ISTA 421 + INFO 521
Introduction to Machine Learning

Lecture 5: Basis Functions, Cross Validation (CV), Model Selection (by CV), Regularization

Clayton T. Morrison
claytonm@email.arizona.edu
Harvill 437A
Phone 621-6609

5 September 2018

SISTA 1

Next Topics

- Non-linear response:
  - Basis Functions
- Assessing Generalization
  - Problems: Under or Overfitting
  - Assessment framework: Cross validation
- Model selection
  - Method 1: Using Cross Validation
- Regularized Least Squares

- Probability Review
  - Definitions and Probability Calculus
  - Expectation
  - Continuous probability
  - Distributions
  - Likelihood

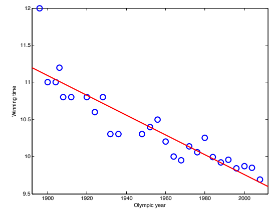
SISTA 2

The Normal Equations

For model: $t = f(x_1, \ldots, x_k; w_0, \ldots, w_k) = \sum_{i=0}^{k} x_i w_i$

$$w_0 = \bar{t} - w_1 x$$

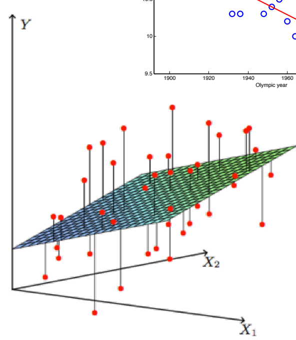
$$w_1 = \frac{\overline{xt} - \overline{x}\overline{t}}{\overline{x^2} - (\overline{x})^2}$$



$$\hat{\mathbf{w}} = \left(\mathbf{X}^\top \mathbf{X}\right)^{-1} \mathbf{X}^\top \mathbf{t}$$

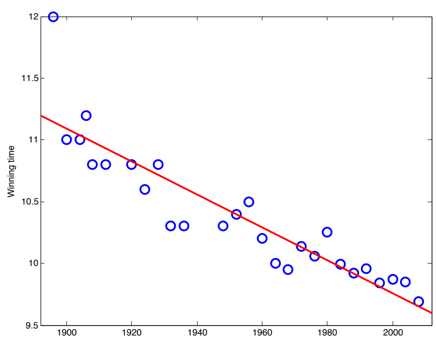
$$\mathbf{w} = \begin{bmatrix} w_0 \\ w_1 \end{bmatrix}, \quad \mathbf{x}_n = \begin{bmatrix} 1 \\ x_n \end{bmatrix}$$

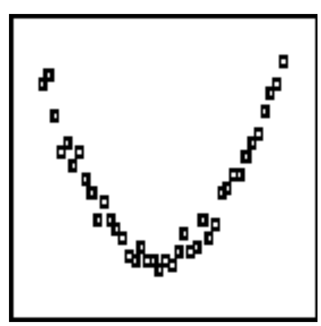
$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^\top \\ \mathbf{x}_2^\top \\ \vdots \\ \mathbf{x}_N^\top \end{bmatrix} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_N \end{bmatrix}, \quad \mathbf{t} = \begin{bmatrix} t_1 \\ t_2 \\ \vdots \\ t_N \end{bmatrix}$$



3

Linear (in response) has its limit!





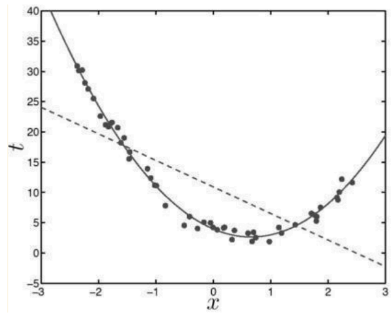
5

Nonlinear Response

- We can extend the power of linear LMS best fit to models that have a *non-linear* **response**.

$$f(x;\mathbf{w}) = \mathbf{w}^{\mathsf{T}}\mathbf{x} = w_0 + w_1 x + w_2 x^2$$

$$\mathbf{x}_n = \begin{bmatrix} 1 \\ x_n \\ x_n^2 \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ \vdots & \vdots & \vdots \\ 1 & x_N & x_N^2 \end{bmatrix}$$

