

Machine Learning II

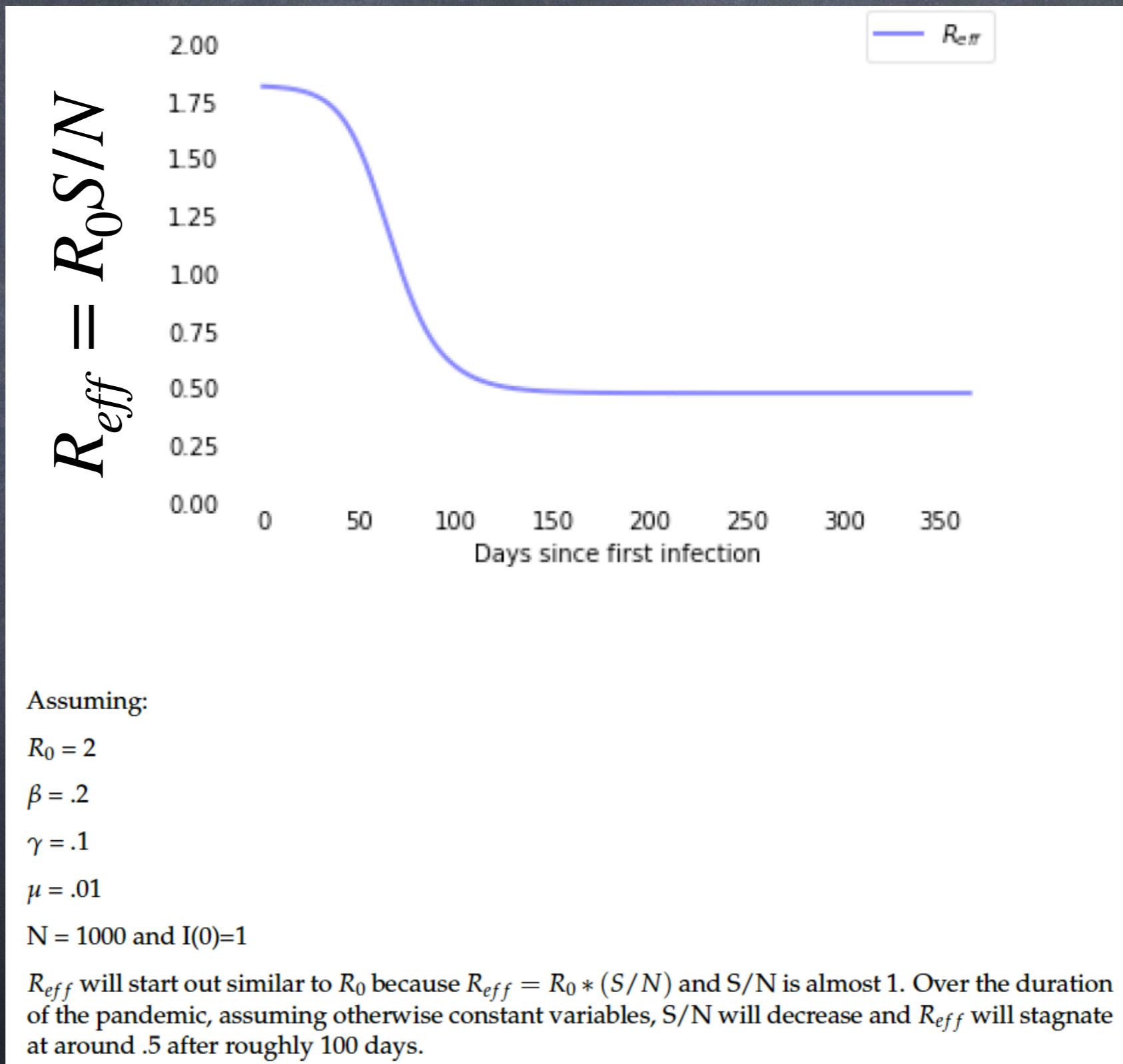
Week #4

Jan Nagler

Comments to Assignment 1

$$R_{eff} = R_0 S/N$$

Please take care to have correct labels



Comments to Assignment 1

Extensive, reflected in points given
(#38 received)

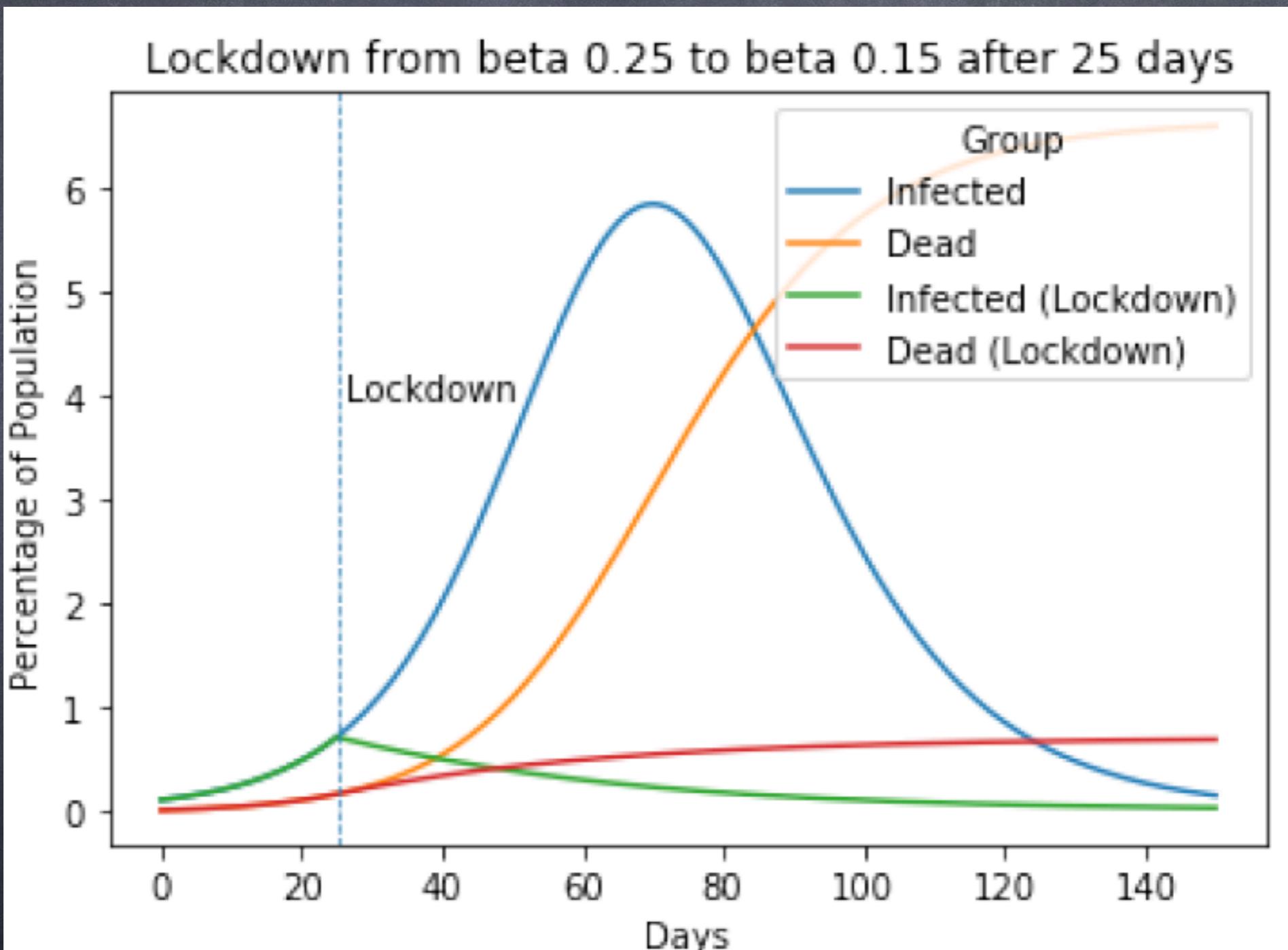
Output only **significant digits** and not 16 or 128

