

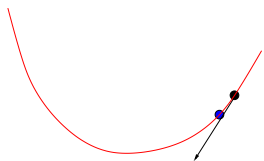
# Life Lessons from Machine Learning

James Kwok

# Make Life Simple

- survival of the fittest? do not **over-fit**
- how?
  - **regularizers**: penalize model complexity
  - own less live more: **sparse**
  - be humble: not high rank, go **low-rank**

- finding model parameters: optimization
- Newton's method, cutting plane, semidefinite program, ...
- **gradient descent**



## LOOP

- 1 find descent direction
- 2 choose stepsize
- 3 descent

- scalable, good for big data

# Life is Stochastic

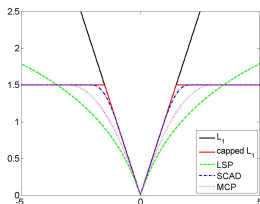


- make it less stochastic
- **variance reduction** in stochastic gradient descent

# Life is not Smooth



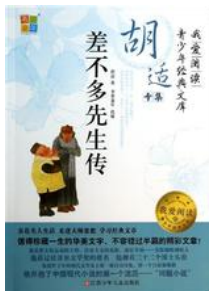
- nonsmoothness is good!
- nonsmooth loss / regularizer lead to better performance



- more challenging problem  $\rightarrow$  use more powerful tools

# Don't be too Calculative

- no need to get **exact** solutions
- **approximation solutions** are good enough in practice



# Team Work!

- “the burden is too heavy for me”
- **distributed** algorithm (**servers** and **workers**)
- no need to wait for everybody



- encourage **asynchrony**

# Summary

- 1 make life simple (use regularizers)
- 2 live life simple (scalable optimization)
- 3 make life less stochastic (variance reduction)
- 4 accept that life is not smooth (advanced optimization techniques)
- 5 don't be too calculative (approximation algorithms)
- 6 team work (distributed processing)