Lecture slides for

Introduction to Applied Linear Algebra: Vectors, Matrices, and Least Squares

Developed by Stephen Boyd   Lieven Vandenberghe
Modified by John Duchi

17. Constrained least squares applications

Outline

Portfolio optimization

Linear quadratic control

Linear quadratic state estimation

Portfolio allocation weights

- we invest a total of V dollars in n different *assets* (stocks, bonds, $\ldots$) over some period (one day, week, month, $\ldots$)
- can include *short positions*, assets you borrow and sell at the beginning, but must return to the borrower at the end of the period
- *portfolio allocation weight vector* w gives the fraction of our total portfolio value held in each asset
- Vw_j is the dollar value of asset j you hold
- $\mathbf{1}^T w = 1$, with negative w_i meaning a short position
- $w = (-0.2, 0.0, 1.2)$ means we take a short position of $0.2V$ in asset 1, don't hold any of asset 2, and hold $1.2V$ in asset 3

Leverage, long-only portfolios, and cash

- *leverage* is $L = |w_1| + \cdots + |w_n|$
  ($(L-1)/2$ is also sometimes used)
- $L = 1$ when all weights are nonnegative ('long only portfolio')
- $w = \mathbf{1}/n$ is called the *uniform portfolio*

- we often assume asset n is 'risk-free' (or cash or T-bills)
- so $w = e_n$ means the portfolio is all cash

Return over a period

- $\tilde{r}_j$ is the *return* of asset j over the period
- $\tilde{r}_j$ is the fractional increase in price or value (decrease if negative)
- often expressed as a percentage, like $+1.1\%$ or -2.3%
- full *portfolio return* is

$$\frac{V^+ - V}{V} = \tilde{r}^T w$$

  where V^+ is the portfolio value at the end of the period
- if you hold portfolio for t periods with returns $r_1, \ldots, r_t$ value is

$$V_{t+1} = V_1(1 + r_1)(1 + r_2) \cdots (1 + r_t)$$

- portfolio value versus time traditionally plotted using $V_1 = \$10000$

Return matrix

- hold portfolio with weights w over T periods
- define $T \times n$ (asset) *return matrix*, with R_{tj} the return of asset j in period t
- row t of R is $\tilde{r}_t^T$, where $\tilde{r}_t$ is the asset return vector over period t
- column j of R is time series of asset j returns
- portfolio returns vector (time series) is T-vector $r = Rw$
- if last asset is risk-free, the last column of R is $\mu^{\mathrm{rf}}\mathbf{1}$, where μ^{rf} is the risk-free per-period interest rate

Portfolio return and risk

- r is time series (vector) of portfolio returns
- *average return* or just *return* is $\mathbf{avg}(r)$
- *risk* is $\mathbf{std}(r)$
- these are the per-period return and risk

- for small per-period returns we have

$$\begin{aligned} V_{T+1} &= V_1(1+r_1)\cdots(1+r_T) \\ &\approx V_1 + V_1(r_1+\cdots+r_T) \\ &= V_1 + T\,\mathbf{avg}(r)V_1 \end{aligned}$$

- so return approximates the average per-period increase in portfolio value

Annualized return and risk

- mean return and risk are often expressed in *annualized form* (*i.e.*, per year)
- if there are P trading periods per year

$$\text{annualized return} = P\,\mathbf{avg}(r), \qquad \text{annualized risk} = \sqrt{P}\,\mathbf{std}(r)$$

  (the squareroot in risk annualization comes from the assumption that the fluctuations in return around the mean are independent)

- if returns are daily, with 250 trading days in a year

$$\text{annualized return} = 250\,\mathbf{avg}(r), \qquad \text{annualized risk} = \sqrt{250}\,\mathbf{std}(r)$$

Portfolio optimization

- how should we choose the portfolio weight vector w?
- we want high (mean) portfolio return, low portfolio risk

- we know past *realized asset returns* but not future ones
- we will choose w that would have worked well on past returns
- … and hope it will work well going forward (just like data fitting)

Portfolio optimization

$$\begin{array}{ll} \text{minimize} & \mathbf{std}(Rw)^2 = (1/T)\|Rw - \rho\mathbf{1}\|^2 \\ \text{subject to} & \mathbf{1}^T w = 1 \\ & \mathbf{avg}(Rw) = \rho \end{array}$$

