Recitation 4

ML 10701

Zeyu Jin

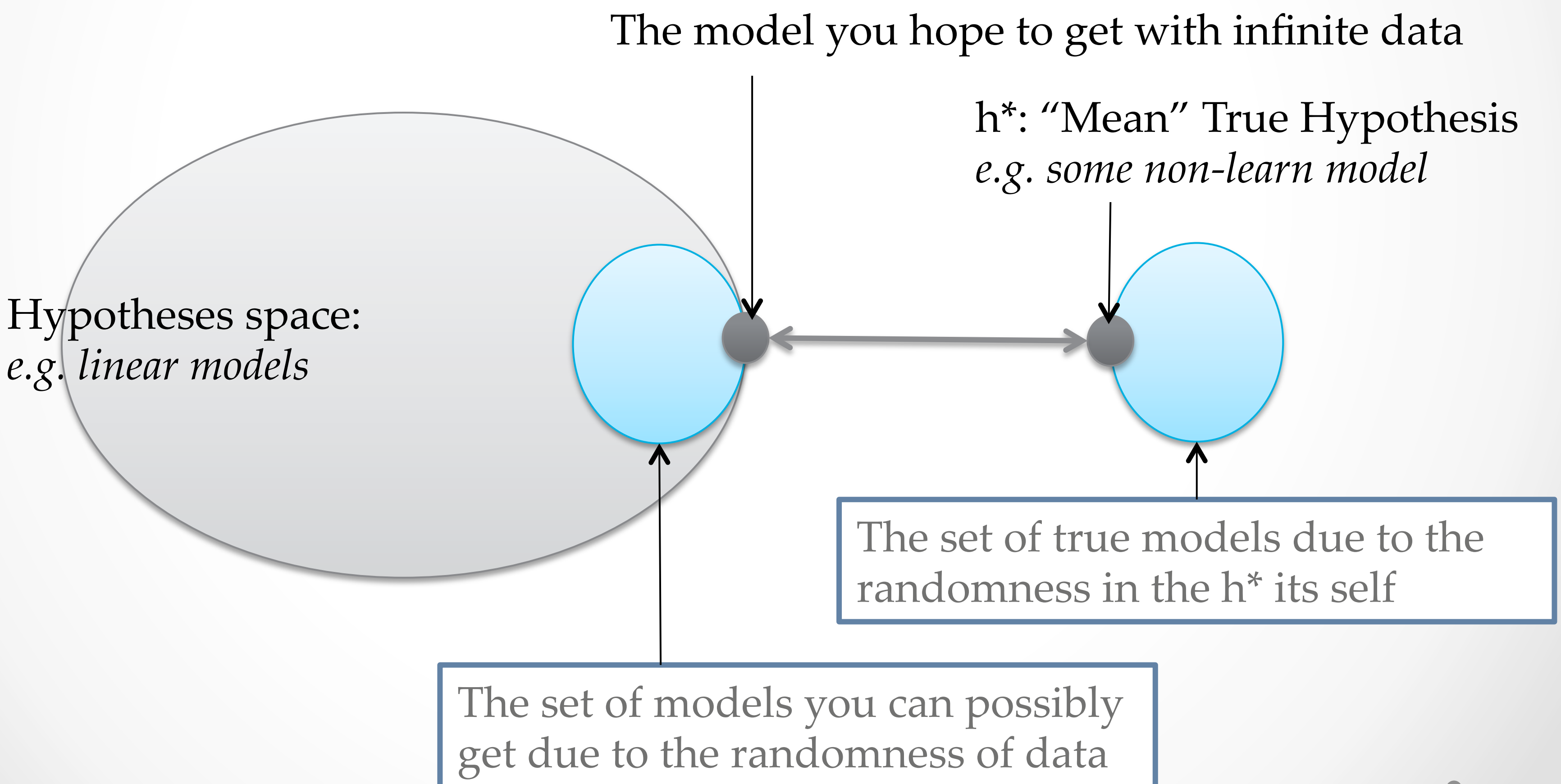
Outline

- Bias & Variance Trade-off
- Convex optimization

- A little bit about KNN

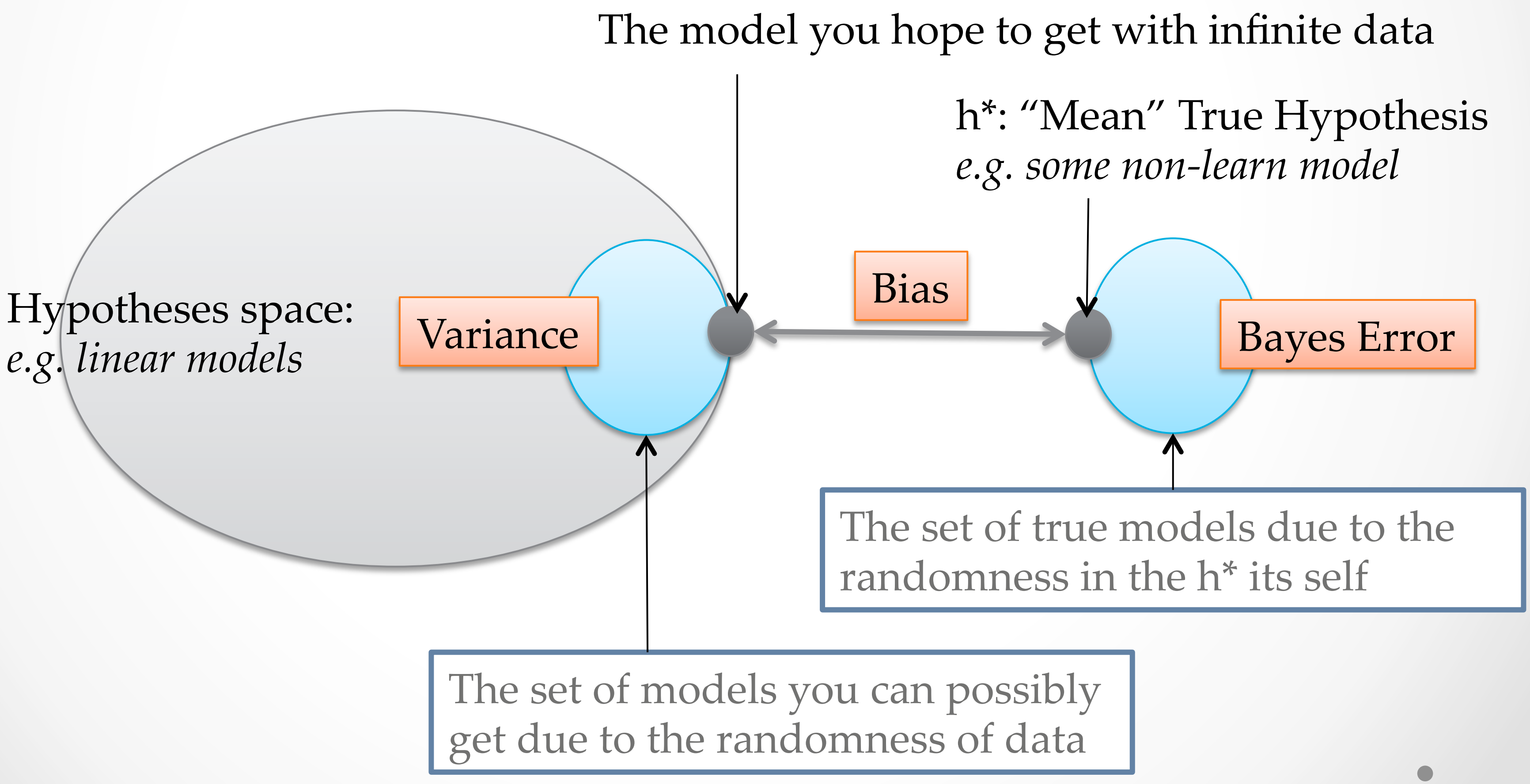
Bias Variance & Model Selection

- Bias-variance Decomposition



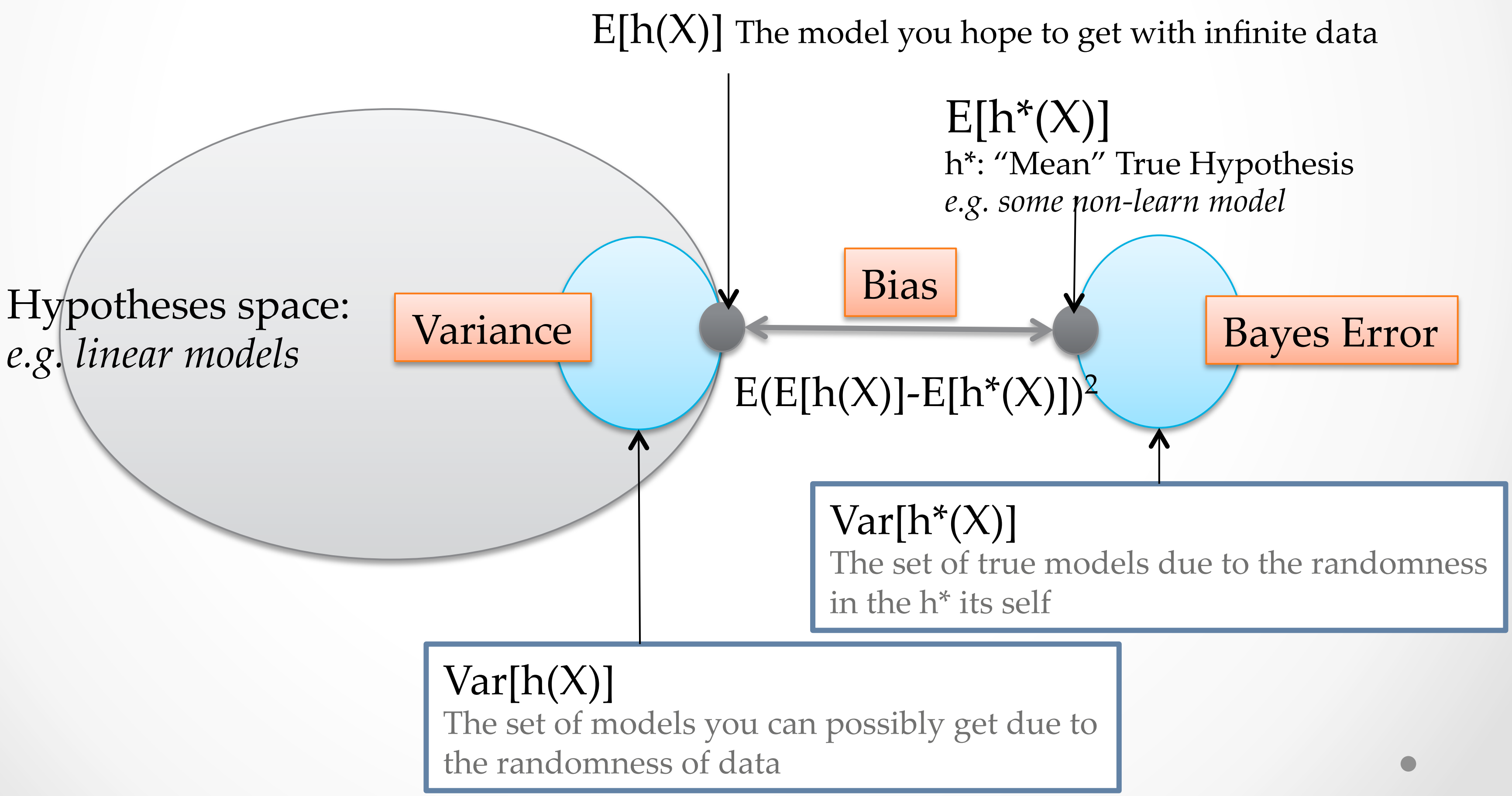