Avoid wrong labels or broken formulas

Avoid display of warnings and error messages

Avoid misnamed or incomplete filenames

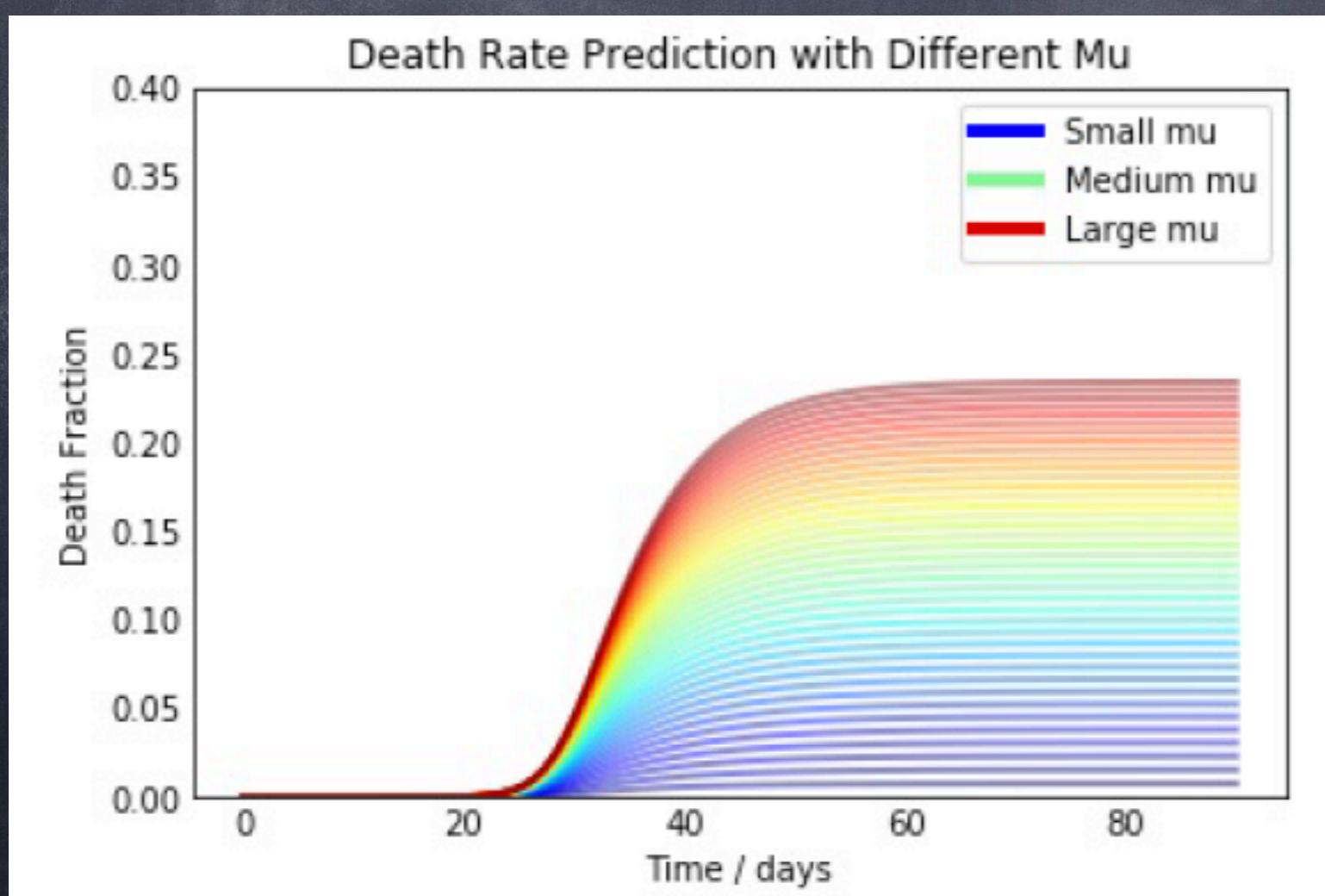
Comments to Assignment 1



Comments to Assignment 1

```
#####
# Constraint:
# gamma + miu <= 1
# 0 <= gamma, miu, beta <= 1
#####
# beta / R_nought - miu might introduce a negative gamma, which is not
→realistic.
```

Yaozhuahuai He



Yuge Xue(Aurora)



Expectation Maximization (EM)



Expectation Maximization

...when there is something missing

„Zurzeit können
keine Informationen
dargestellt werden.“

Conflicting bus display in Zurich:
“Currently no information can be displayed”

Expectation Maximization (EM)

Sort of soft clustering approach, based on information theory

Mixture models

Generative “model”, means pdf
typically each cluster is Gaussian or Multinomial (e.g., coin)

Hidden / latent variables:
Parameters of pdf

Gaussians: mean & co(variance)

Multinomial: probabilities of events

EM estimates iteratively the parameters,
by the lower bound trick that
maximizes the $\log(p(x|\theta))$

$$\log p(x|\theta)$$

Expectation Maximization (EM)

Goal: Maximize log likelihood

$$\theta_{MLE} = \operatorname{argmax}_{\theta} \log p(x | \theta)$$

(instead of maximize pdf itself,
avoid problem of underflow small prob by using log,
analytically: easier to handle sums than products)

Problem: Hard to do in closed form, e.g., many local maxima,
gradient methods, Newton-based maybe hard to compute or to implement

Expectation Maximization (EM)

Theory based on **variational methods**,
i.e., free distribution is introduced for obtaining
lower bound error in optimization step

Advantages:

Simple concept & implementation, each iteration leads typically to improvement

Improvement (rate of convergence) fast in the beginning

EM works best when characterisation of process is not substantially incomplete

Disadvantages:

EM can require **many** iterations
Higher **dimensionality** may substantially slow down the
E-step and reduce improvement

Expectation Maximization (EM)

#1: EM Coin toss (Primer pdf + iPad)