- w is the weight vector we seek
- R is the returns matrix for *past returns*
- Rw is the (past) portfolio return time series
- require mean (past) return ρ
- we minimize risk for specified value of return
- solutions w are *Pareto optimal*

- we are really asking what *would have been* the best constant allocation, had we known future returns

Portfolio optimization via constrained least squares

$$\begin{array}{ll} \text{minimize} & \|Rw - \rho\mathbf{1}\|^2 \\ \text{subject to} & \left[\begin{array}{c} \mathbf{1}^T \\ \mu^T \end{array}\right] w = \left[\begin{array}{c} 1 \\ \rho \end{array}\right] \end{array}$$

- $\mu = R^T\mathbf{1}/T$ is n-vector of (past) asset returns
- ρ is required (past) portfolio return
- an equality constrained least squares problem, with solution

$$\left[\begin{array}{c} w \\ z_1 \\ z_2 \end{array}\right] = \left[\begin{array}{ccc} 2R^TR & \mathbf{1} & \mu \\ \mathbf{1}^T & 0 & 0 \\ \mu^T & 0 & 0 \end{array}\right]^{-1} \left[\begin{array}{c} 2\rho T\mu \\ 1 \\ \rho \end{array}\right]$$

Optimal portfolios

- perform significantly better than individual assets
- risk-return curve forms a straight line
- one end of the line is the risk-free asset
- *two-fund theorem:* optimal portfolio w is an affine function of ρ

$$\begin{bmatrix} w \\ z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} 2R^TR & \mathbf{1} & \mu \\ \mathbf{1}^T & 0 & 0 \\ \mu^T & 0 & 0 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} + \rho \begin{bmatrix} 2R^TR & \mathbf{1} & \mu \\ \mathbf{1}^T & 0 & 0 \\ \mu^T & 0 & 0 \end{bmatrix}^{-1} \begin{bmatrix} 2T\mu \\ 0 \\ 1 \end{bmatrix}$$

The big assumption

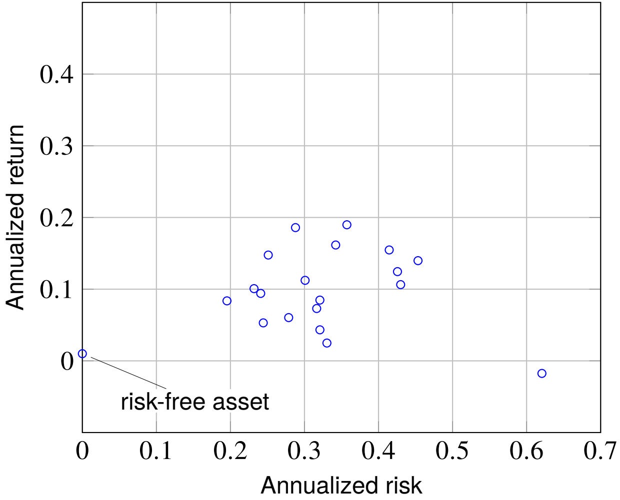
- now we make the big assumption (BA):

  FUTURE RETURNS WILL LOOK SOMETHING LIKE PAST ONES

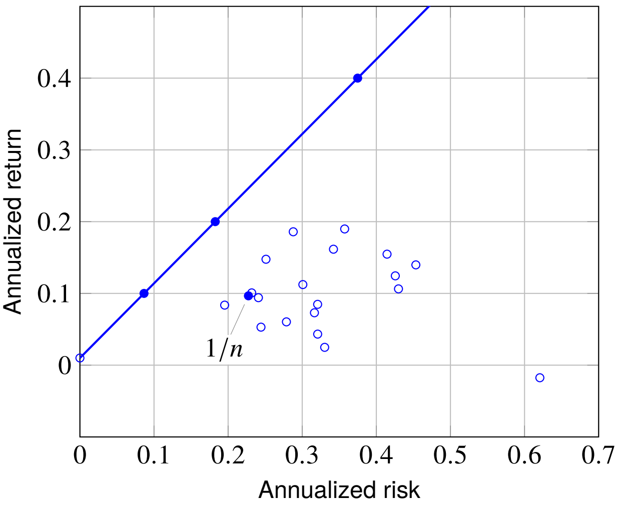
  - you are warned this need not hold, every time you invest
  - it is often reasonably true
  - in periods of ‘market shift’ it’s much less true
- if BA holds (even approximately), then a good weight vector for past (realized) returns should be good for future (unknown) returns
- for example:
  - choose w based on last 2 years of returns
  - then use w for next 6 months