Bias Variance & Model Selection

- Bias-variance Decomposition



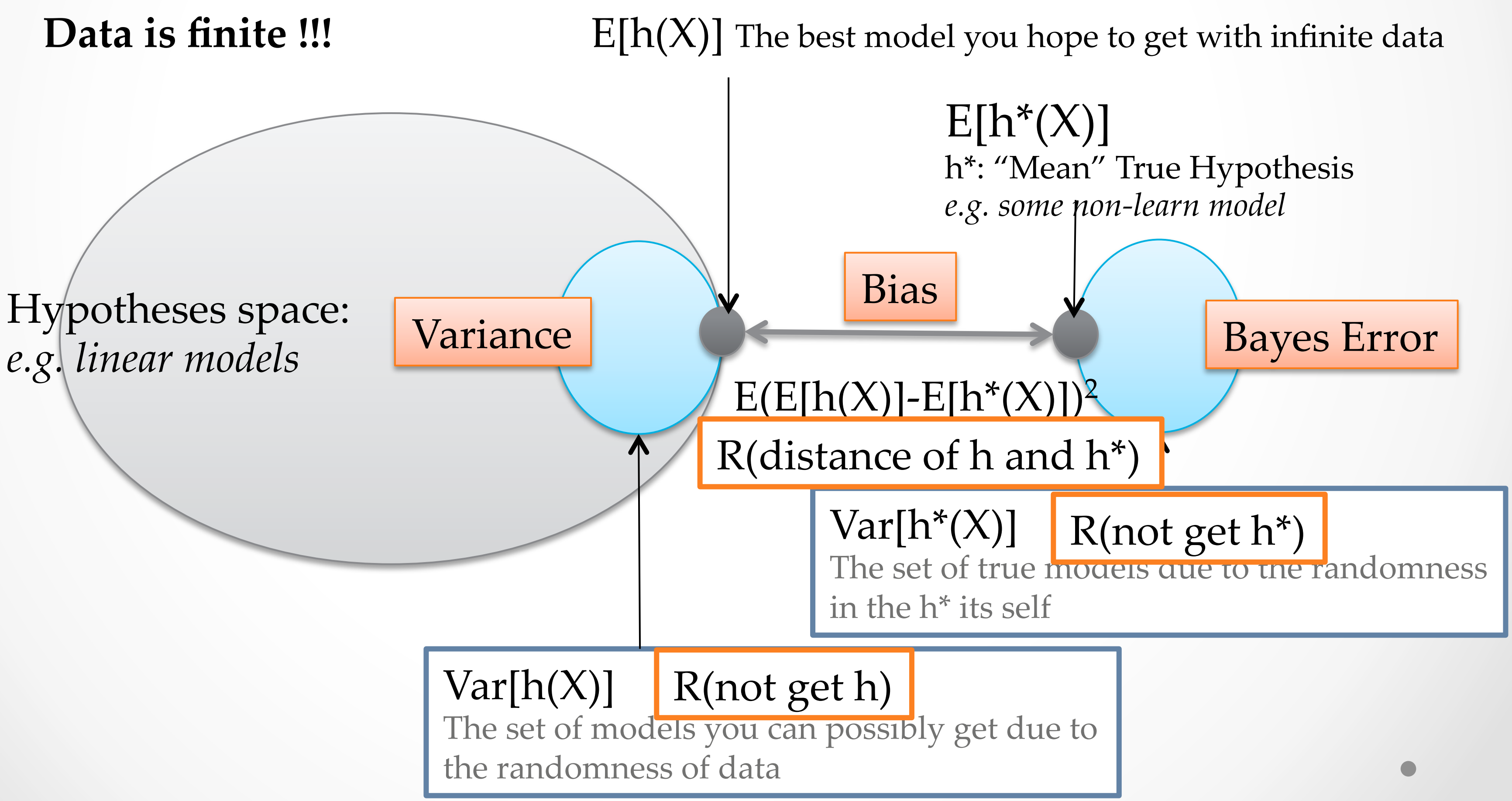
Bias Variance & Model Selection

- Bias-variance Decomposition



Bias Variance & Model Selection

- Bias-variance Decomposition



Bias Variance & Model Selection

- Bias-variance Decomposition

R(h(X),h*(X)) = Var[h(X)] + E(E[h(X)]-E[h*(X)])^2 + Var[h*(X)]

$$R(f) = \mathbb{E}[(f(X)-Y)^2] = \mathbb{E}[(f(X)-\mathbb{E}[f(X)])^2] + \mathbb{E}[(\mathbb{E}[f(X)] - f^*(X))^2] + \sigma^2$$

E(E[h(X)]-E[h*(X)])^2

R(distance of h and h*)

Var[h*(X)] R(not get h*)

The set of true models due to the randomness in the h* its self

Var[h(X)] R(not get h)