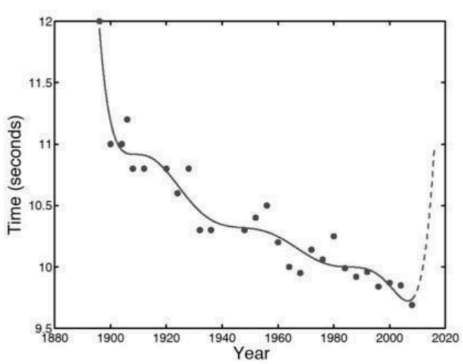


Fitting the parameters **w** still works the same! The only difference is that we square the *x* values *at the input phase* (for each of the elements of the third column vector)

SISTA 6

Generalize to Models of k^th-order Polynomials

$$f(x;\mathbf{w}) = \sum_{k=0}^{K} w_k x^k \qquad \mathbf{X} = \begin{bmatrix} x_1^0 & x_1^1 & x_1^2 & \cdots & x_1^K \\ x_2^0 & x_2^1 & x_2^2 & \cdots & x_2^K \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ x_N^0 & x_N^1 & x_N^2 & \cdots & x_N^K \end{bmatrix}$$



Note: this is **not** creating more ***independent*** sources of information about individuals, but it ***IS*** giving the model the capacity to consider **non-linear *components*** of what original inputs there are.

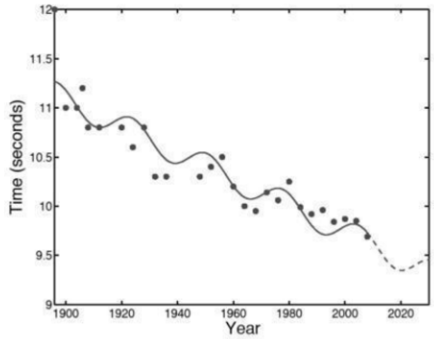
And we're still just learning **LINEAR COMBINATIONS** of those **components**

SISTA 7

Linear Combination of *Basis Functions* Projecting the Data (not just polynomials)

$$\mathbf{X} = \begin{bmatrix} h_1(x_1) & h_2(x_1) & \cdots & h_K(x_1) \\ h_1(x_2) & h_2(x_2) & \cdots & h_K(x_2) \\ \vdots & \vdots & \cdots & \vdots \\ h_1(x_N) & h_2(x_N) & \cdots & h_K(x_N) \end{bmatrix}$$

$$h_1(x) = 1$$
$$h_2(x) = x$$
$$h_3(x) = \sin\left(\frac{x-a}{b}\right)$$
$$f(x;\mathbf{w}) = w_0 + w_1 x + w_2 \sin\left(\frac{x-a}{b}\right).$$

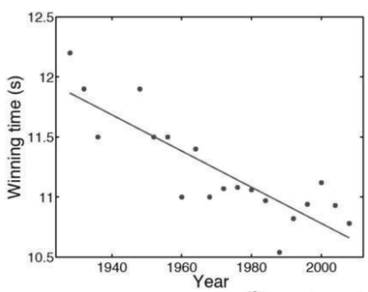


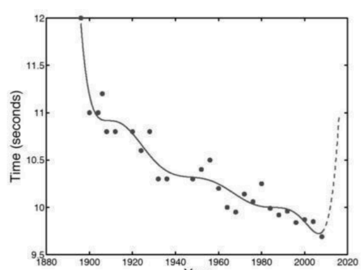
Careful !!
a and *b* must be **constants**

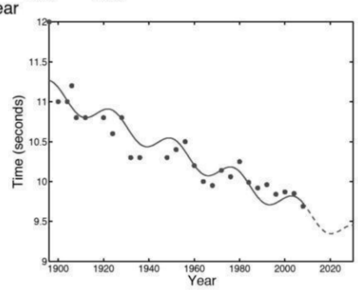
All *parameters* (as variables being adjusted) must be ***linearly*** combined

SISTA 8

Which Model is better: 1st order, 8th order ?



