Line fitting

problem: fit points $d_1, \ldots, d_N \in \mathbb{R}^2$ by a line

1. find condition for existence of a line (any line in $\mathbb{R}^2$) that passes through the points

2. how would you test the condition in MATLAB?

3. implement a method for exact line fitting

Solution for part 1

the points $d_i = (a_i, b_i)$, $i = 1, \ldots, N$ lie on line

$$\Updownarrow$$

there is $(R_1, R_2, R_3) \neq 0$, such that
$R_1 a_i + R_2 b_i + R_3 = 0$, for $i = 1, \ldots, N$

$$\Updownarrow$$

there is $(R_1, R_2, R_3) \neq 0$, such that

$$\begin{bmatrix} R_1 & R_2 & R_3 \end{bmatrix} \begin{bmatrix} a_1 & \cdots & a_N \\ b_1 & \cdots & b_N \\ 1 & \cdots & 1 \end{bmatrix} = 0$$

$$\Updownarrow$$

$$\operatorname{rank}\left(\begin{bmatrix} a_1 & \cdots & a_N \\ b_1 & \cdots & b_N \\ 1 & \cdots & 1 \end{bmatrix} \right) \leq 2$$

Solution for part 2

given matrix `d` which columns are the data points

exact fitting condition:

```
N    = size(d, 2)
dext = [d; ones(1, N)];

if (rank(dext) < 3)
    disp('exact fit exists')
else
    disp('exact fit does not exist')
end
```

Solution for part 3

given matrix `d` which columns are the data points

exact fitting method:

```
N    = size(d, 2)
dext = [d; ones(1, N)];

r = null(dext')';
```

Note

$\mathscr{B} = \{\, d \mid Rd = 0 \,\}$ — linear static model

$\mathscr{B} = \{\, d \mid R \left[\begin{smallmatrix} d \\ 1 \end{smallmatrix}\right] = 0 \,\}$ — affine static model

in exact modeling

$$\begin{gathered}\text{affine fitting} \\ \Updownarrow \\ \text{data centering + linear modeling}\end{gathered}$$

homework: is the same true in approximate modeling?

Conic section fitting

problem: fit points $d_1, \dots, d_N \in \mathbb{R}^2$ by conic section

$$\mathscr{B}(S,u,v) = \{\, d \in \mathbb{R}^2 \mid d^\top S d + u^\top d + v = 0 \,\}$$

1. find condition for existence of an exact fit
2. propose numerical method for exact fitting
3. implement the method and test it on the data

$$d_1 = \begin{bmatrix} -1 \\ -1 \end{bmatrix}, \quad d_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \quad d_3 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad d_4 = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

Solution for part 1

the points $d_i = (a_i, b_i)$, $i = 1, \ldots, N$ lie on conic section

$$\Updownarrow$$

$\exists\ S = S^\top$, u, v, at least one of them nonzero, such that
$d_i^\top S d_i + u^\top d_i + v = 0$, for $i = 1, \ldots, N$

$$\Updownarrow$$

there is $(s_{11}, s_{12}, s_{22}, u_1, u_2, v) \neq 0$, such that

$$\begin{bmatrix} s_{11} & 2s_{12} & u_1 & s_{22} & u_2 & v \end{bmatrix} \begin{bmatrix} a_1^2 & \cdots & a_N^2 \\ a_1 b_1 & \cdots & a_N b_N \\ a_1 & \cdots & a_N \\ b_1^2 & \cdots & b_N^2 \\ b_1 & \cdots & b_N \\ 1 & \cdots & 1 \end{bmatrix} = 0$$

Solution for part 1 (continued)

the points $d_i = (a_i, b_i)$, $i = 1, \ldots, N$ lie on conic section

$$\Updownarrow$$

$$\operatorname{rank}\left(\begin{bmatrix} a_1^2 & \cdots & a_N^2 \\ a_1 b_1 & \cdots & a_N b_N \\ a_1 & \cdots & a_N \\ b_1^2 & \cdots & b_N^2 \\ b_1 & \cdots & b_N \\ 1 & \cdots & 1 \end{bmatrix}\right) \leq 5$$

```
f = @(a, b) [a .^ 2; a .* b; a; b .^ 2; b;
             ones(size(a))];
```