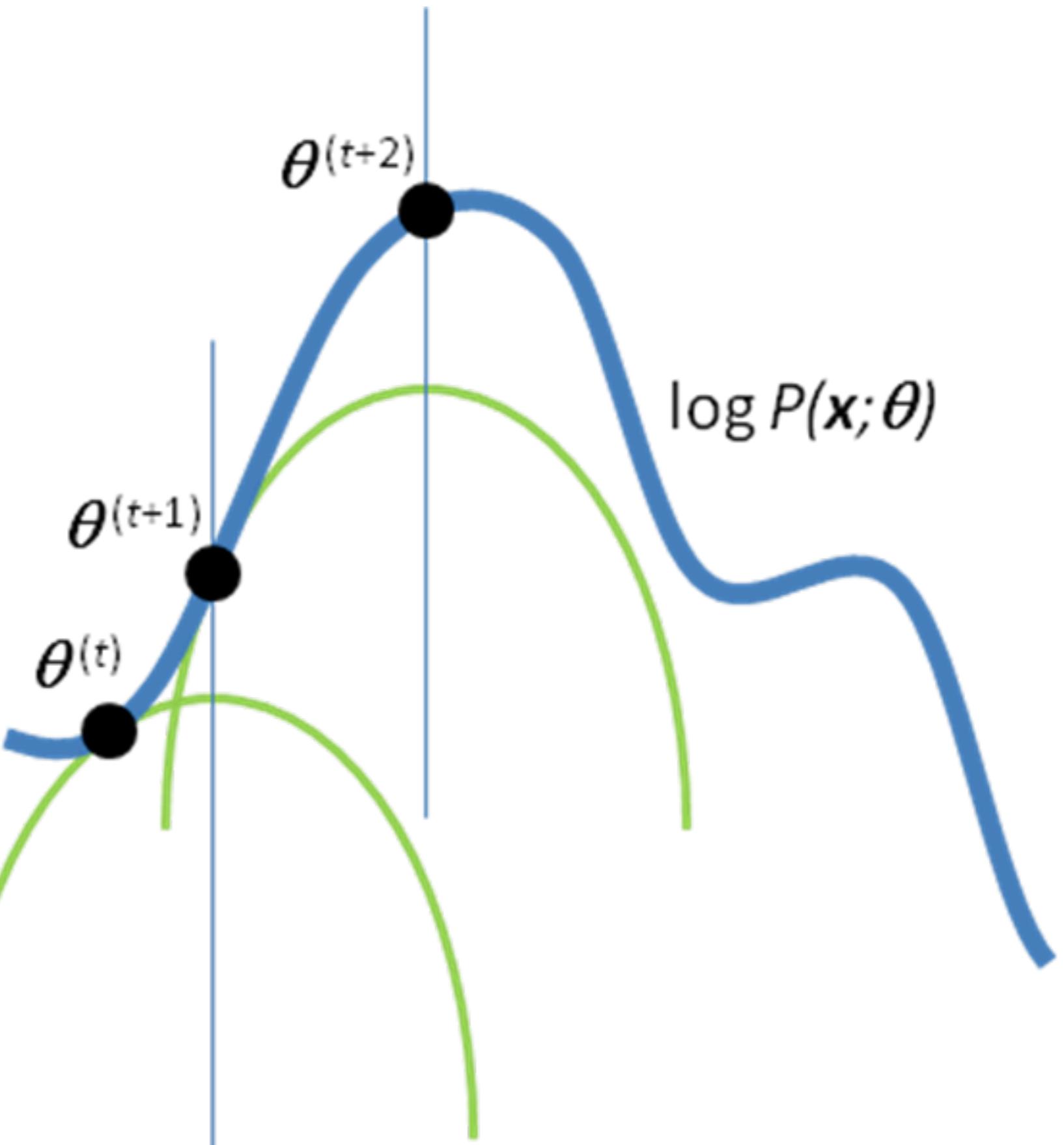
#2: General theoretical framework EM

#3: Theoretical framework EM coin toss

#4: EM Gaussian Mixtures

#5: Final comments

Expectation Maximization



Expectation Maximization (EM) – Coin flips – M Step

$$Q = Q_i(z_i) := p(z_i, x_i \mid \theta)$$

$$\begin{aligned} &= E \left[\log \prod_{i \leq n} [p_A p^{x_i} (1-p)^{1-x_i}]^{z_i} [p_B q^{x_i} (1-q)^{1-x_i}]^{1-z_i} \right] \\ &= \sum_{i \leq n} E[z_i | x_i, \theta^{(t)}] [\log(p_A) + x_i \log p + (1-x_i) \log(1-p)] \\ &\quad + (1 - E[z_i | x_i, \theta^{(t)}]) [\log p_B + x_i \log q + (1-x_i) \log(1-q)] \end{aligned}$$

From this, we seek:

$$\frac{\partial Q}{\partial p_A} = 0 \quad \frac{\partial Q}{\partial p} = 0 \quad \frac{\partial Q}{\partial q} = 0$$

iPad / exercise

Expectation Maximization

EM is a Bayesian statistical inference method

Relations between

Maximum Likelihood Estimation (MLE)

Maximum A posteriori (MAP) estimate

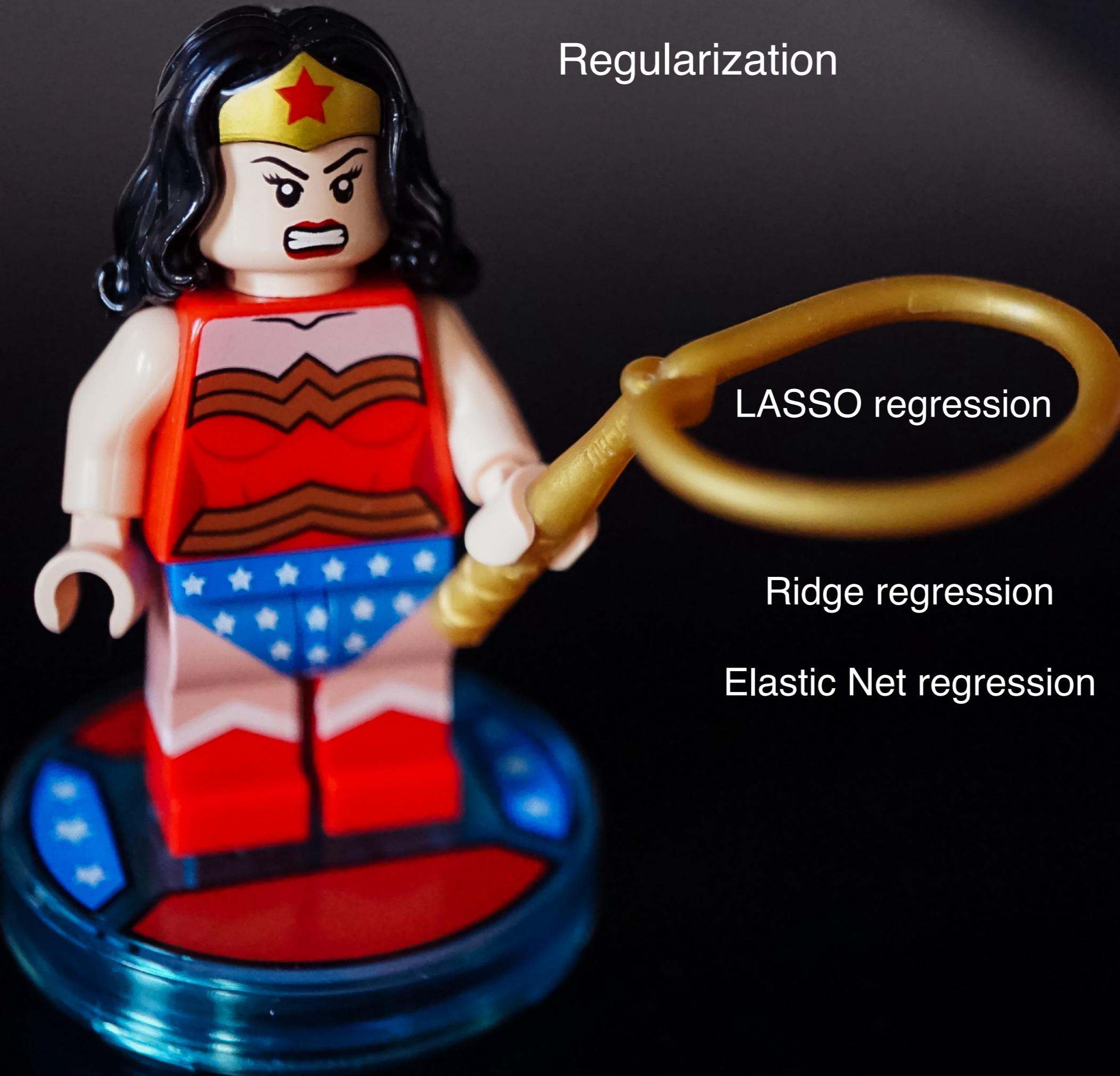
and Expectation-Maximization (EM) algorithm

$$\theta_{MLE} = \operatorname{argmax}_{\theta} \log p(x | \theta)$$

$$\theta_{MAP} = \operatorname{argmax}_{\theta} \log p(\theta | x) = \operatorname{argmax}_{\theta} \log \left(\frac{p(x | \theta)g(\theta)}{\int_{\theta'} p(x | \theta')g(\theta') d\theta'} \right)$$

$$= \operatorname{argmax}_{\theta} \log [p(x | \theta)g(\theta)]$$

Regularization



LASSO regression

Ridge regression

Elastic Net regression

Regularization

Ridge, Lasso and Elastic Net regression are seeking to alleviate the consequences of correlations / dependencies / multicollinearity of fit parameters

Aim is a parsimonious model (low variance, low bias)

Regularization imposes an upper threshold on the values taken by the coefficients, thereby producing a more parsimonious solution, and a set of coefficients with smaller variance, together with negative coefficient correlations.