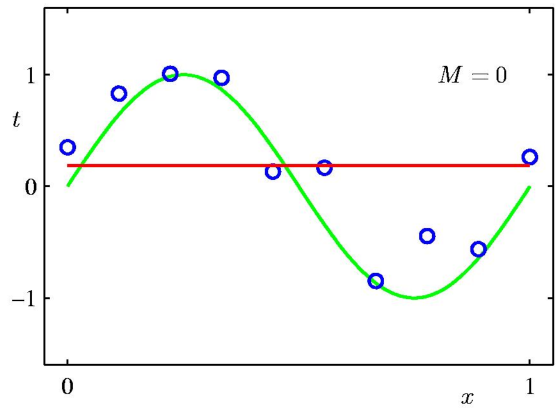




… periodic?

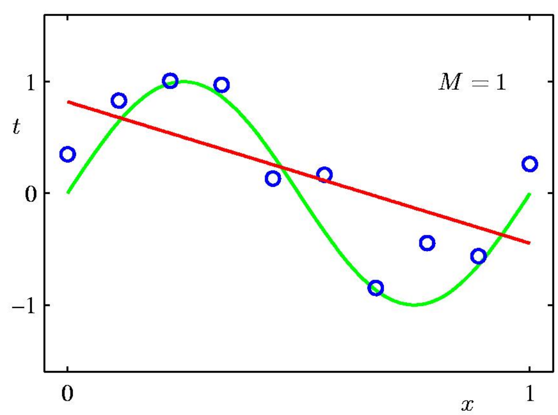
SISTA 9

0^{th} Order Polynomial

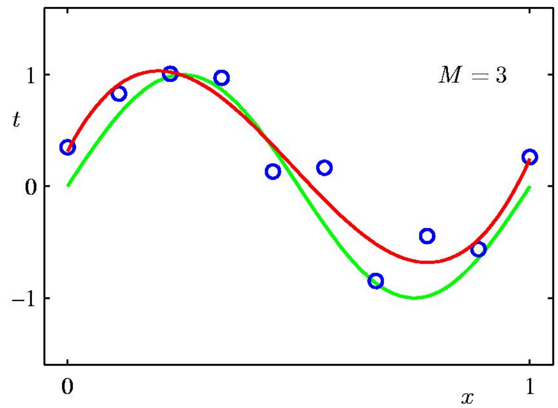


Underfitting

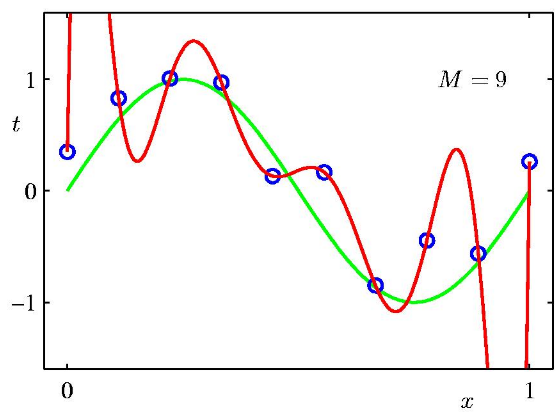
1^{st} Order Polynomial



3rd Order Polynomial



9th Order Polynomial

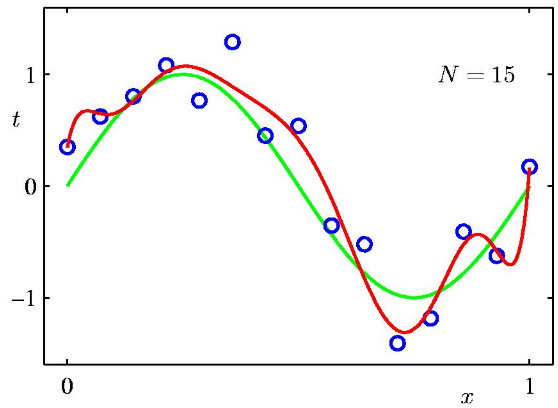


Overfitting

Data Set Size:

$N = 15$

9th Order Polynomial



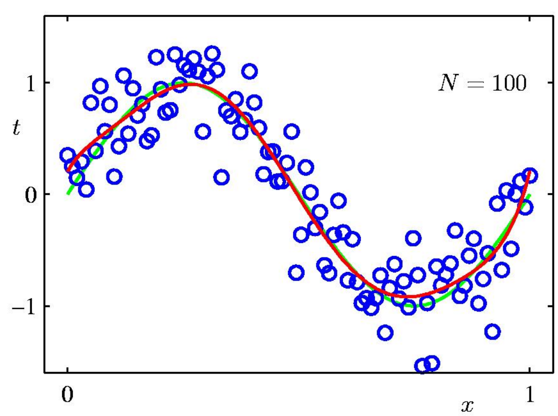
More data helps constrain the model best fit

SISTA 14

Data Set Size:

$N = 100$

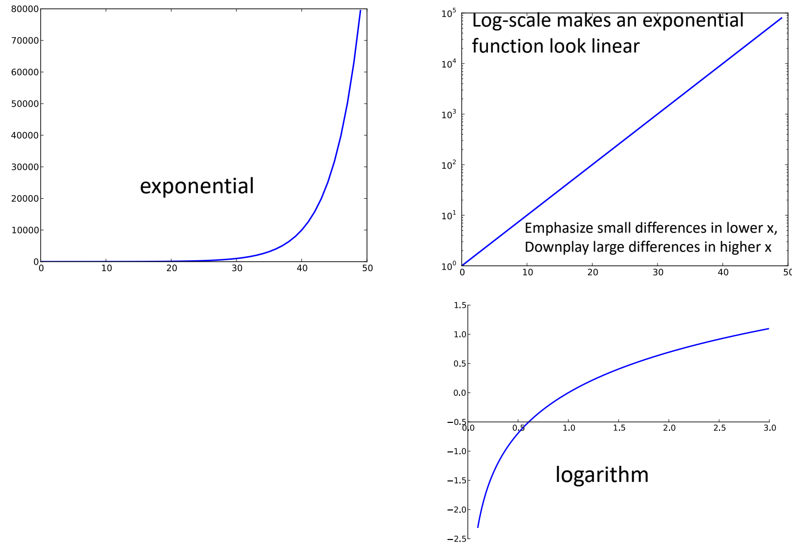
9th Order Polynomial



More data helps constrain the model best fit

SISTA 15

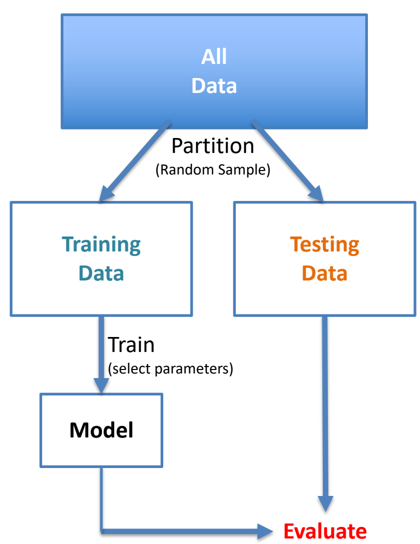
Sidenote: Log scale



16

Training versus Testing

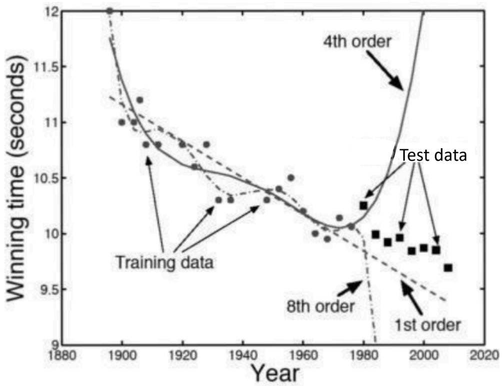
- The **Test Set** *CANNOT* influence *any* decisions about model parameter choices.
- The experimenter/designer (i.e., you) should **not** look at the Test Set until test (evaluation) time (after training).