Solution for part 2 and 3

finding exact models

```
R = null(f(d(1, :), d(2, :))')';
```

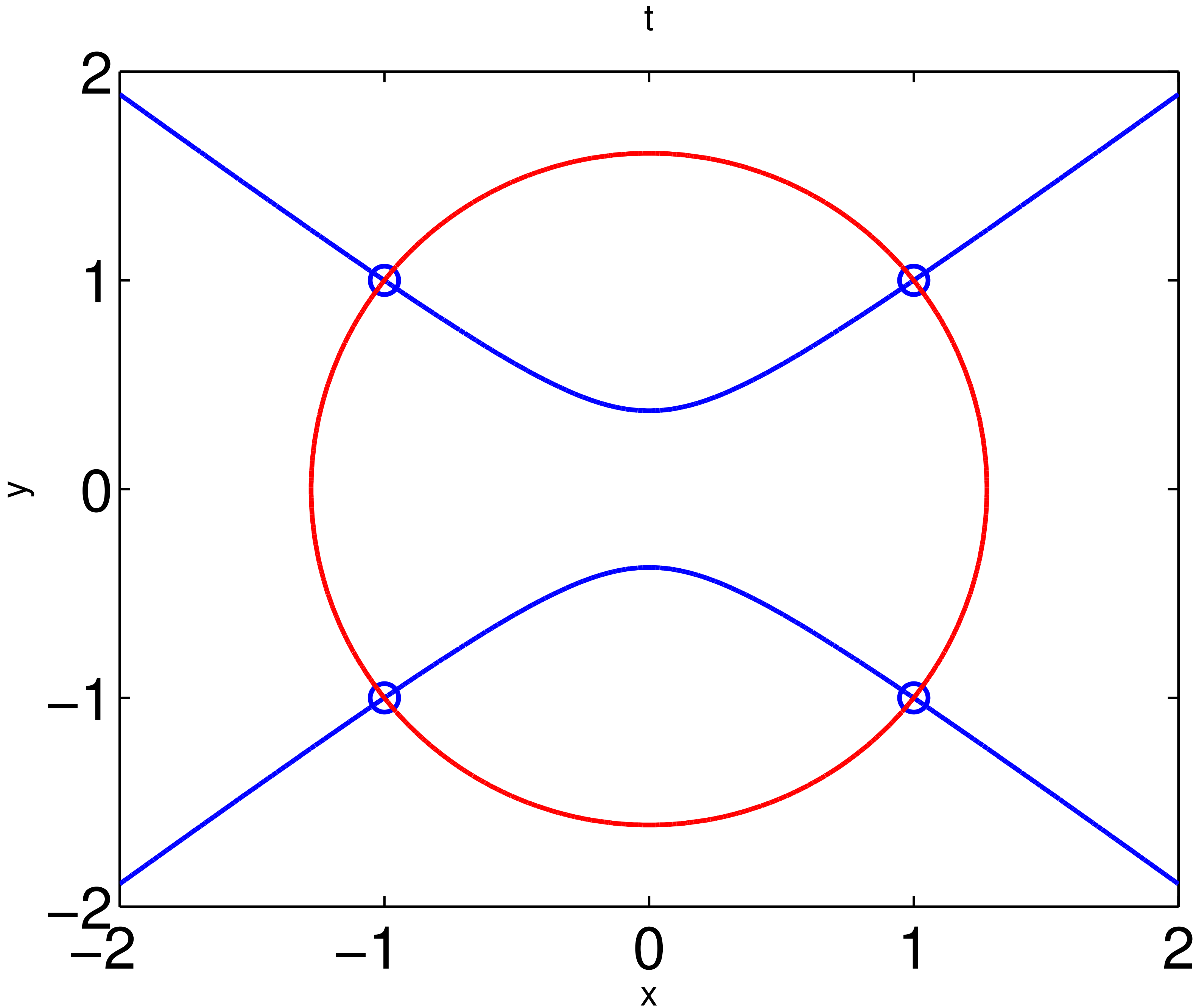
plotting model

```
function H = plot_model(th, f, ax, c)
H = ezplot(@(a, b) th * f(a, b), ax);
for h = H', set(h, 'color', c, 'linewidth', 2); e
```

show results

```
plot(d(1, :), d(2, :), 'o', 'markersize', 12)
ax = 2 * axis;
for i = 1:size(R, 1)
  hold on, plot_model(R(i, :), f, ax, c(i));
end
```