Example

20 assets over 2000 days



Pareto optimal portfolios

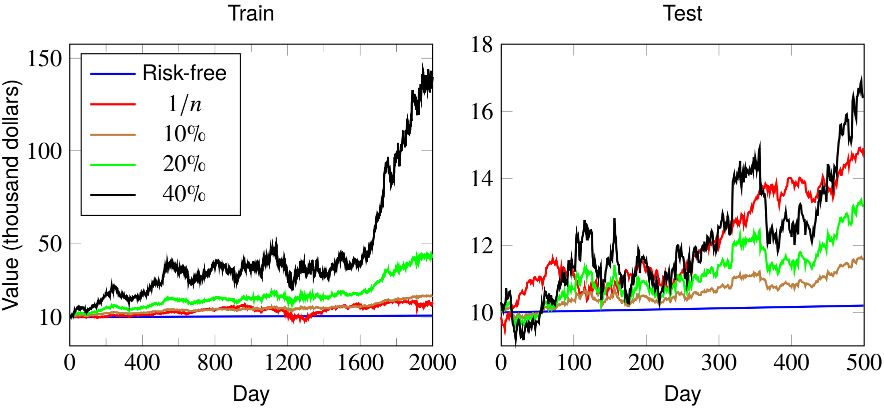


Five portfolios

	Return		Risk		
Portfolio	Train	Test	Train	Test	Leverage
risk-free	0.01	0.01	0.00	0.00	1.00
$\rho = 10\%$	0.10	0.08	0.09	0.07	1.96
$\rho = 20\%$	0.20	0.15	0.18	0.15	3.03
$\rho = 40\%$	0.40	0.30	0.38	0.31	5.48
$1/n$ (uniform weights)	0.10	0.21	0.23	0.13	1.00

- train period of 2000 days used to compute optimal portfolio
- test period is different 500-day period

Total portfolio value



Outline

Portfolio optimization

Linear quadratic control

Linear quadratic state estimation

Linear dynamical system

$$x_{t+1} = A_t x_t + B_t u_t, \quad y_t = C_t x_t, \quad t = 1, 2, \ldots$$

- n-vector x_t is *state* at time t
- m-vector u_t is *input* at time t
- p-vector y_t is *output* at time t
- $n \times n$ matrix A_t is *dynamics matrix*
- $n \times m$ matrix B_t is *input matrix*
- $p \times n$ matrix C_t is *output matrix*
- x_t, u_t, y_t often represent deviations from a standard operating condition

Linear quadratic control

$$
\begin{array}{ll}
\text{minimize} & J_{\text{output}} + \rho J_{\text{input}} \\
\text{subject to} & x_{t+1} = A_t x_t + B_t u_t, \quad t = 1, \ldots, T-1 \\
& x_1 = x^{\text{init}}, \quad x_T = x^{\text{des}}
\end{array}
$$

- variables are state sequence $x_1, \ldots, x_T$ and input sequence $u_1, \ldots, u_{T-1}$
- two objectives are quadratic functions of state and input sequences:

$$
\begin{aligned}
J_{\text{output}} &= \|y_1\|^2 + \cdots + \|y_T\|^2 = \|C_1 x_1\|^2 + \cdots + \|C_T x_T\|^2 \\
J_{\text{input}} &= \|u_1\|^2 + \cdots + \|u_{T-1}\|^2
\end{aligned}
$$

- first constraint imposes the linear dynamics equations
- second set of constraints specifies the initial and final state
- ρ is positive parameter used to trade off the two objectives

Constrained least squares formulation

$$\begin{array}{ll} \text{minimize} & \|C_1x_1\|^2 + \cdots + \|C_Tx_T\|^2 + \rho\|u_1\|^2 + \cdots + \rho\|u_{T-1}\|^2 \\ \text{subject to} & x_{t+1} = A_tx_t + B_tu_t, \quad t = 1, \ldots, T-1 \\ & x_1 = x^{\text{init}}, \quad x_T = x^{\text{des}} \end{array}$$

- can be written as

$$\begin{array}{ll} \text{minimize} & \|\tilde{A}z - \tilde{b}\|^2 \\ \text{subject to} & \tilde{C}z = \tilde{d} \end{array}$$

