teilblock-Matrizen reduziert werden.*

Proof. Es gelte $A * W \subseteq W$, und es sei $n \in W^\perp$. Dann gilt für jedes $w \in W$:

$$\langle A * n, w\rangle = \langle n, A' * w\rangle = \langle n, A * w_1\rangle = \langle n, w_2\rangle = 0.$$

Dabei verwenden wir, dass das Element $A' * w$ des Zeilenraumes (Spaltenraumes von A') auch ein Element des Spaltenraumes von A ist, und in der letzten Gleichung dass $A * w_1 = w_2 \in W$ gilt. □

EINIGE AUSFührungen zum Thema Wahrscheinlichkeit und Polynomen

Wenn wir ein einfaches Experiment, das Würfeln mit einem fairen Würfel betrachten, dann ist bekanntlich jede Augenzahl, zwischen 1 und 6 gleich wahrscheinlich, mit Wahrscheinlichkeit $= 1/6 = 16,67\%$. Das bedeutet praktisch, dass mein beim Würfeln nach 600 Wiederholungen mit ca. 100 Einsern, Zweiern, etc rechnen kann. Die prozentmässige Abweichung wird mit zunehmender Zahl der Wiederholungen auch geringer (sagen wir bei 6 Millionen Wiederholungen ist sie schon sehr gering). Daher ist die ërwartetePunktezahl (der Erwartungswert dieser zufaelligen Variablen) einfach $100 + 200 + ... + 600/600 = 3.5$. Würfelt man nun mit zwei Würfeln, so hat man insgesamt 36 gleich wahrscheinliche Ereignisse. Nach "Hausverstand" (der in einem mathematische Definition gefasst wird), ist die Augenzahl beim Würfeln mit 2 Würfeln *unabhängig*, d.h. ob ich zweimal hintereinander würfle oder mit einem blauen und einem roten Würfel gleichzeitig würfle, die Wahrscheinlichkeit, eines bestimmten Paares, sagen wir $RotW = 3, BlauW = 5$ ist natürlich $1/6 \cdot 1/6 = 1/36$. Anders gesprochen, es gibt 36 gleichwahrscheinliche Fälle von Paaren. Die Wahrscheinlichkeit einer bestimmten *Summe* der *Gesamtaugenzahl* hängt also davon ab, wieviele der Paare günstig sind (sagen für die Realisierung der Zahl 5). Die Antwort ist (f.d. konkreten Fall): Die Paare $(1, 4), (2, 3), (3, 2), (4, 1)$ sind guenstig, d.h. die Wahrscheinlichkeit die Summe 5 zu bekommen, ist $4/36 = 1/9$.

Hat man eine groessere Zahl von Wiederholungen, wird die Zahl der (gleichwahrscheinlichen) Möglichkeiten jeweils um den Faktor 6 größer, d.h. bei 12 Würfeln gibt es schon 6^12 Kombinationen (der rote,blaue, grüne, gelbe... Würfel...), d.i. 2176782336. Die kombinatorischen Probleme nehmen der rapide zu.

Man kann nun leicht (auf den ersten Blick erscheint dies aus der Luft gegriffen, das sollte aber kein Problem sein) eine *it isomorphe Struktur* beim Ausmultiplizieren von Polynomen finden. Multipliziert man zwei Polynome miteinander, sagen wir 2 allgemeine kubische Polynome, dann bekommt man zuerst ebenfalls 4^2 Terme, die dann sortiert werden nach Potenzen, sodass das Produktpolynom $r(x) := p(x) \cdot q(x)$ Koeffizienten hat, die nach der Cauchy'schen Produktregel zu bestimmen sind.

$$c_k = \sum_{j+l=k} a_j b_l = \sum_{j=0}^{k} a_j b_{k-j} = \sum_{l=0}^{k} a_{k-l} b_l. \tag{10}$$

Dieser *ISOMORPHISMUS* (von Problemen) kann nun genutzt werden, um das kombinatorische Problem mittels FFT (Interpretation als Übergang von Koeffizienten zu Werten des Polynoms über den Einheitswurzeln) zu realisieren. Beispielsweise kann die Verteilung von Werten, die beim Würfeln mit 47 Würfeln (oder 47-mal Würfeln) vorherrt. Dazu muss man "nur" eine enstprechende Potenz des Polynoms $p(x) = (x + x^2 + \cdots + x^6)/6$ bilden, und das geht gut mit Hilfe der FFT.