Solution for part 2 and 3 (continued)



Recursive sequence fitting

problem: fit $w = \big(w(1),\dots,w(T)\big)$ by model

$$\mathscr{B} = \{\, w \mid R_0 w + R_1 \sigma w + \cdots + R_\ell \sigma^\ell w = 0 \,\}$$

1. find condition for existence of an exact fit
   first, with, and then, without knowledge of ℓ

2. propose numerical method for exact fitting
   find the smallest ℓ, for which exact model exists

3. implement the method and test it on the data

$$(1, 2, 4, 7, 13, 24, 44, 81)$$

Solution for part 1

$$w = \big(w(1), \dots, w(T)\big) \in \mathscr{B}$$
$$\mathscr{B} = \{\, w \mid R_0 w + R_1 \sigma w + \cdots + R_\ell \sigma^\ell w = 0 \,\}$$

$$\Updownarrow$$

$$R_0 w(t) + R_1 w(t+1) + \cdots + R_\ell w(t+\ell) = 0$$
$$\text{for } t = 1, \dots, T-\ell$$

$$\Updownarrow$$

$$\operatorname{rank}\big(\mathscr{H}_{\ell+1}(w)\big) \leq \ell$$

$$\mathscr{H}_{\ell+1}(w) := \begin{bmatrix} w(1) & w(2) & \cdots & w(T-\ell) \\ w(2) & w(3) & \cdots & w(T-\ell+1) \\ \vdots & \vdots & & \vdots \\ w(\ell+1) & w(\ell+2) & \cdots & w(T) \end{bmatrix}$$

relation at time $t = 1$

$$R_0 w(1) + R_1 w(2) + \cdots + R_\ell w(\ell + 1) = 0$$

in matrix form:

$$\begin{bmatrix} R_0 & R_1 & \cdots & R_\ell \end{bmatrix} \begin{bmatrix} w(1) \\ w(2) \\ \vdots \\ w(\ell+1) \end{bmatrix} = 0$$

relation at time $t = 2$

$$R_0 w(2) + R_1 w(3) + \cdots + R_\ell w(\ell + 2) = 0$$

in matrix form:

$$\begin{bmatrix} R_0 & R_1 & \cdots & R_\ell \end{bmatrix} \begin{bmatrix} w(2) \\ w(3) \\ \vdots \\ w(\ell + 2) \end{bmatrix} = 0$$

relation at time $t = T - \ell$

$$R_0 w(T-\ell) + R_1 w(T-\ell+1) + \cdots + R_\ell w(T) = 0$$

in matrix form:

$$\begin{bmatrix} R_0 & R_1 & \cdots & R_\ell \end{bmatrix} \begin{bmatrix} w(T-\ell) \\ w(T-\ell+1) \\ w(T-\ell+2) \\ \vdots \\ w(T) \end{bmatrix} = 0$$

Solution for part 2 and 3

with ℓ unknown, do the test for $\ell = 1, 2, \ldots$

algorithm

```
for ell = 1:ell_max
   if (rank(H(w, ell + 1)) == ell)
      break
   end
end
```

in the example, $\ell = 3$ and $R = \begin{bmatrix} 1 & 1 & 1 & -1 \end{bmatrix}$

Checking whether a sequence is trajectory

1. given sequence w and polynomial R, propose method for checking numerically whether $w \in \mathscr{B} = \ker\big(R(\sigma)\big)$

2. implement it in a function `w_in_ker(w, r)`

3. test it on the trajectory

$$w = (u_d, y_d) = \big((0,1),(0,1),(0,1),(0,1)\big)$$

and the system

$$\mathscr{B} = \ker\big(R(\sigma)\big), \qquad R(z) = \begin{bmatrix} 1 & -1 \end{bmatrix} + \begin{bmatrix} -1 & 1 \end{bmatrix} z$$