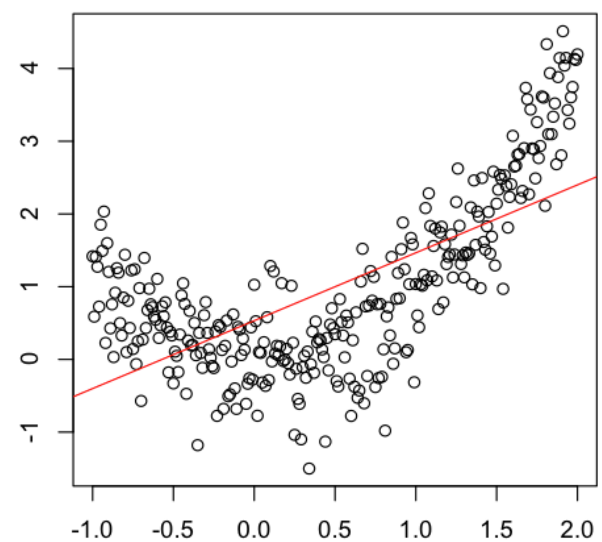
The set of models you can possibly get due to the randomness of data

Bias Variance & Model Selection

- Bias-variance Decomposition

$$R(h(X),h^*(X)) = Var[h(X)] + E(E[h(X)]-E[h^*(X)])^2 + Var[h^*(X)]$$

Case study: Regression



True Hypothesis plus variance: $h^*(X) = \beta_2 x^2 + \beta_1 x + \beta_0 + \epsilon$

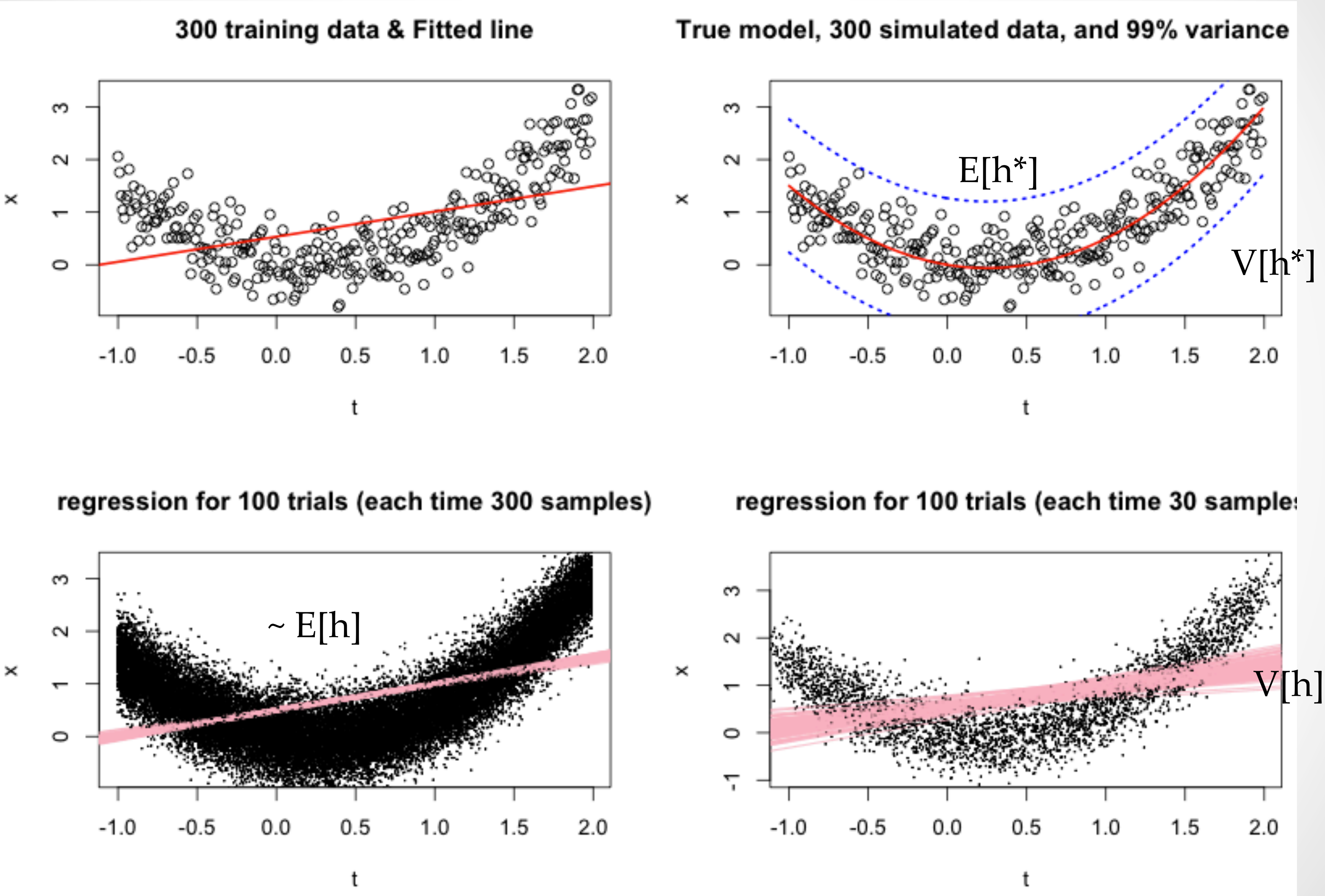
Estimated Hypothesis $h(X) = \widehat{\beta_1}(x^n)x + \widehat{\beta_0}(x^n)$

Variance of estimation $V\big(h(X)\big) = V_{x^n}\big[\widehat{\beta_1}(x^n)X + \widehat{\beta_0}(x^n)\big]$