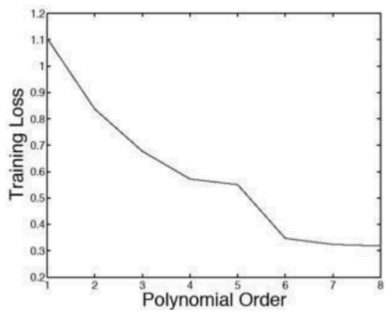


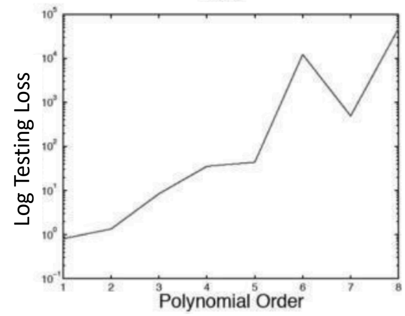
17

Training versus Testing

- The **Test Set** *CANNOT* influence *any* decisions about model parameter choices.
- The experimenter/designer (i.e., you) should **not** look at the Test Set until test (evaluation) time (after training).





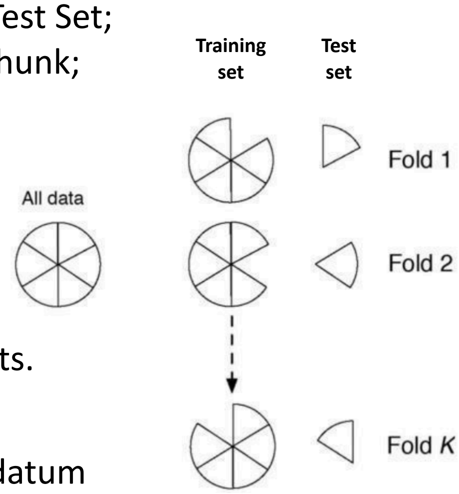


Cross-Validation

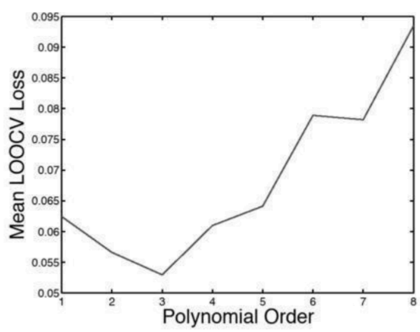
Randomly partition data into *k* chunks of (approx.) equal size;
"hold out" one chunk as the Test Set;
train on everything but that chunk;
test with the chunk.
Repeat this for all chunks.

What this does:
Estimates the error
of a number of possible
models trained on data subsets.

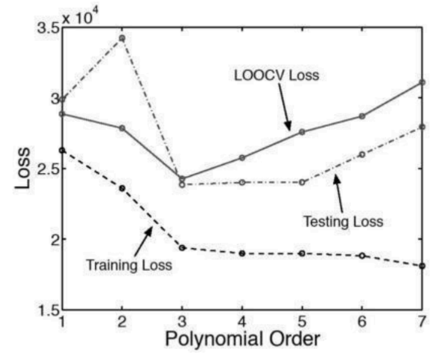
Leave-one-out-CV (LOOCV)
... same thing, but chunk = 1 datum



Model Selection: Using CV



On Men's 100 meter data
Trying different orders of polynomials
for the models

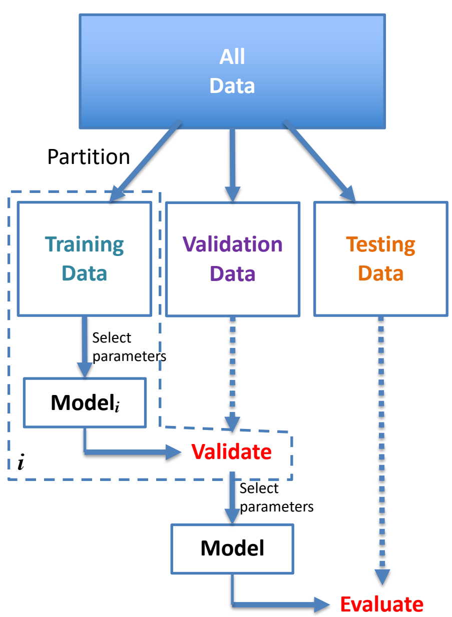


Study with artificial data (3rd order poly)
Sample size: 50
Test error based on 1000 indep samples

