- Least Square fitting

$$\begin{bmatrix} p(t_1) \\ \vdots \\ p(t_m) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & t_1 & t_1^2 & \cdots & t_1^n \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & t_m & t_m^2 & \cdots & t_m^n \end{bmatrix}}_{\text{Vandermonde matrix}} \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix}$$

full-rank if $\begin{cases} t_i \neq t_j & \forall\ i \neq j \\ m \geq n \end{cases}$

if A is full rank. $Aa = 0$
then $a = 0$
if n- polynomial is 0 at m points
the the polynomial is 0

- Growing Set of Regressor

the family of regression problem

$$\min_x \ \left\| \sum_i^p a_i x_i - y \right\|_2^2$$

a_i's are ordered

$$\begin{bmatrix} | & | & & | \\ a_1 & a_2 & \cdots & a_n \\ | & | & & | \end{bmatrix}$$

for $p = 1 \cdots n$

As p increases, get better fit

$$x_p = R_p^{-1} Q_p^T y$$

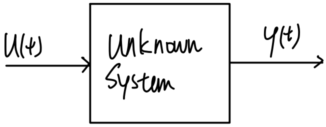
if we have a loop

for $p = 1 \cdots n$
    x_p = least_square (A[:, :p], y)

we can compute the QR of A[:, :p] from A[:, :p-1]

- Least-Square System Identification

measure input $u(t)$ and output $y(t)$ for $t=0,\dots,N$ of unknown system



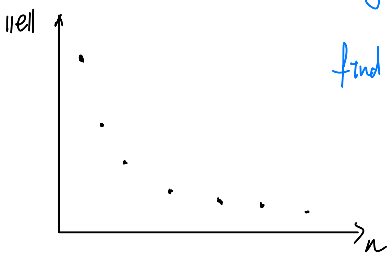
e.g. moving average model with n delays

$$\hat{y}(t) = h_0 \cdot u(t) + h_1 \cdot u(t-1) + \cdots + h_n\, u(t-n)$$

$$\begin{bmatrix} \hat{y}(n) \\ \hat{y}(n+1) \\ \vdots \\ \hat{y}(N) \end{bmatrix} = \underbrace{\begin{bmatrix} u(n) & u(n-1) & \cdots & u(0) \\ u(n+1) & u(n) & \cdots & u(1) \\ & & & \\ u(N) & u(N-1) & \cdots & u(N-n) \end{bmatrix}}_{n} \begin{bmatrix} h_0 \\ h_1 \\ \vdots \\ h_n \end{bmatrix}$$

prediction error $\begin{bmatrix} y(n) - \hat{y}(n) \\ \vdots \\ y(N) - \hat{y}(N) \end{bmatrix}$

as we increase n, the error goes down



find the elbow point

- Growing set of measurements

$$\min_x \sum_i (a_i^T x - y_i)^2$$

$$x = \left(\sum_i a_i a_i^T\right)^{-1} \sum_i y_i a_i$$

$$A \begin{bmatrix} -a_i^T- \\ \\ \\ \end{bmatrix} \begin{bmatrix} x \end{bmatrix} - \begin{bmatrix} \\ y \\ \\ \end{bmatrix}$$

grow the # of measurements

only make sense when A is full-rank

- Recursive Least-Square (simple version of Kalman filter)

we can compute $x_{ls}(m) = \left(\sum_{i=1}^{m} a_i a_i^T\right)^{-1} \sum_{i=1}^{m} y_i a_i$ $\qquad A = \begin{bmatrix} -a_i^T- \\ \\ \end{bmatrix}$

(Least square using first m measurements)

initialize $P(0) = 0 \in \mathbb{R}^{n\times n}$ $\quad q(0) = 0 \in \mathbb{R}^n$

for $m = 0, 1 \cdots$:

$$P(m+1) = P(m) + a_{m+1} a_{m+1}^T$$

$$q(m+1) = q(m) + y_{m+1} a_{m+1}$$

if P is invertible, $x_{ls}(m) = P(m)^{-1} q(m)$

$a_1 \cdots a_m$ spans $\mathbb{R}^n$

so once P is invertible, it stays invertible

- Fast Rank-1 update

can compute $P(m+1)^{-1} = \left[P(m) + a_{m+1} a_{m+1}^T\right]^{-1}$ efficiently from $P(m)^{-1}$

$$(P + aa^T)^{-1} = P^{-1} - \frac{1}{1 + a^T P^{-1} a} \underbrace{(P^{-1}a)(P^{-1}a)^T}_{\text{rank-1}}$$ "rank-1 downdate"

valid when P and $P + aa^T$ are invertible

- Multi-objective Least-Square

want $J_1 = \|Ax - y\|_2^2$ and $J_2 = \|Fx - g\|_2^2$ both be small

(eg. $F = I$, $g = 0$ for norm regularization)