Variance of true hypothesis $V\big(h^*(X)\big) = V[\epsilon] = \sigma^2$

The optimal hypothesis in your H space: $E[h(X)] = E_{x^n}\big[\widehat{\beta_1}(x^n)\big]x + E_{x^n}\big[\widehat{\beta_0}(x^n)\big] = \widehat{\beta_1}x + \widehat{\beta_0}$

The true hypothesis $E[h^*(X)] = \beta_2 x^2 + \beta_1 x + \beta_0$

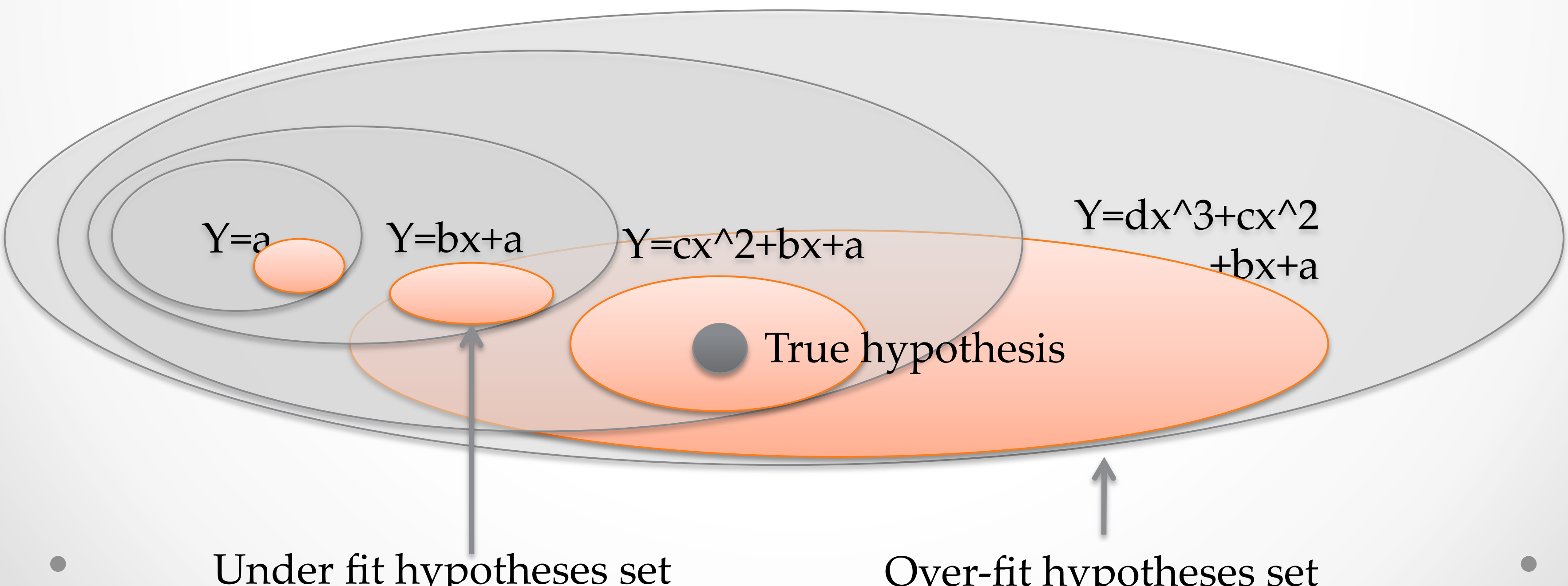

300 training data & Fitted line
True model, 300 simulated data, and 99% variance
E[h*]
V[h*]
regression for 100 trials (each time 300 samples)
~ E[h]
regression for 100 trials (each time 30 sample
V[h]
x
t
-1.0
-0.5
0.0
0.5
1.0
1.5
2.0
-1
0
1
2
3

Bias Variance & Model Selection

- Model Selection

$$R(h(X),h^*(X)) = Var[h(X)] \quad + \quad E(E[h(X)]-E[h^*(X)])^2 \quad + \quad Var[h^*(X)]$$

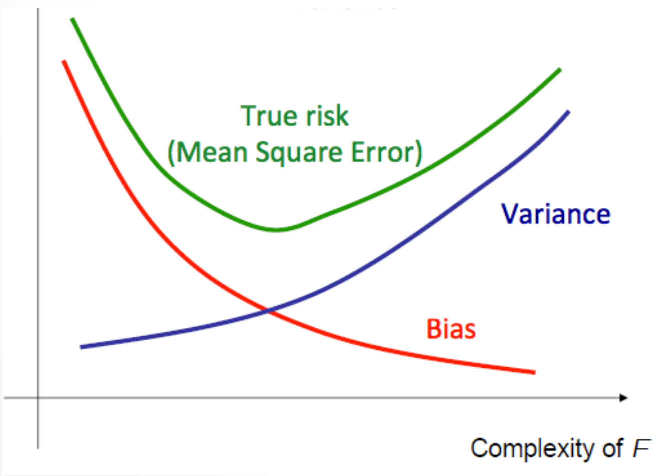
Goal: minimize **risk** by choosing the best hypotheses subspace
Why? Your estimator is based on some assumption of the model class



Bias Variance & Model Selection

- Model Selection

What is true Risk? Risk is test error
- In regression: risk is expected squared error
- In classification
  - risk can be the expected 0/1 loss = test error
  - Or some other form like expected hinge loss (SVM)



Why the true risk increases when Complexity of F gets bigger?