- vector z contains the $Tn + (T-1)m$ variables:

$$z = (x_1, \ldots, x_T, u_1, \ldots, u_{T-1})$$

Constrained least squares formulation

$$\tilde{A} = \left[\begin{array}{ccc|ccc} C_1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & C_T & 0 & \cdots & 0 \\ \hline 0 & \cdots & 0 & \sqrt{\rho} I & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & \sqrt{\rho} I \end{array}\right], \qquad \tilde{b} = 0$$

$$\tilde{C} = \left[\begin{array}{cccccc|cccc} A_1 & -I & 0 & \cdots & 0 & 0 & B_1 & 0 & \cdots & 0 \\ 0 & A_2 & -I & \cdots & 0 & 0 & 0 & B_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & A_{T-1} & -I & 0 & 0 & \cdots & B_{T-1} \\ \hline I & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 & I & 0 & 0 & \cdots & 0 \end{array}\right], \qquad \tilde{d} = \left[\begin{array}{c} 0 \\ 0 \\ \vdots \\ 0 \\ \hline x^{\text{init}} \\ x^{\text{des}} \end{array}\right]$$

Example

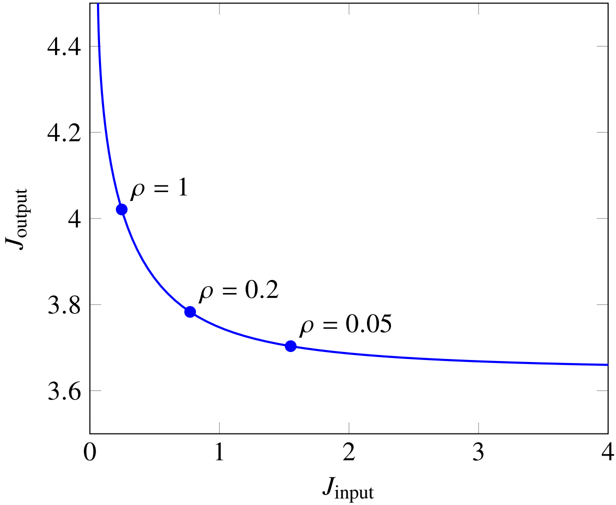
- time-invariant system: system matrices are constant

$$A = \begin{bmatrix} 0.855 & 1.161 & 0.667 \\ 0.015 & 1.073 & 0.053 \\ -0.084 & 0.059 & 1.022 \end{bmatrix}, \qquad B = \begin{bmatrix} -0.076 \\ -0.139 \\ 0.342 \end{bmatrix},$$

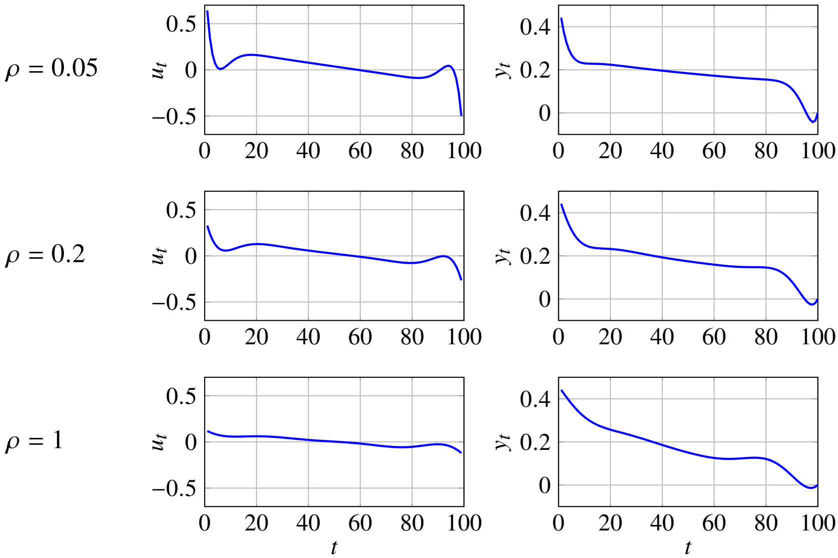
$$C = \begin{bmatrix} 0.218 & -3.597 & -1.683 \end{bmatrix}$$

- initial condition $x^{\mathrm{init}} = (0.496, -0.745, 1.394)$
- target or desired final state $x^{\mathrm{des}} = 0$
- $T = 100$