SISTA 20

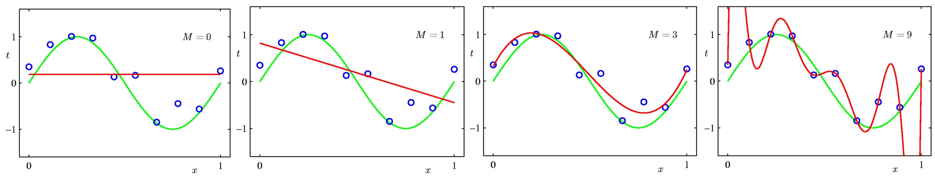
Validation Set

- In some cases, you need to select additional parameters based on trained model.
  - E.g., feature set selection
- In this case, need an additional, *independent* validation set.
- Logic is the same:
  - Training must be independent of validation
  - Validation must be independent of Test
- This could be "embedded" in a cross-validation framework



SISTA 21

Polynomial Coefficients



	$M=0$	$M=1$	$M=3$	$M=9$
$w_0^\star$	0.19	0.82	0.31	0.35
$w_1^\star$		-1.27	7.99	232.37
$w_2^\star$			-25.43	-5321.83
$w_3^\star$			17.37	48568.31
$w_4^\star$				-231639.30
$w_5^\star$				640042.26
$w_6^\star$				-1061800.52
$w_7^\star$				1042400.18
$w_8^\star$				-557682.99
$w_9^\star$				125201.43

Regularization

- Penalize large coefficient values: add magnitude of all of the weights (e.g., their sum) to the loss.

$$\sum_i w_i^2 = \mathbf{w}^\top \mathbf{w} \qquad \mathcal{L}' = \mathcal{L} + \lambda \mathbf{w}^\top \mathbf{w}$$

$$\begin{aligned}
\mathcal{L}' &= \mathcal{L} + \lambda \mathbf{w}^\top \mathbf{w} \\
&= \frac{1}{N}\mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w} - \frac{2}{N}\mathbf{w}^\top \mathbf{X}^\top \mathbf{t} + \lambda \mathbf{w}^\top \mathbf{w} \\
\frac{\partial \mathcal{L}'}{\partial \mathbf{w}} &= \frac{2}{N}\mathbf{X}^\top \mathbf{X}\mathbf{w} - \frac{2}{N}\mathbf{X}^\top \mathbf{t} + 2\lambda \mathbf{w}
\end{aligned}$$

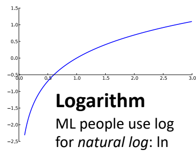
Note: We've already removed $\mathbf{t}^\top\mathbf{t}$ from $\mathcal{L}$ because we'll be taking the derivative with respect to **w**.

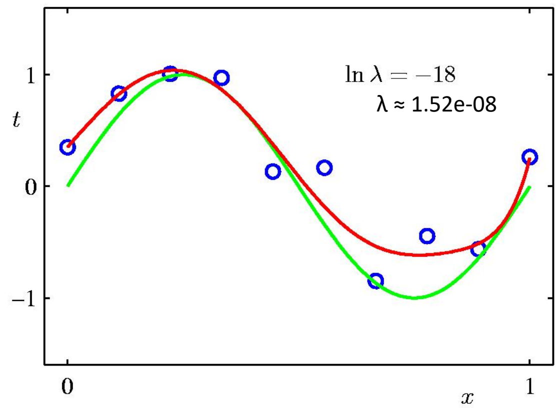
$$\begin{aligned}
\frac{2}{N}\mathbf{X}^\top \mathbf{X}\mathbf{w} - \frac{2}{N}\mathbf{X}^\top \mathbf{t} + 2\lambda \mathbf{w} &= 0 \\
\left(\mathbf{X}^\top \mathbf{X} + N\lambda \mathbf{I}\right)\mathbf{w} &= \mathbf{X}^\top \mathbf{t} \\
\hat{\mathbf{w}} &= \left(\mathbf{X}^\top \mathbf{X} + N\lambda \mathbf{I}\right)^{-1} \mathbf{X}^\top \mathbf{t}
\end{aligned}$$

Including a regularization term also ensures the inverse matrix is non-singular (which happens when $X^\top X$ has some columns that are colinear, or nearly so (leading to very large magnitude **w** values); near colinearity is not uncommon in real data).