We have a larger hypotheses space => We have more possible models that can fit the random drawn data

Bias Variance & Model Selection

- Model Selection

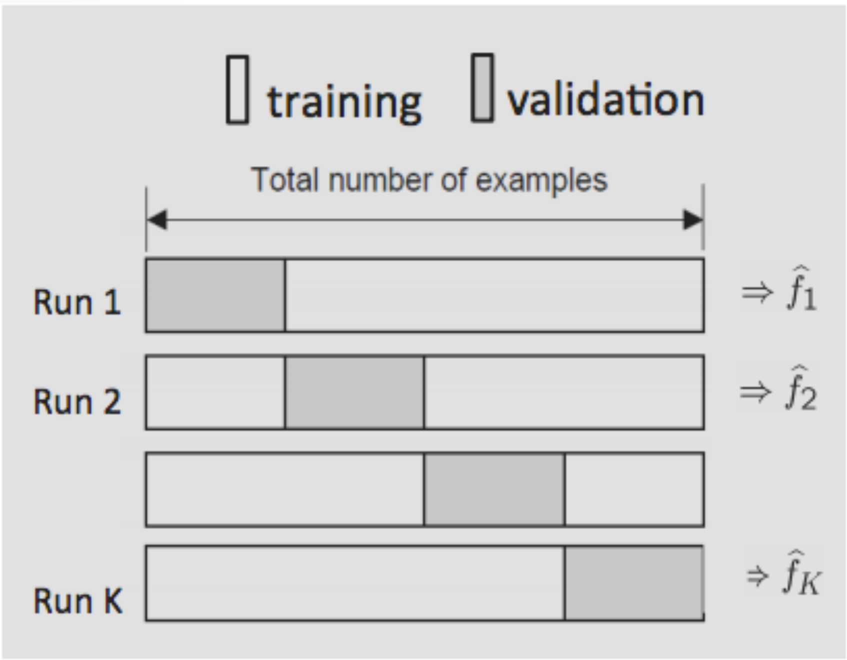
If we **know** the true risk, we can always get an optimal hypotheses set
But, we do not know it…

How to **estimate** the true risk?

1. CV and GCV
2. Structural risk minimization: regularization, panelizing using prior
3. AIC and BIC scoring, MDL, etc
4. Other criteria like Cp…

Bias Variance & Model Selection

- Model selection
  - CV & GCV



Estimating risk directly

Assumption:
p(X) ~ uniform({x1…xn})

It is approximately right when validation set is large enough

Bias Variance & Model Selection

- Model selection
  - CV & GCV

Estimating risk directly

Assumption:
  p(X) ~ uniform({x1…xn})

It is approximately right when validation set is large enough

K

Size of validation set

More data for training
⇒ Less biased

Less data for validating
=> Validation result inconsistent (large variance)

Bias Variance & Model Selection

- Model selection
  - Structural risk minimization

Penalize the model complexity in likelihood function

$$\widehat{f}_n = \arg\min_{f \in \mathcal{F}} \left\{ \widehat{R}_n(f) + C(f) \right\}$$

Without a prior: the information content of hypothesis space is huge because we have equal probability for each hypothesis set

Having a prior: the information of hypotheses space is reduced since we know which part of hypotheses space is more likely and thus reduces the complexity.

Leads to biased but less varied estimation

Bias Variance & Model Selection

- Model selection
  - Other criteria

Penalize the model complexity in likelihood function

Another reason to panelize the estimated risk

In regression, the bias of empirical risk is

$$\text{bias}(\widehat{R}_{\text{tr}}(S)) = \mathbb{E}(\widehat{R}_{\text{tr}}(S)) - R(S)) = -\frac{2}{n}\sum_{i=1} \text{Cov}(\widehat{Y}_i, Y_i)$$

Which is always a under-estimated risk

The under estimation needs to be added back to get a better approximation of R(S), the true risk

Bias Variance & Model Selection

- Model selection
  - Other Criteria

R(S) = R_{tr}(S) + something

Cp statistics

$$\widehat{R}(S) = \widehat{R}_{tr}(S) + \frac{2|S|\widehat{\sigma}^2}{n}$$

Cross validation is an approximation of Cp

$$\widehat{R}_{CV}(S) = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{Y_i - \widehat{Y}_i(S)}{1 - H_{ii}(S)}\right)^2$$

$$\widehat{R}_{CV}(S) \approx \frac{1}{n}\frac{\mathrm{RSS}(S)}{\left(1 - \frac{|S|}{n}\right)^2}. \quad \widehat{R}_{CV}(S) \approx \widehat{R}_{tr}(S) + \frac{2\widehat{\sigma}^2|S|}{n}$$

Bias Variance & Model Selection

- Model selection
  - AIC and BIC try to estimate true likelihood

AIC(S) = $-2\ell_S + 2|S|,$ Minimize AIC(S)

$$\mathrm{BIC}(S) = \ell_S - \frac{|S|}{2}\log n$$ Minimize -2BIC(S)