Regularization

Ridge Regression:

Performs L2 regularization,

i.e. adds penalty equivalent to square of the magnitude of coefficients,
with

minimization objective =

sum of squared residuals + factor * (sum of square of coefficients)

Lasso Regression:

Performs L1 regularization,

i.e. adds penalty equivalent to absolute value of the magnitude of coefficients
with

minimization objective =

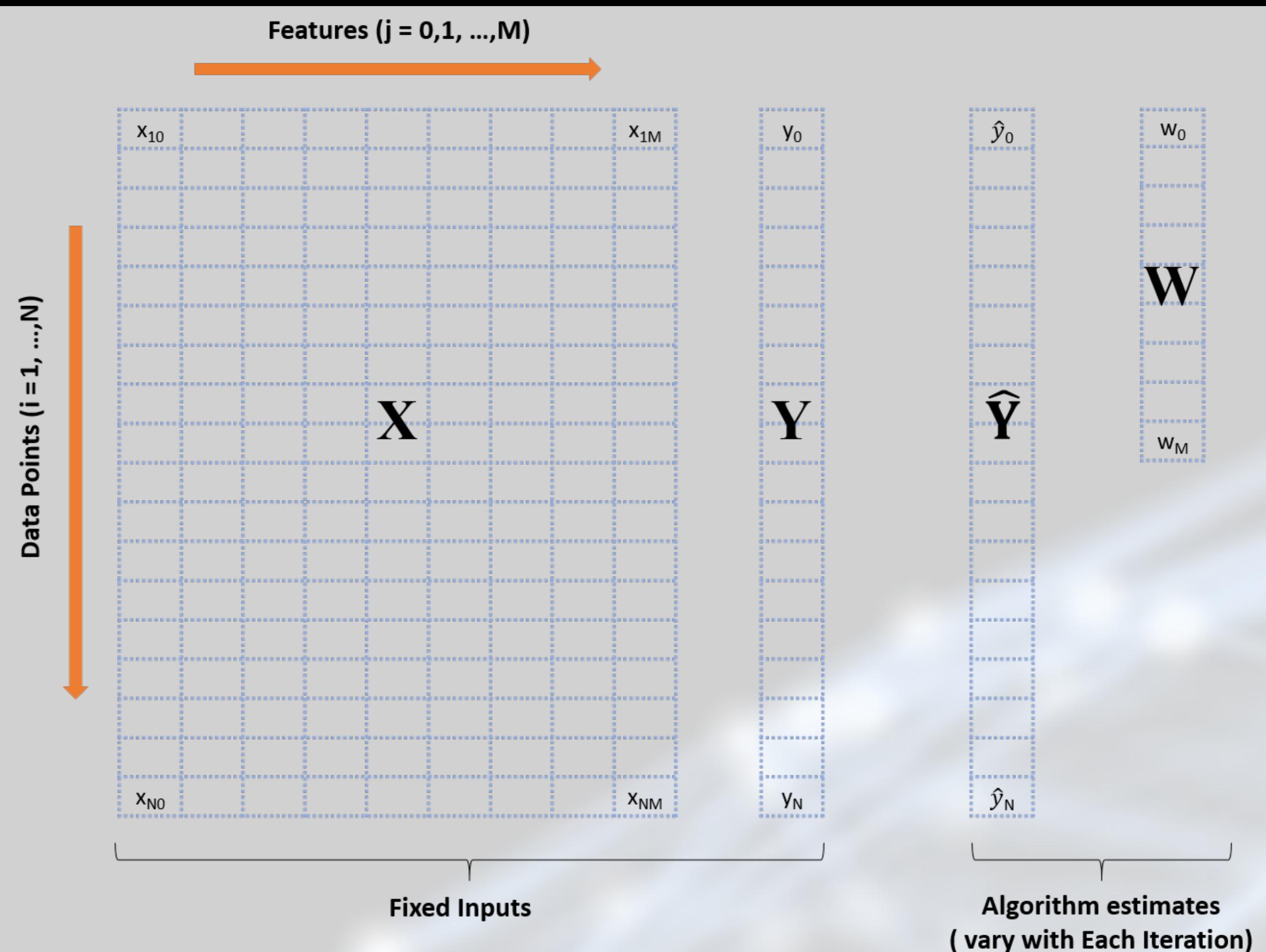
ssr + factor * (sum of absolute value of coefficients)

Elastic Net: Combination

sum of squared residuals = OLS



Regression: General case, general notation (incl. deep nets)



Ridge regression

Minimize

$$\sum_{i=1}^N \left[y_i - \sum_{j=0}^M w_j x_{ij} \right]^2 + \lambda \sum_{j=0}^M w_j^2$$

penalty

is equivalent to (proof in lecture)

Minimize $\sum_{i=1}^N \left[y_i - \sum_{j=0}^M w_j x_{ij} \right]^2$ under constraint $\sum_{j=0}^M w_j^2 < const$

shrinkage property

$$\lambda = 0 \rightarrow const = \infty$$

$$\lambda > 0 \rightarrow const < \infty$$

Ridge is an example of convex optimization
(strictly convex in omegas)



Ridge has a closed-form solution, as derived in the lecture

Least Absolute Shrinkage and Selection Operator (LASSO) regression

$$\text{Minimize} \quad \sum_{i=1}^N \left[y_i - \sum_{j=0}^M w_j x_{ij} \right]^2 + \lambda \sum_{j=0}^M |w_j|$$



is equivalent to (proof in lecture)

$$\text{Minimize} \quad \sum_{i=1}^N \left[y_i - \sum_{j=0}^M w_j x_{ij} \right]^2 \quad \text{under constraint} \quad \sum_{j=0}^M |w_j| < const$$

$$\lambda = 0 \rightarrow const = \infty$$

$$\lambda > 0 \rightarrow const < \infty$$

LASSO is an example of nonlinear optimization
(quadratic programming yields efficiently approximation)

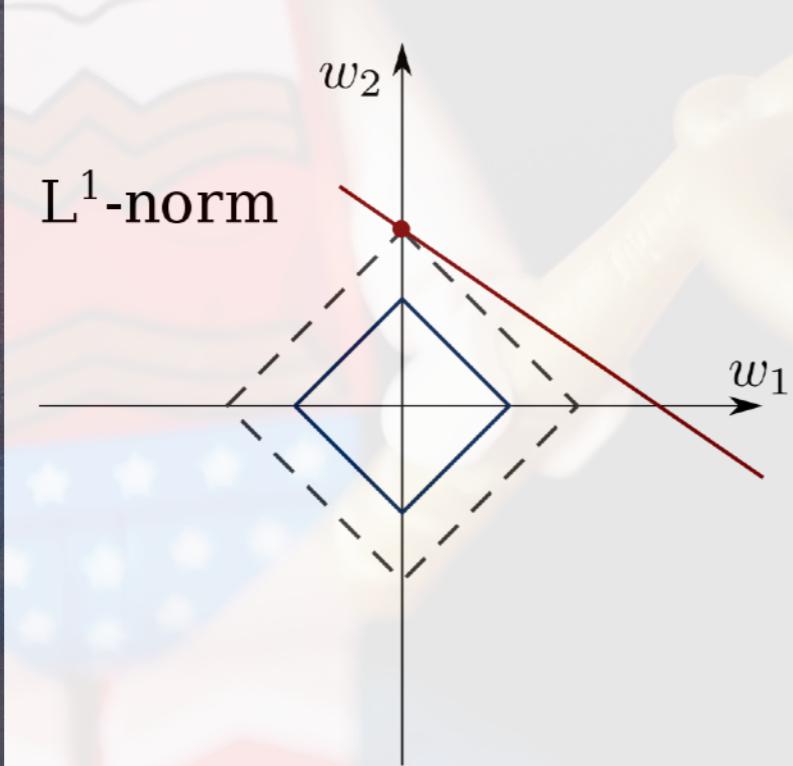


LASSO has no closed-form solution (nonlinearity)