Regularization:

$$\ln \lambda = -18$$



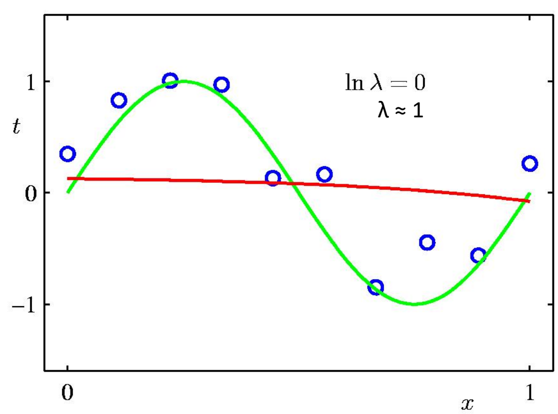


A good choice of lambda applies the right amount of regularization

24

Regularization:

$$\ln \lambda = 0$$

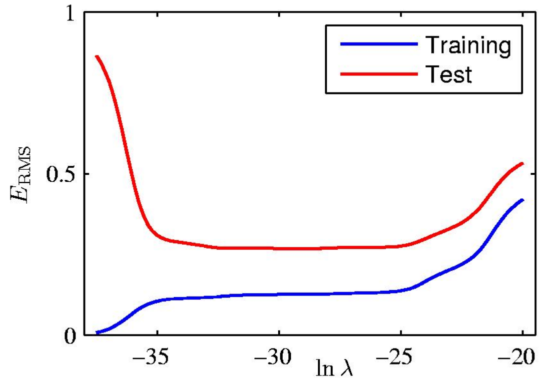


Lambda is too high: too much regularization, too little flexibility
(Not “enough” “weights resources” to apply to parameters)

25

Regularization:

E_{RMS} Vs. $\ln \lambda$



λ ≈ 6.3e-16 9.4e-14 1.4e-11 2.1e-9

"Grid search" for best lambda. The lowest independent "Test" performance is guide to choose best lambda. NOTE: This is using the "test" data to help select a parameter (so really like a "validation" set); need another independent test set to evaluation generalization error.

26

Polynomial Coefficients

	λ = 0 $\ln \lambda = -\infty$	λ = very small $\ln \lambda = -18$	λ = 1 $\ln \lambda = 0$
$w_0^\star$	0.35	0.35	0.13
$w_1^\star$	232.37	4.74	-0.05
$w_2^\star$	-5321.83	-0.77	-0.06
$w_3^\star$	48568.31	-31.97	-0.05
$w_4^\star$	-231639.30	-3.89	-0.03
$w_5^\star$	640042.26	55.28	-0.02
$w_6^\star$	-1061800.52	41.32	-0.01
$w_7^\star$	1042400.18	-45.95	-0.00
$w_8^\star$	-557682.99	-91.53	0.00
$w_9^\star$	125201.43	72.68	0.01

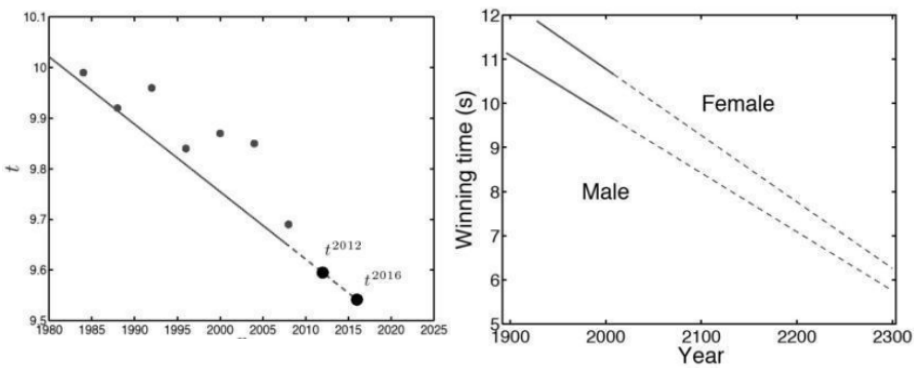
Under constrained (Under regularized)

Over constrained (Over regularized)

27

Predicting with a learned model

Prediction: $t_{new} = \hat{\mathbf{w}}^\top \mathbf{x}_{new} = \sum_{i=0}^{k} x_{new,i} w_i$



2592: look out boys! 3000: -3.5 seconds ??!