Optimal trade-off curve



Three points on the trade-off curve



Linear state feedback control

- linear state feedback control uses the input

$$u_t = Kx_t, \quad t = 1, 2, \ldots$$

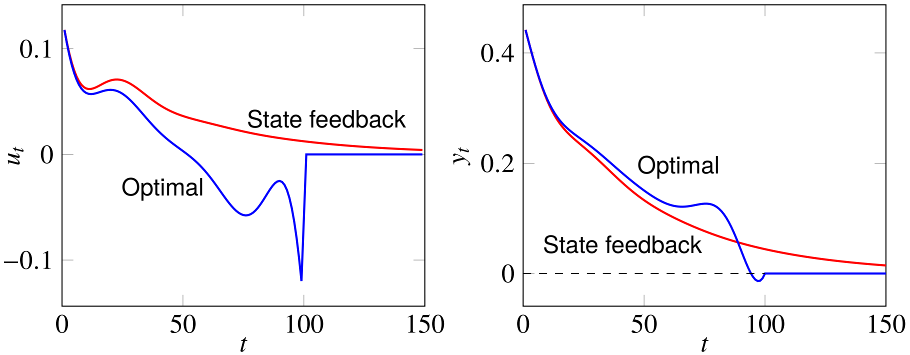
- K is *state feedback gain matrix*
- widely used, especially when x_t should converge to zero, T is not specified

- one choice for K: solve linear quadratic control problem with $x^{\text{des}} = 0$
- solution u_t is a linear function of x^{init}, so u_1 can be written as

$$u_1 = Kx^{\text{init}}$$

- columns of K can be found by computing u_1 for $x^{\text{init}} = e_1, \ldots, e_n$
- use this K as state feedback gain matrix

Example



- system matrices of previous example
- blue curve uses optimal linear quadratic control for $T = 100$
- red curve uses simple linear state feedback $u_t = Kx_t$

Outline

Portfolio optimization

Linear quadratic control

Linear quadratic state estimation

State estimation

- linear dynamical system model:

$$x_{t+1} = A_t x_t + B_t w_t, \quad y_t = C_t x_t + v_t, \quad t = 1, 2, \ldots$$

- x_t is *state* (n-vector)
- y_t is *measurement* (p-vector)
- w_t is *input* or *process noise* (m-vector)
- v_t is *measurement noise* or *measurement residual* (p-vector)
- we know A_t, B_t, C_t, and measurements $y_1, \ldots, y_T$
- w_t, v_t are unknown, but assumed small
- *state estimation*: estimate/guess $x_1, \ldots, x_T$

Least squares state estimation

$$\begin{array}{ll} \text{minimize} & J_{\text{meas}} + \lambda J_{\text{proc}} \\ \text{subject to} & x_{t+1} = A_t x_t + B_t w_t, \quad t = 1, \ldots, T-1 \end{array}$$

- variables: states $x_1, \ldots, x_T$ and input noise $w_1, \ldots, w_{T-1}$
- primary objective J_{meas} is sum of squares of measurement residuals:

$$J_{\text{meas}} = \|C_1 x_1 - y_1\|^2 + \cdots + \|C_T x_T - y_T\|^2$$

- secondary objective J_{proc} is sum of squares of process noise

$$J_{\text{proc}} = \|w_1\|^2 + \cdots + \|w_{T-1}\|^2$$

- $\lambda > 0$ is a parameter, trades off measurement and process errors

Constrained least squares formulation

$$\begin{array}{ll} \text{minimize} & \|C_1x_1 - y_1\|^2 + \cdots + \|C_Tx_T - y_T\|^2 + \lambda(\|w_1\|^2 + \cdots + \|w_{T-1}\|^2) \\ \text{subject to} & x_{t+1} = A_tx_t + B_tw_t, \quad t = 1, \ldots, T-1 \end{array}$$

- can be written as

$$\begin{array}{ll} \text{minimize} & \|\tilde{A}z - \tilde{b}\|^2 \\ \text{subject to} & \tilde{C}z = \tilde{d} \end{array}$$