LASSO

hard thresholding

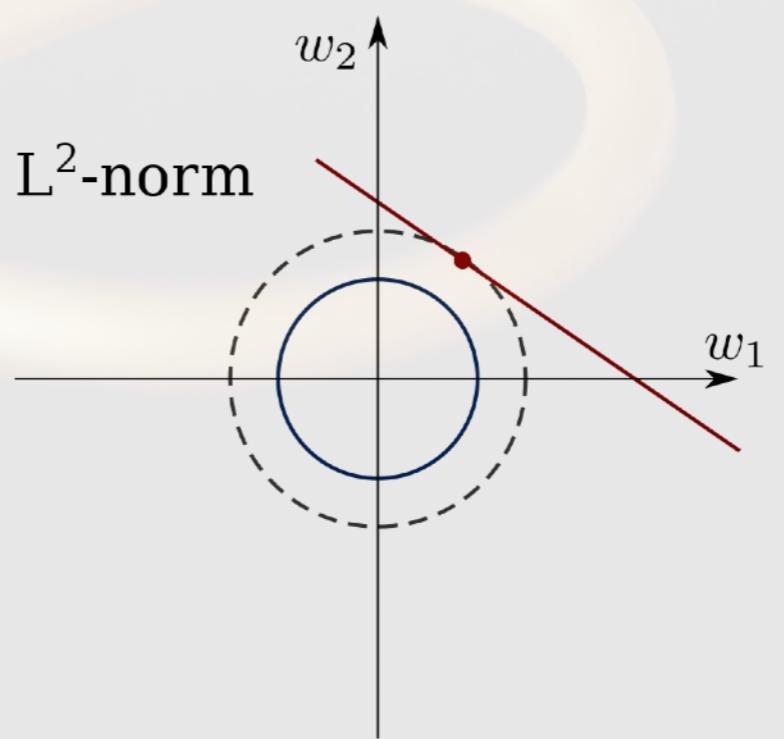
$$|w_1| + |w_2| < const$$



Ridge

soft thresholding

$$|w_1|^2 + |w_2|^2 < const$$



LASSO can be optimized for
prediction error
or

feature selection

but typically not both!

Some omegas become zero!

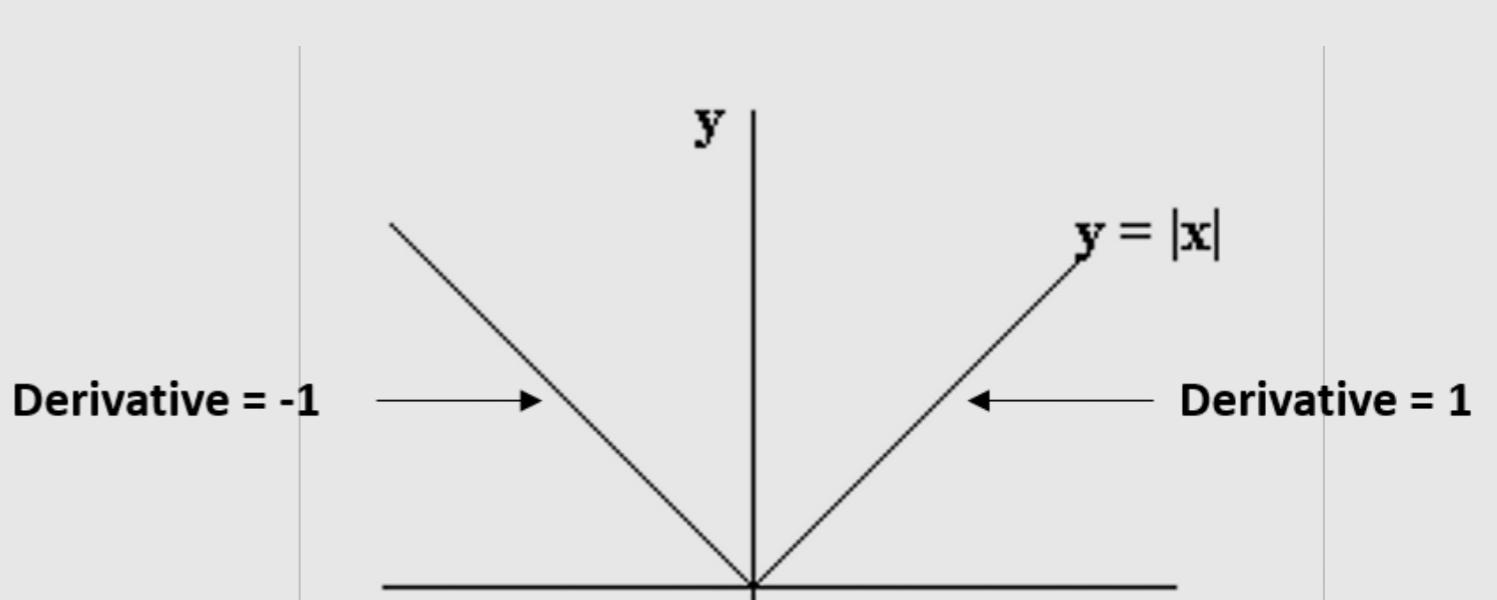
Lambda choice for ridge:
Information criterion or
min. prediction error / CV,
omega never zero

Ridge Gradient Descent

$$w_j^{t+1} = w_j^t - \eta \left[-2 \sum_{i=1}^N x_{ij} \left\{ y_i - \sum_{k=0}^M w_k * x_{ik} \right\} + 2\lambda w_j \right]$$

$$w_j^{t+1} = (1 - 2\lambda\eta)w_j^t + 2\eta \sum_{i=1}^N x_{ij} \left\{ y_i - \sum_{k=0}^M w_k * x_{ik} \right\}$$

Lasso Gradient Descent?



iPad, theory, proof (live, similar to next 4 slides)

Proof idea: Const $\leftrightarrow \lambda$

♦

Min $\underset{\underline{w}}{\parallel \underline{y} - \underline{x}\underline{w} \parallel^2}$, $\|\underline{w}\|_2^2 \leq c = \text{const.}$

Solution

$$g(\lambda) = \inf_{\underline{w}} L(\underline{w}, \lambda)$$

Lagrange Method

$$= \inf_{\underline{w}} [\parallel \underline{y} - \underline{x}\underline{w} \parallel^2 + \lambda (\|\underline{w}\|_2^2 - c)]$$

$$= \parallel \underline{y} - \underline{x} (\underline{x}^T \underline{x} + \lambda \underline{I})^{-1} \underline{x}^T \underline{y} \parallel^2$$

$$\frac{\partial g}{\partial \lambda} = 0$$

$$\frac{\partial g}{\partial \lambda} = \dots = 0 \Rightarrow$$

$$g(\lambda) = \| y - \underbrace{\underbrace{x^T x}_{= \lambda} + \lambda \underbrace{1 \cdot 1^T}_{= 1^T} \underbrace{x^T y}_{= y^T} \|^2$$

+ ~~λ~~ $(\| (\underbrace{x^T x}_{= \lambda} + \lambda \underbrace{1 \cdot 1^T}_{= 1^T})^{-1} \underbrace{x^T y}_{= y^T} \|^2 - c)$

$$\frac{\partial g}{\partial \lambda} = \underbrace{\dots}_{\text{constant}} \Rightarrow C_{\text{const}} = \| f(\lambda) \|_2^2$$

7.1

$$\boxed{C_{\text{const}} = f(\lambda)}$$

$$\boxed{\lambda = h(\text{const.})}$$

Observation 1 $\text{rss}(\underline{\omega})$ is convex in $\underline{\omega}$

$$\Leftrightarrow \underline{\underline{X}}^T \underline{\underline{X}} \quad \text{full rank}$$

Observation 2

$$\begin{aligned}\frac{\partial}{\partial \underline{\omega}} (\underline{\underline{X}}^T \underline{\underline{A}} \underline{\underline{\omega}}) &= \underline{\underline{A}} \underline{\underline{\omega}} + \underline{\underline{A}}^T \underline{\underline{\omega}} \\ &= (\underline{\underline{A}} + \underline{\underline{A}}^T) \underline{\underline{\omega}}\end{aligned}$$

M1 $\text{rss}(\underline{\omega}, \lambda) = (\underline{\underline{y}} - \underline{\underline{X}} \underline{\omega})^T (\underline{\underline{y}} - \underline{\underline{X}} \underline{\omega}) + \lambda \underline{\omega}^T \underline{\omega}$

↳ $\frac{\partial}{\partial \underline{\omega}} \text{rss}(\underline{\omega}, \lambda) = 2(\underline{\underline{X}}^T \underline{\underline{X}}) \underline{\omega} + 2\lambda \underline{\omega} - 2\underline{\underline{X}}^T \underline{\underline{y}} = 0$