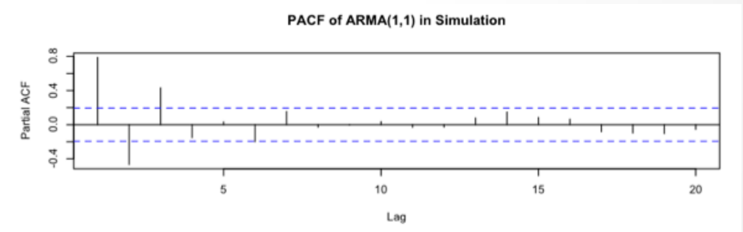
Bias Variance & Model Selection

- Model selection
  - AIC and BIC try to estimate true likelihood

Example: time series
Select the best ARMA(p,q) model

AIC

ma ar	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]
[1,]	4.77426	4.58545	4.54928	4.55494	4.55046	4.55623	4.56017	4.56624
[2,]	4.57242	4.55858	4.54877	4.56254	4.55497	4.56117	4.54987	4.55632
[3,]	4.55466	4.56130	4.54991	4.55617	4.54734	4.55147	4.55608	4.55942
[4,]	4.56129	4.55926	4.57457	4.55816	4.55688	4.53982	4.57491	4.56285
[5,]	4.56678	4.56451	4.55606	4.56690	4.53940	4.54652	4.53956	4.54563
[6,]	4.55892	4.56531	4.56586	4.57268	4.55747	4.56946	4.55835	4.56479
[7,]	4.56491	4.55202	4.57143	4.56021	4.56695	4.55845	4.56521	4.55682
[8,]	4.56286	4.54985	4.52739	4.56665	4.57316	4.56505	4.56742	4.57683
[9,]	4.56871	4.55571	4.54614	4.57354	4.57454	4.57233	4.54787	4.54488
[10,]	4.57443	4.58104	4.57003	4.52419	4.54137	4.56380	4.56666	4.50609
[11,]	4.58098	4.58984	4.55609	4.58530	4.55364	4.59737	4.57340	4.56734
[12,]	4.57566	4.57933	4.58238	4.52325	4.52279	4.52049	4.56033	4.54244
[13,]	4.58138	4.54642	4.57841	4.52604	4.56233	4.54471	4.55926	4.55922



More complex

BIC

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]
[1,]	3.81120	3.63472	3.61086	3.62883	3.63667	3.65475	3.67101	3.68940
[2,]	3.62168	3.62016	3.62267	3.64875	3.65350	3.67201	3.67303	3.69180
[3,]	3.61624	3.63519	3.63612	3.65469	3.65818	3.67463	3.69156	3.70722
[4,]	3.63519	3.64547	3.67310	3.66900	3.68004	3.67530	3.72271	3.72296
[5,]	3.65299	3.66304	3.66691	3.69006	3.67487	3.69432	3.69966	3.71805
[6,]	3.65744	3.67616	3.68902	3.70815	3.70526	3.72957	3.73077	3.74953
[7,]	3.67576	3.67518	3.70691	3.70800	3.72706	3.73087	3.74995	3.75387
[8,]	3.68602	3.68533	3.67518	3.72676	3.74558	3.74979	3.76448	3.78620
[9,]	3.70418	3.70350	3.70624	3.74596	3.75928	3.76939	3.75724	3.76657
[10,]	3.72223	3.74115	3.74246	3.70893	3.73842	3.77317	3.78834	3.74009
[11,]	3.74109	3.76226	3.74083	3.78235	3.76301	3.81906	3.80740	3.81366
[12,]	3.74808	3.76407	3.77944	3.73263	3.74448	3.75450	3.80665	3.80107
[13,]	3.76612	3.74347	3.78779	3.74773	3.79633	3.79103	3.81789	3.83017

The partial correlation shows
that the true model should be around ARMA(3,?)

Convex Optimization

- Overview
  - What is Optimization?

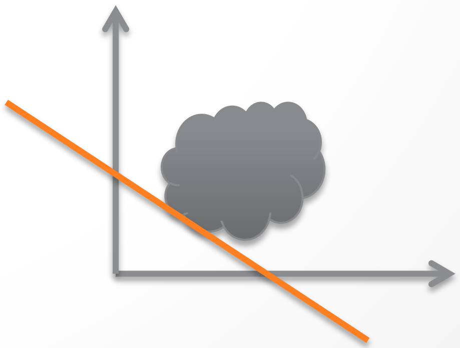
$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq b_i, \quad i = 1, \ldots, m. \end{array}$$

Least square problem:

$$\text{minimize} \quad f_0(x) = \|Ax - b\|_2^2 = \sum_{i=1}^{k} (a_i^T x - b_i)^2.$$

Linear Programming

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & a_i^T x \leq b_i, \quad i = 1, \ldots, m. \end{array}$$



Convex Optimization

- Overview
  - What is **Convex** Optimization?

The normal optimization problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq b_i, \quad i = 1, \ldots, m. \end{array}$$

Plus convexity constraint

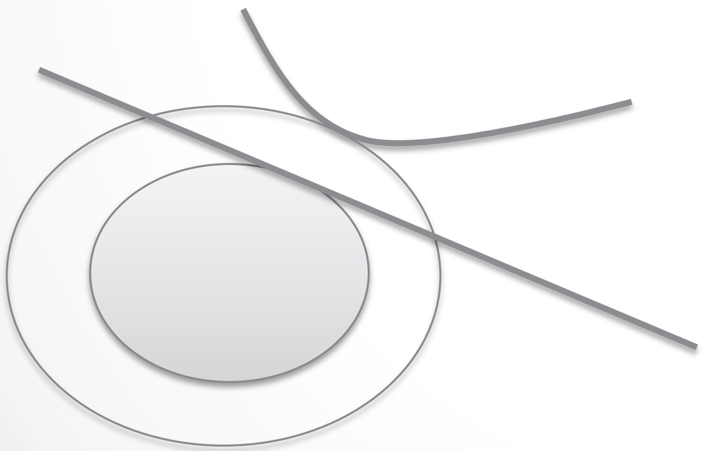
where the functions $f_0, \ldots, f_m : \mathbf{R}^n \rightarrow \mathbf{R}$ are convex, *i.e.*, satisfy

$$f_i(\alpha x + \beta y) \leq \alpha f_i(x) + \beta f_i(y)$$

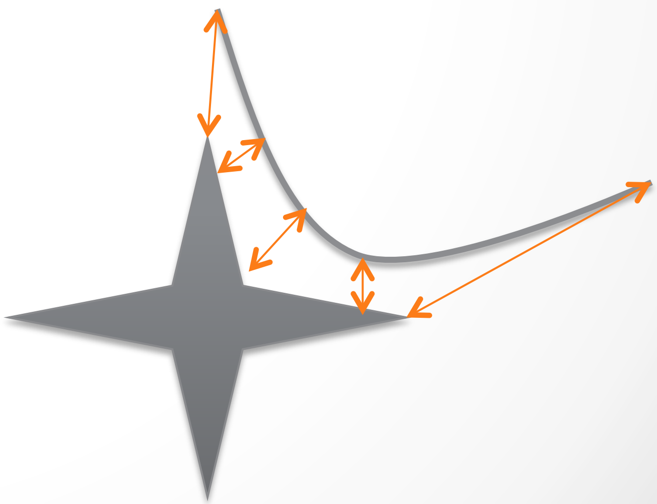
Convex Optimization

- Overview
  - Why convex function optimizible?