Solution for part 1

$$\mathrm{w}\in \ker\big(R(\sigma)\big)$$

$$\iff R(\sigma)w = 0$$

$$\iff R_0 w(t) + R_1 w(t+1) + \cdots + R_\ell w(t+\ell) = 0$$

$$\text{for } t = 1, \ldots, T-\ell$$

$$\iff \underbrace{\begin{bmatrix} R_0 & R_1 & \cdots & R_\ell & & & \\ & R_0 & R_1 & \cdots & R_\ell & & \\ & & \ddots & \ddots & & \ddots & \\ & & & R_0 & R_1 & \cdots & R_\ell \end{bmatrix}}_{\mathscr{M}_T(R)\in\mathbb{R}^{\mathtt{p}(T-\ell)\times qT}} \underbrace{\begin{bmatrix} w(1) \\ w(2) \\ \vdots \\ w(T) \end{bmatrix}}_{\mathrm{vec}(w)} = 0$$

numerical test: $\|\mathscr{M}_T(R)\,\mathrm{vec}(w)\| < \varepsilon$ (with tolerance ε)

Another solution for part 1

w$\in \ker\big(R(\sigma)\big)$

$$\begin{aligned} \iff \quad & \mathscr{M}_T(R)\,\mathrm{vec}(w) = 0 \\ \iff \quad & R\mathscr{H}_{\ell+1}(w) = 0 \end{aligned}$$

where

$$\underbrace{\begin{bmatrix} R_0 & R_1 & \cdots & R_\ell \end{bmatrix}}_{R} \underbrace{\begin{bmatrix} w(1) & w(2) & \cdots & w(T-\ell) \\ w(2) & w(3) & \cdots & \\ \vdots & \vdots & & \vdots \\ w(\ell+1) & w(\ell+2) & \cdots & w(T) \end{bmatrix}}_{\mathscr{H}_{\ell+1}(w) \in \mathbb{R}^{q(\ell+1)\times(T-\ell)}} = 0$$

numerical test: $\|R\mathscr{H}_{\ell+1}(w)\| < \varepsilon$

Solution for part 2

```
function a = w_in_ker(w, r, ell)
a = norm(r * blkhank(w, ell + 1)) < 1e-8;
```

block-Hankel matrix $\mathscr{H}_L(w)$ constructor

```
function H = blkhank(w, i, j)
[q, T] = size(w);
if T < q, w = w'; [q, T] = size(w); end
if nargin < 3, j = T - i + 1; end
H = zeros(i * q, j);
for ii = 1:i
  H(((ii - 1) * q + 1):(ii * q), :) ...
                    = w(:, ii:(ii + j - 1));
end
```

Solution for part 2 (continued)

```
w = [0 0 0 0; 1 1 1 1];
r = [1 -1 -1 1]; ell = 1;
w_in_ker(w, r, 1)
```

homework

use image representation to check

$$w \overset{?}{\in} \text{image}\left(P(\sigma)\right) \qquad (\texttt{w\_in\_im})$$

use state space representation to check

$$w \overset{?}{\in} \mathscr{B}(A, B, C, D) \qquad (\texttt{w\_in\_ss})$$

Transfer function $\mapsto$ kernel representation

1. what model $\mathscr{B}_{\text{tf}}(H)$ is specified by transfer function

$$H(z) = \frac{q(z)}{p(z)} = \frac{q_0 + q_1 z^1 + \cdots + q_\ell z^\ell}{p_0 + p_1 z^1 + \cdots + p_\ell z^\ell}$$

2. find R, such that

$$\mathscr{B}_{\text{tf}}(H) = \ker(R)$$

3. write function `tf2r` converting H (`tf` object) to R and function `r2tf` converting R to H

Solution for part 1 and 2

the transfer function H represents model

$$\mathscr{B}_{\mathrm{tf}}(H) = \{\, w = \left[\begin{smallmatrix} u \\ y \end{smallmatrix}\right] \mid p(\sigma)y = q(\sigma)u \,\}$$

the corresponding kernel representation is

$$\underbrace{\begin{bmatrix} q(\sigma) & -p(\sigma) \end{bmatrix}}_{R(\sigma)} \begin{bmatrix} u \\ y \end{bmatrix} = 0$$

note: $y = \mathscr{Z}^{-1}\big(H\mathscr{Z}(u)\big)$ assumes zero initial conditions

homework: include initial conditions in $y = \mathscr{Z}^{-1}\big(H\mathscr{Z}(u)\big)$

Solution for part 3

```
function r = tf2r(H)
[Q P] = tfdata(tf(H), 'v');
R = vec(fliplr([Q; -P]))';

function H = r2tf(R)
Q =   fliplr(R(1:2:end));
P = - fliplr(R(2:2:end));
H = tf(Q, P, -1);
```

note: MATLAB uses descending order of coefficients