$\Rightarrow 2(\underline{\underline{X}}^T \underline{\underline{X}}) \underline{\omega} + 2\lambda \underline{\omega} = 2\underline{\underline{X}}^T \underline{\underline{y}}$ Linear eqn. in $\underline{\omega}$

$$\Rightarrow \underline{w} = \left(\underline{\underline{X}}^T \underline{\underline{X}} + \lambda \underline{\underline{I}} \right)^{-1} \underline{\underline{X}}^T \underline{\underline{y}}$$

↑

Closed form solution
of Ridge Problem

$$\begin{aligned} \underline{\underline{A}} \underline{\underline{x}} &= \underline{\underline{b}} \\ \underline{\underline{x}} &= \underline{\underline{A}}^{-1} \underline{\underline{b}} \end{aligned}$$

Elastic Net

LASSO – many variables useless

$$|w_1| + |w_2| < const$$

ssr + L1 penalty

Ridge – most variables useful

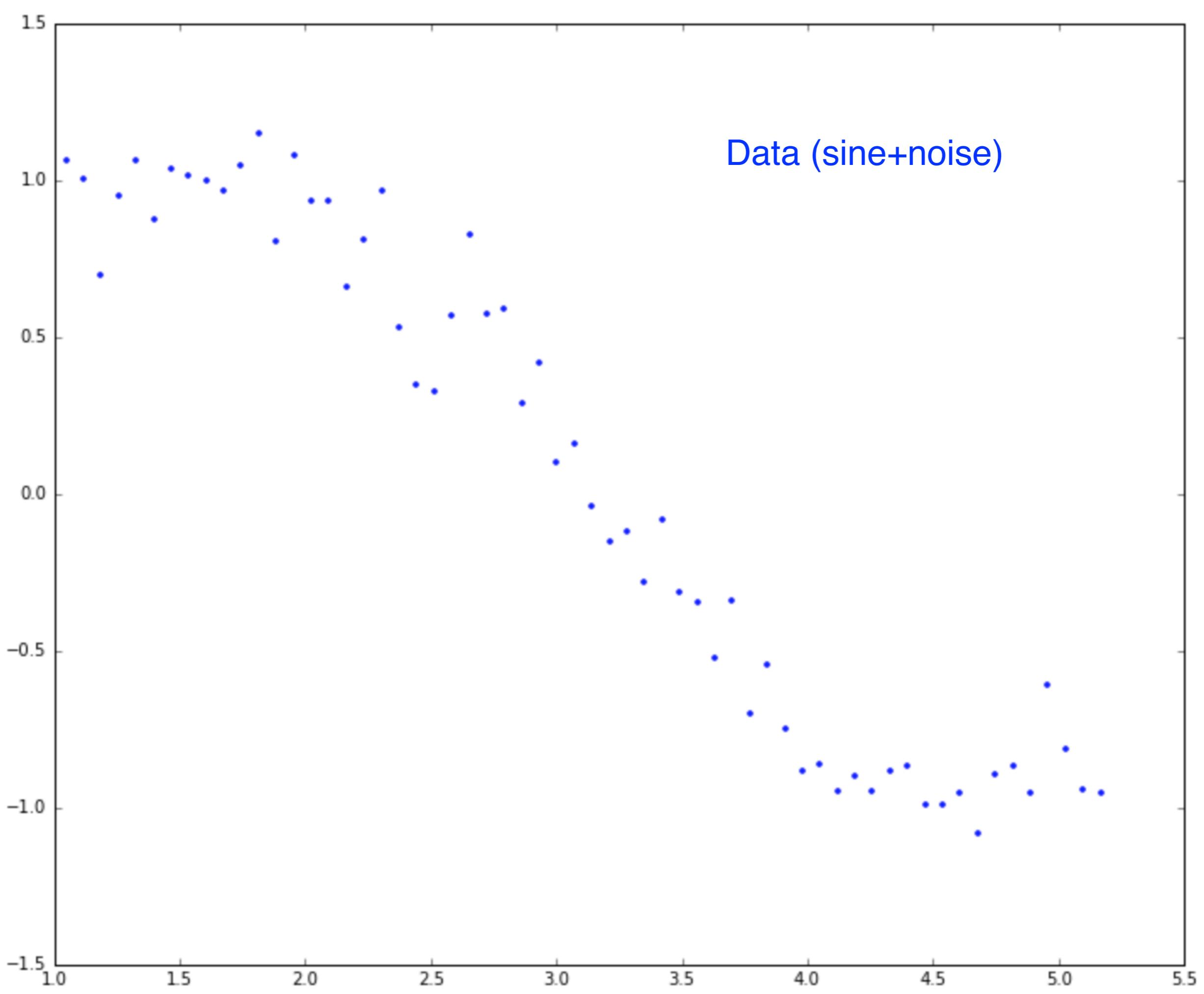
$$|w_1|^2 + |w_2|^2 < const$$

ssr + L2 penalty

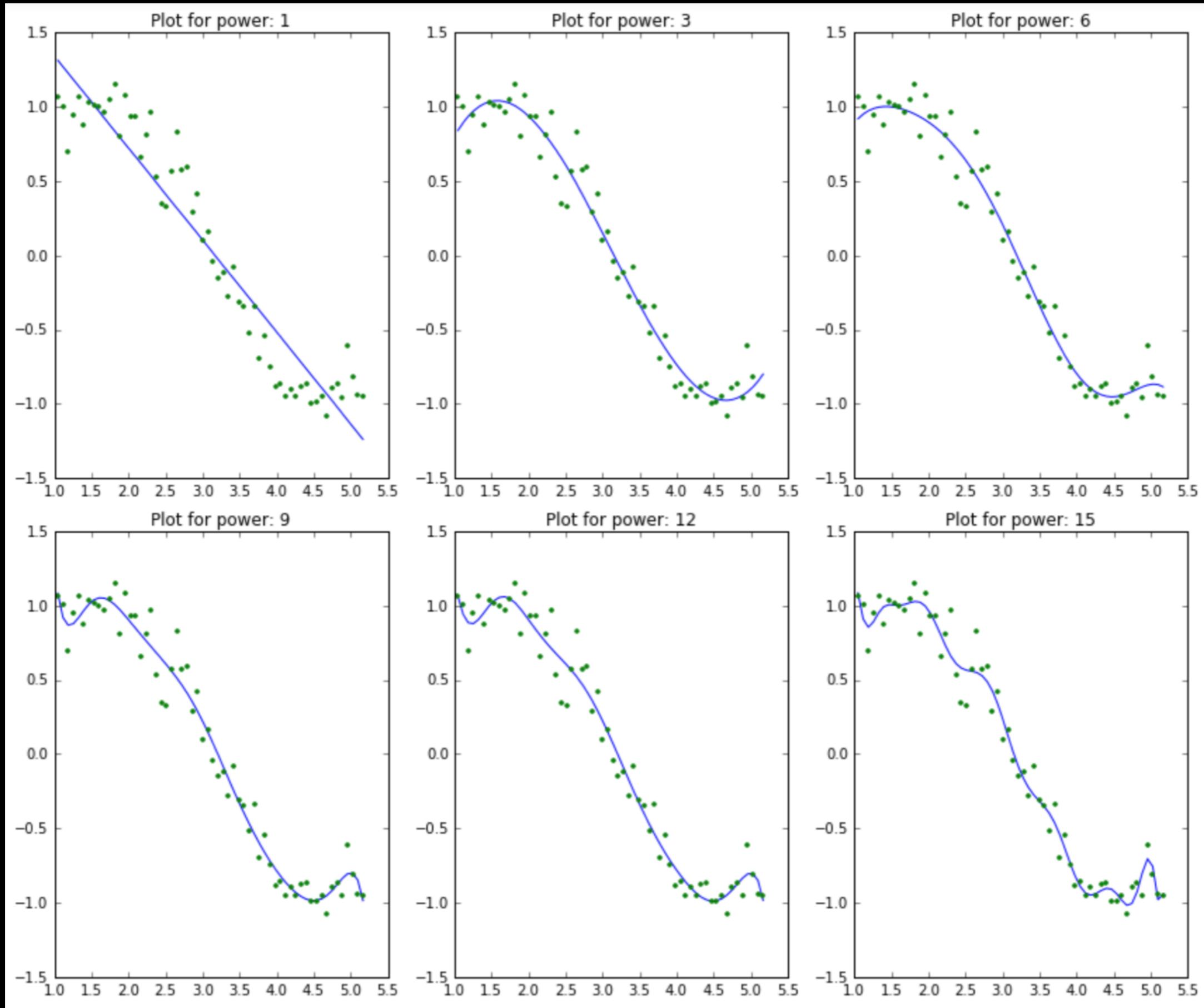
makes Elastic Net

$$\sum_{i=1}^N \left[y_i - \sum_{j=0}^M w_j x_{ij} \right]^2 + \lambda_1 \sum_{j=0}^M |w_j| + \lambda_2 \sum_{j=0}^M w_j^2$$