- vector z contains the $Tn + (T-1)m$ variables:

$$z = (x_1, \ldots, x_T, w_1, \ldots, w_{T-1})$$

Constrained least squares formulation

$$\tilde{A} = \left[\begin{array}{cccc|ccc} C_1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & C_2 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & C_T & 0 & \cdots & 0 \\ \hline 0 & 0 & \cdots & 0 & \sqrt{\lambda}I & \cdots & 0 \\ \vdots & \vdots & & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & \sqrt{\lambda}I \end{array}\right], \qquad \tilde{b} = \left[\begin{array}{c} y_1 \\ y_2 \\ \vdots \\ y_T \\ \hline 0 \\ \vdots \\ 0 \end{array}\right]$$

$$\tilde{C} = \left[\begin{array}{cccccc|cccc} A_1 & -I & 0 & \cdots & 0 & 0 & B_1 & 0 & \cdots & 0 \\ 0 & A_2 & -I & \cdots & 0 & 0 & 0 & B_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & A_{T-1} & -I & 0 & 0 & \cdots & B_{T-1} \end{array}\right], \qquad \tilde{d} = 0$$

Missing measurements

- suppose we have measurements y_t for $t \in \mathcal{T}$, a subset of $\{1, \ldots, T\}$
- measurements for $t \notin \mathcal{T}$ are missing
- to estimate states, use same formulation but with

$$J_{\text{meas}} = \sum_{t \in \mathcal{T}} \|C_t x_t - y_t\|^2$$

- from estimated states $\hat{x}_t$, can estimate missing measurements

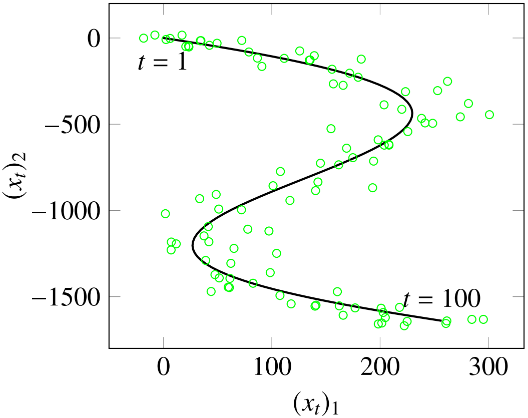
$$\hat{y}_t = C_t \hat{x}_t, \quad t \notin \mathcal{T}$$

Example

$$A_t = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad B_t = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \qquad C_t = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

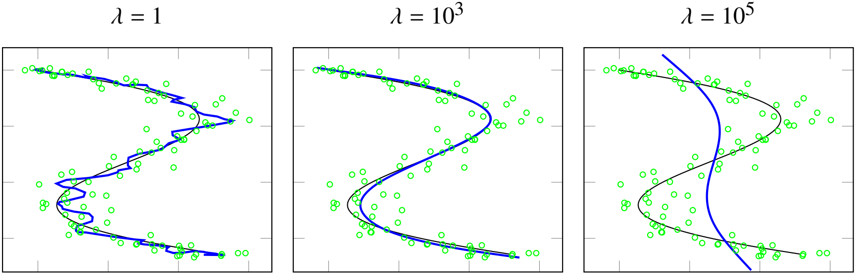
- simple model of mass moving in a 2-D plane
- $x_t = (p_t, z_t)$: 2-vector p_t is position, 2-vector z_t is the velocity
- $y_t = C_t x_t + w_t$ is noisy measurement of position
- $T = 100$

Measurements and true positions



- solid line is exact position $C_t x_t$
- 100 noisy measurements y_t shown as circles

Position estimates

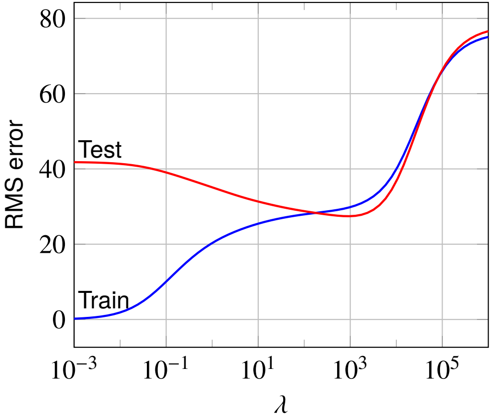


blue lines show position estimates for three values of λ

Cross-validation

- randomly remove 20% (say) of the measurements and use as test set
- for many values of λ, estimate states using other (*training*) measurements
- for each λ, evaluate RMS measurement residuals on test set
- choose λ to (approximately) minimize the RMS test residuals

Example



- cross-validation method applied to previous example
- remove 20 of the 100 measurements
- suggests using $\lambda \approx 10^3$