Convex =>
Local minimum = Global minimum



Non-convex =>
multiple local minimum



Convex Optimization

- Overview
    - How do we optimize Convex problem?

min $f_0(x)$

s.t. $f_i(x) <= b_i$
$i=1,2,\ldots,n$

$f_i(x)$ are convex

Most of convex problems:
Gradient descent, simulated annealing, EM (Slower)

Only a subset of convex problems:
Quadratic Programming (Faster)

If question can be solved by QP, then QP is preferred,
if not, we can try to convert the problem into a QP solvable problem

Convex Optimization

- Quadratic Programming
  - Sophisticated "technology" solving the optimization problem of

$$\min_U \frac{u^T R u}{2} + d^T u + c$$

Objective function: quadratic

$$\begin{array}{l} a_{11}u_1 + a_{12}u_2 + \ldots \le b_1 \\ \vdots \qquad \vdots \qquad \vdots \\ a_{n1}u_1 + a_{n2}u_2 + \ldots \le b_n \end{array}$$

Linear inequality constraints

$$\begin{array}{l} a_{n+1,1}u_1 + a_{n+1,2}u_2 + \ldots = b_{n+1} \\ \vdots \qquad \vdots \qquad \vdots \\ a_{n+k,1}u_1 + a_{n+k,2}u_2 + \ldots = b_{n+k} \end{array}$$

Equality constraints

Convex Optimization

- Quadratic Programming
  - Example: SVM

Linearly Separable

$$\min_{w} w^T w, \qquad s.t.$$

$$y_j(w^T x_j + b) \geq 1$$

☺

Non-linearly Separable

$$\min_{w} w^T w + \sum_{i-1}^{n} C\epsilon_i, \qquad s.t.$$

$$y_j(w^T x_j + b) \geq 1 - \epsilon_i$$

$$\epsilon_i \geq 0$$

☺

Quadratic Programming

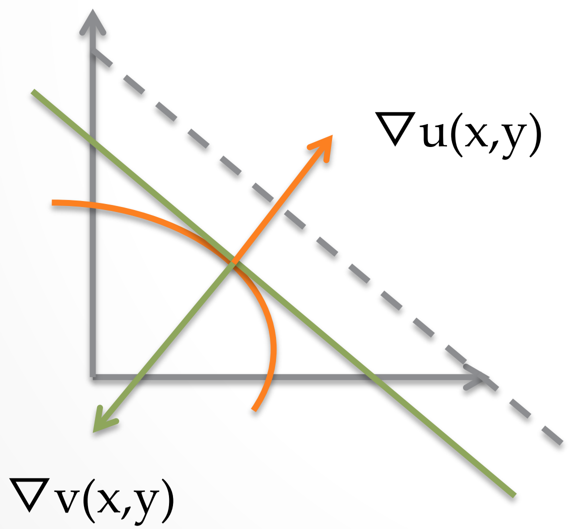
$$\min_U \frac{u^T R u}{2} + d^T u + c \qquad \text{s.t.}$$

$$\begin{array}{ll} a_{11}u_1 + a_{12}u_2 + \ldots \leq b_1 & a_{n+1,1}u_1 + a_{n+1,2}u_2 + \ldots = b_{n+1} \\ \vdots \quad \vdots \quad \vdots & \vdots \quad \vdots \quad \vdots \\ a_{n1}u_1 + a_{n2}u_2 + \ldots \leq b_n & a_{n+k,1}u_1 + a_{n+k,2}u_2 + \ldots = b_{n+k} \end{array}$$

Convex Optimization

- Quadratic Programming
  - Dual form

Lagrange Multiplier



Minimize v(x,y)
s.t. u(x,y) = C

The gradient of v and u should be perpendicular to each other

=>

$$\nabla u(x,y) = \lambda \nabla v(x,y)$$

Convex Optimization

- Quadratic Programming
  - Primal vs. dual

Primal optimization problem (variables x):

$$\begin{aligned} &\text{minimize} && f_0(x) = \textstyle\sum_{i=1}^n x_i \log x_i \\ &\text{subject to} && Ax \preceq b \\ & && \mathbf{1}^T x = 1 \end{aligned}$$

Dual optimization problem (variables λ, ν):

$$\begin{aligned} &\text{maximize} && -b^T\lambda - \nu - e^{-\nu-1} \textstyle\sum_{i=1}^n e^{-a_i^T \lambda} \\ &\text{subject to} && \lambda \succeq 0 \end{aligned}$$

Convex Optimization

- Quadratic Programming
  - Why we want to use Dual form

QP: More efficient

Works for some problems that are not obviously QP at the first glance

In SVM: kernel tricks !!!
In the dimension of w is infinity, we cannot solve it by its primal form

KNN

- Decision boundary
  - Which one is more likely to over-fit the data?
  - Which one's K is larger?
  - What will the boundary if varying the value of of K