Elastic Net: Good for correlated parameters, combines both strengths, typically (λ_1, λ_2) via cross-validation



Fit: OLS Minimization for polynomials of order 1,3,6,9,12,15



Courtesy: K. Jain

Fit coefficients & Residuals

(OLS Minimization for polynomials of order 1-15)

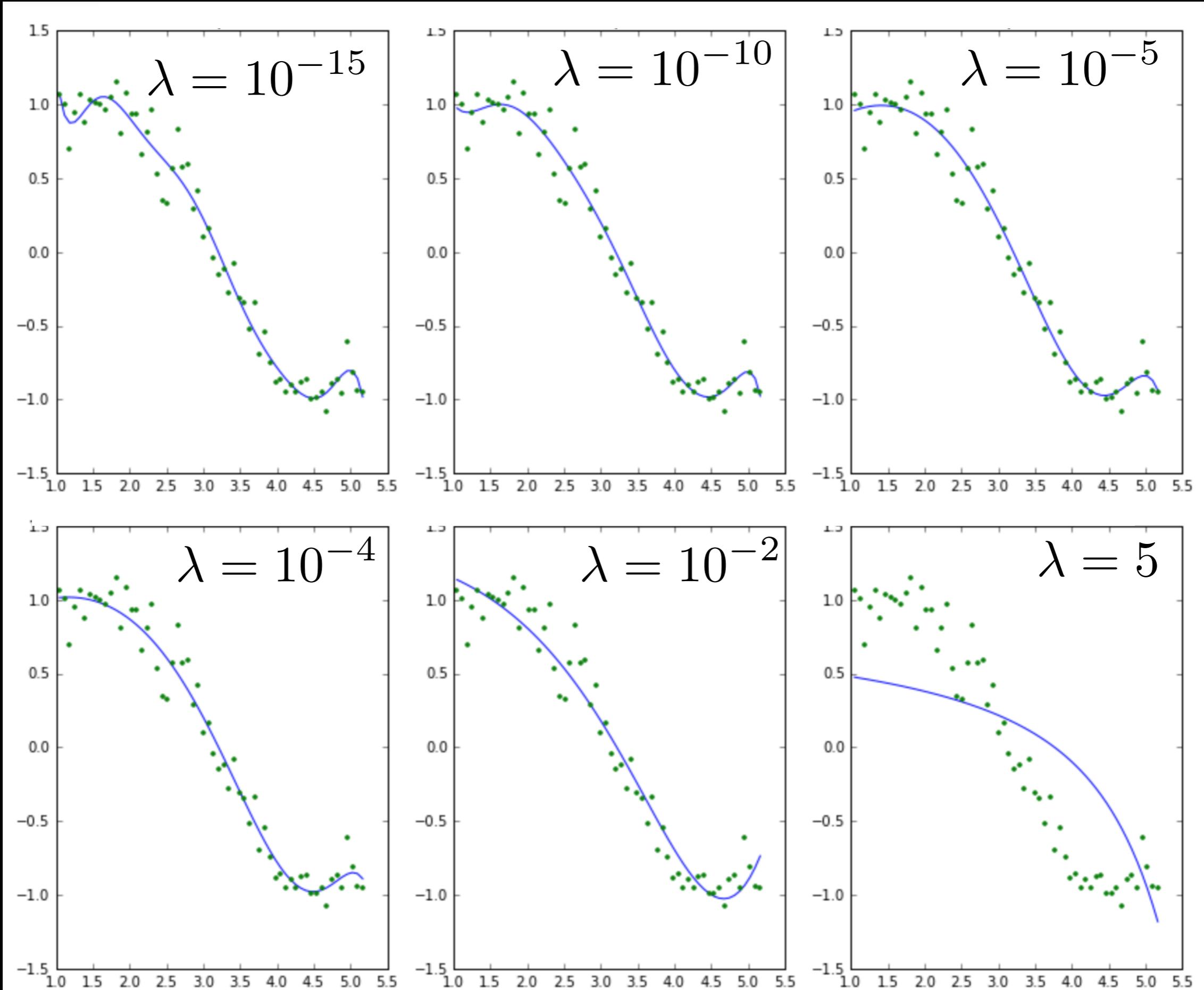
	rss	intercept	coef_x_1	coef_x_2	coef_x_3	coef_x_4	coef_x_5	coef_x_6	coef_x_7	coef_x_8	coef_x_9	coef_x_10	coef_x_11
model_pow_1	3.3	2	-0.62	NaN	NaN								
model_pow_2	3.3	1.9	-0.58	-0.006	NaN	NaN							
model_pow_3	1.1	-1.1	3	-1.3	0.14	NaN	NaN						
model_pow_4	1.1	-0.27	1.7	-0.53	-0.036	0.014	NaN	NaN	NaN	NaN	NaN	NaN	NaN
model_pow_5	1	3	-5.1	4.7	-1.9	0.33	-0.021	NaN	NaN	NaN	NaN	NaN	NaN
model_pow_6	0.99	-2.8	9.5	-9.7	5.2	-1.6	0.23	-0.014	NaN	NaN	NaN	NaN	NaN
model_pow_7	0.93	19	-56	69	-45	17	-3.5	0.4	-0.019	NaN	NaN	NaN	NaN
model_pow_8	0.92	43	-1.4e+02	1.8e+02	-1.3e+02	58	-15	2.4	-0.21	0.0077	NaN	NaN	NaN
model_pow_9	0.87	1.7e+02	-6.1e+02	9.6e+02	-8.5e+02	4.6e+02	-1.6e+02	37	-5.2	0.42	-0.015	NaN	NaN
model_pow_10	0.87	1.4e+02	-4.9e+02	7.3e+02	-6e+02	2.9e+02	-87	15	-0.81	-0.14	0.026	-0.0013	NaN
model_pow_11	0.87	-75	5.1e+02	-1.3e+03	1.9e+03	-1.6e+03	9.1e+02	-3.5e+02	91	-16	1.8	-0.12	0.0034
model_pow_12	0.87	-3.4e+02	1.9e+03	-4.4e+03	6e+03	-5.2e+03	3.1e+03	-1.3e+03	3.8e+02	-80	12	-1.1	0.062
model_pow_13	0.86	3.2e+03	-1.8e+04	4.5e+04	-6.7e+04	6.6e+04	-4.6e+04	2.3e+04	-8.5e+03	2.3e+03	-4.5e+02	62	-5.7
model_pow_14	0.79	2.4e+04	-1.4e+05	3.8e+05	-6.1e+05	6.6e+05	-5e+05	2.8e+05	-1.2e+05	3.7e+04	-8.5e+03	1.5e+03	-1.8e+02
model_pow_15	0.7	-3.6e+04	2.4e+05	-7.5e+05	1.4e+06	-1.7e+06	1.5e+06	-1e+06	5e+05	-1.9e+05	5.4e+04	-1.2e+04	1.9e+03

Coefficients increase exponentially with model complexity!

Regularization needed!

Courtesy: K. Jain

Ridge regression (regularization)



Courtesy: K. Jain

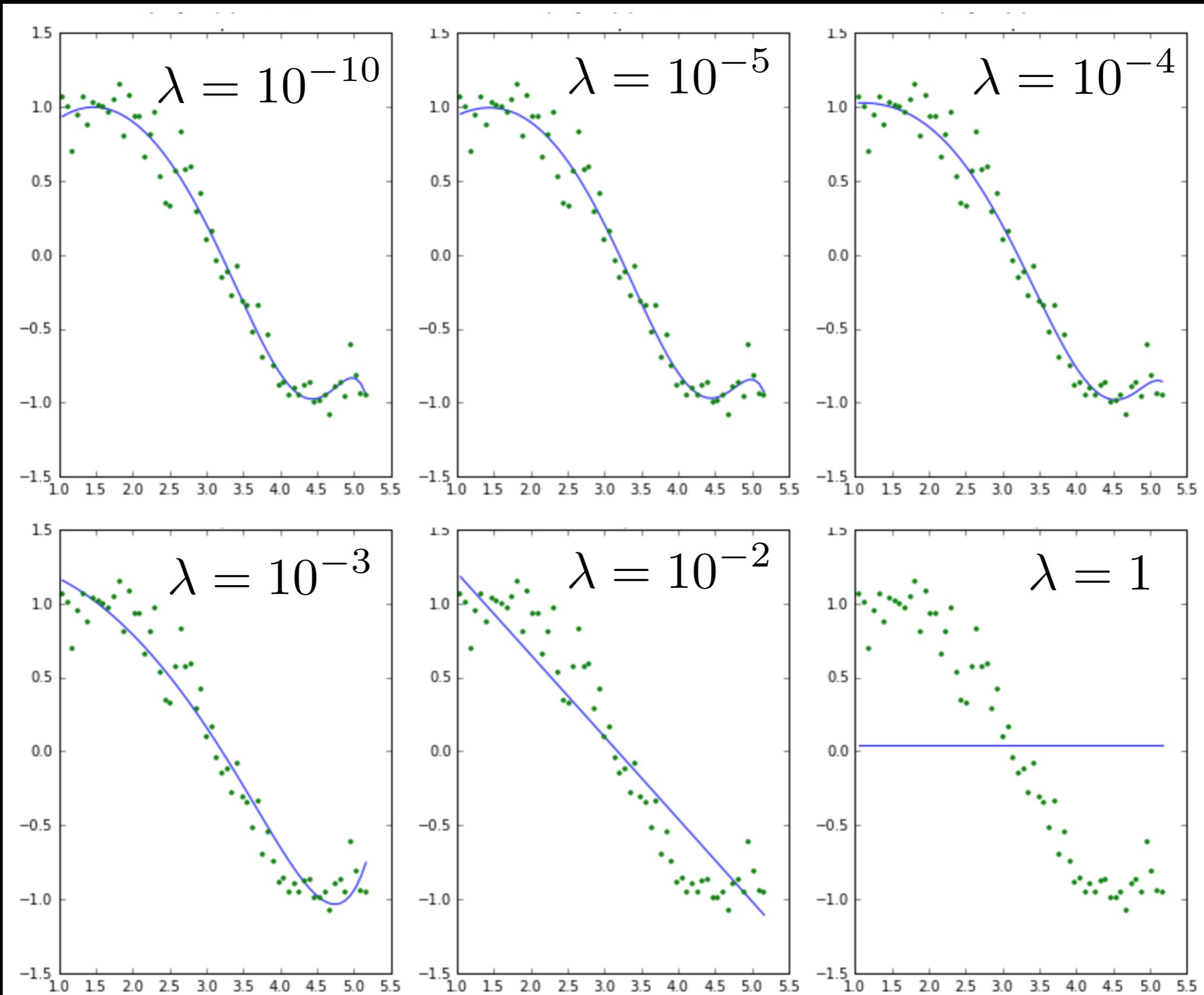
Fit coefficients & Residuals

(OLS Minimization with L2 penalty = Ridge regularization)

λ

	rss	intercept	coef_x_1	coef_x_2	coef_x_3	coef_x_4	coef_x_5	coef_x_6	coef_x_7	coef_x_8	coef_x_9	coef_x_10	coef_x_11	co
alpha_1e-15	0.87	95	-3e+02	3.8e+02	-2.4e+02	66	0.96	-4.8	0.64	0.15	-0.026	-0.0054	0.00086	0.
alpha_1e-10	0.92	11	-29	31	-15	2.9	0.17	-0.091	-0.011	0.002	0.00064	2.4e-05	-2e-05	-4
alpha_1e-08	0.95	1.3	-1.5	1.7	-0.68	0.039	0.016	0.00016	-0.00036	-5.4e-05	-2.9e-07	1.1e-06	1.9e-07	2e
alpha_0.0001	0.96	0.56	0.55	-0.13	-0.026	-0.0028	-0.00011	4.1e-05	1.5e-05	3.7e-06	7.4e-07	1.3e-07	1.9e-08	1.
alpha_0.001	1	0.82	0.31	-0.087	-0.02	-0.0028	-0.00022	1.8e-05	1.2e-05	3.4e-06	7.3e-07	1.3e-07	1.9e-08	1.
alpha_0.01	1.4	1.3	-0.088	-0.052	-0.01	-0.0014	-0.00013	7.2e-07	4.1e-06	1.3e-06	3e-07	5.6e-08	9e-09	1.
alpha_1	5.6	0.97	-0.14	-0.019	-0.003	-0.00047	-7e-05	-9.9e-06	-1.3e-06	-1.4e-07	-9.3e-09	1.3e-09	7.8e-10	2.
alpha_5	14	0.55	-0.059	-0.0085	-0.0014	-0.00024	-4.1e-05	-6.9e-06	-1.1e-06	-1.9e-07	-3.1e-08	-5.1e-09	-8.2e-10	-1
alpha_10	18	0.4	-0.037	-0.0055	-0.00095	-0.00017	-3e-05	-5.2e-06	-9.2e-07	-1.6e-07	-2.9e-08	-5.1e-09	-9.1e-10	-1
alpha_20	23	0.28	-0.022	-0.0034	-0.0006	-0.00011	-2e-05	-3.6e-06	-6.6e-07	-1.2e-07	-2.2e-08	-4e-09	-7.5e-10	-1

LASSO regression (regularization)



Courtesy: K. Jain

Fit coefficients & Residuals

(OLS Minimization with L1 penalty = **LASSO** regularization)

λ

	rss	intercept	coef_x_1	coef_x_2	coef_x_3	coef_x_4	coef_x_5	coef_x_6	coef_x_7	coef_x_8	coef_x_9	coef_x_10	coef_x_11	co
alpha_1e-15	0.96	0.22	1.1	-0.37	0.00089	0.0016	-0.00012	-6.4e-05	-6.3e-06	1.4e-06	7.8e-07	2.1e-07	4e-08	5.4
alpha_1e-10	0.96	0.22	1.1	-0.37	0.00088	0.0016	-0.00012	-6.4e-05	-6.3e-06	1.4e-06	7.8e-07	2.1e-07	4e-08	5.4
alpha_1e-08	0.96	0.22	1.1	-0.37	0.00077	0.0016	-0.00011	-6.4e-05	-6.3e-06	1.4e-06	7.8e-07	2.1e-07	4e-08	5.3
alpha_1e-05	0.96	0.5	0.6	-0.13	-0.038	-0	0	0	0	7.7e-06	1e-06	7.7e-08	0	0
alpha_0.0001	1	0.9	0.17	-0	-0.048	-0	-0	0	0	9.5e-06	5.1e-07	0	0	0
alpha_0.001	1.7	1.3	-0	-0.13	-0	-0	-0	0	0	0	0	0	1.5e-08	7.5
alpha_0.01	3.6	1.8	-0.55	-0.00056	-0	-0	-0	-0	-0	-0	-0	-0	0	0
alpha_1	37	0.038	-0	-0	-0	-0	-0	-0	-0	-0	-0	-0	-0	-0
alpha_5	37	0.038	-0	-0	-0	-0	-0	-0	-0	-0	-0	-0	-0	-0
alpha_10	37	0.038	-0	-0	-0	-0	-0	-0	-0	-0	-0	-0	-0	-0

HIGH SPARSITY

Regularization

LASSO regression is sensitive to correlation,
rather picks randomly few variables from
a larger group of highly correlated features

ElasticNet rather either leaves the whole group of
highly correlated features in (more stable model), or out

ElasticNet tends to select more variables,
typically leads to larger models,
more expensive to train, but more accurate

Markov Chain Monte Carlo (MCMC)

