Connecting the dots with common sense and linear models

Léon Bottou

NEC Labs America

COS 424 – 2/4/2010

Introduction

Useful things:

– understanding probabilities,
– understanding statistical learning theory,
– knowing countless statistical procedures,
– knowing countless machine learning algorithms.

Essential things:

– applying common sense,
– paying attention to details,
– being able to setup experiments,
– and to measure the outcome of experiments,
– and to measure plenty of other things,

Connecting the dots

Question:

Find y given x.

x	y
0.31	1.87
0.25	1.84
3.78	2.23
3.30	3.04
3.83	2.68
-3.29	0.01
-0.90	0.37
-3.61	0.37
0.64	2.05
-0.34	0.96

...

Connecting the dots

Question:

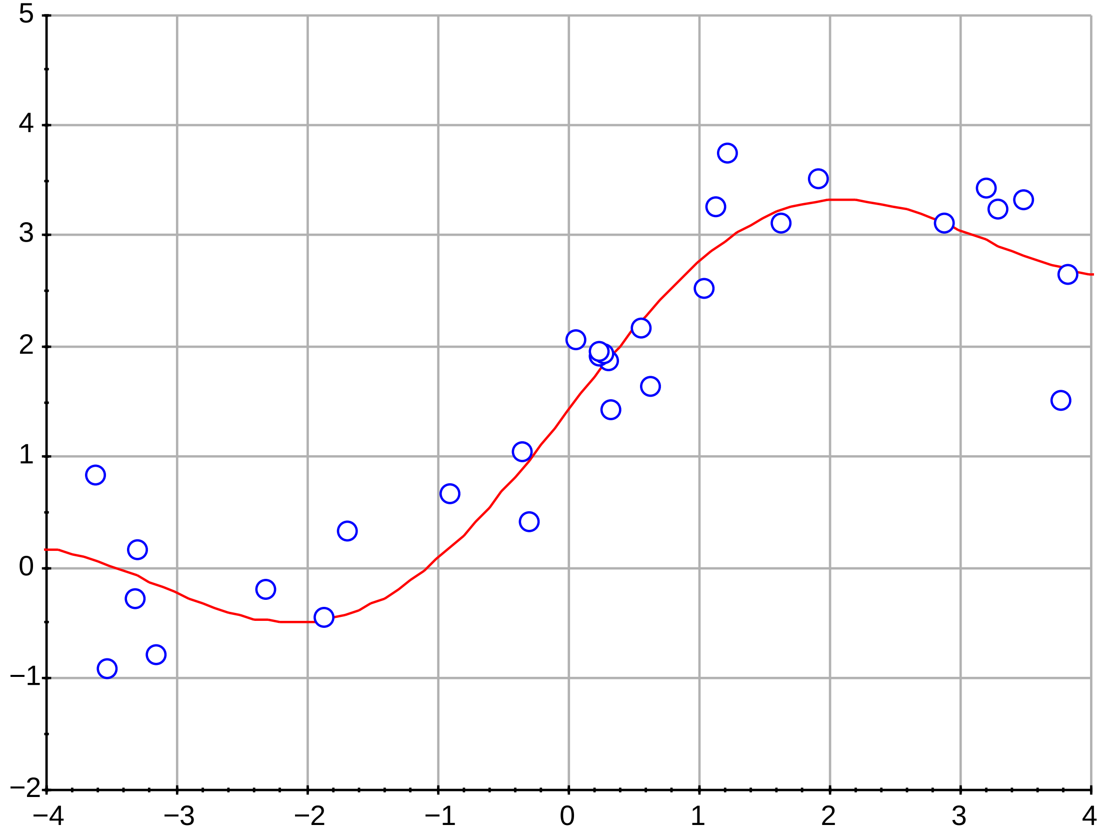
Find y given x.

x	y
0.31	1.87
0.25	1.84
3.78	2.23
3.30	3.04
3.83	2.68
-3.29	0.01
-0.90	0.37
-3.61	0.37
0.64	2.05
-0.34	0.96
-3.53	-0.35
1.63	3.18

. . . .

Answer:

Connect the dots. Read the curve.



Connecting the dots – take two

Question: Find $\mathbf{y}$ given $\mathbf{x}$.

$[\mathbf{x}]_1$	$[\mathbf{x}]_2$	$[\mathbf{x}]_3$	$[\mathbf{x}]_4$	$[\mathbf{x}]_5$	$[\mathbf{x}]_6$	$[\mathbf{x}]_7$	$[\mathbf{x}]_8$	...	$[\mathbf{x}]_{13,123}$	$[\mathbf{x}]_{13,124}$	$[\mathbf{x}]_{13,125}$	$\mathbf{y}$
0.39	0.50	5.84	-4.36	-0.01	7.20	-7.40	-7.16	...	-5.48	0.77	5.03	5.46
7.34	1.92	-5.66	-5.33	-6.15	-3.14	4.53	6.37	...	-2.30	6.45	5.10	5.18
2.27	4.57	4.18	-6.07	-5.47	-6.97	2.67	-3.93	...	2.77	7.46	4.84	6.97
1.09	-2.17	-6.38	5.66	-2.65	-2.81	-0.69	2.76	...	0.42	5.88	0.29	-7.13
2.85	1.79	6.22	1.34	-1.83	3.01	3.99	-1.75	...	0.03	1.55	-3.32	-5.42
-5.67	2.53	-3.47	-0.46	3.21	-2.73	6.65	-0.77	...	-1.41	-3.93	3.14	5.37
3.80	-0.00	1.89	3.24	2.30	-1.45	7.63	-2.12	...	6.47	2.04	3.58	-4.96
7.54	2.47	6.39	4.95	-2.51	-6.46	0.49	-0.61	...	5.10	1.90	1.79	3.20
-7.99	4.93	-2.13	-7.11	-5.10	2.13	6.31	7.00	...	1.71	-2.35	-7.87	-4.70
-6.80	7.33	-0.99	4.17	-7.81	-7.64	4.01	-3.37	...	7.29	-2.41	7.66	-6.70
-0.78	5.34	-5.94	-1.76	3.79	2.92	0.75	7.04	...	-3.87	-1.46	-3.37	-3.66
7.54	2.47	6.39	4.95	-2.51	-6.46	0.49	-0.61	...	5.10	1.90	1.79	3.20
-7.99	4.93	-2.13	-7.11	-5.10	2.13	6.31	7.00	...	1.71	-2.35	-7.87	-4.70
-6.80	7.33	-0.99	4.17	-7.81	-7.64	4.01	-3.37	...	7.29	-2.41	7.66	-6.70

. . . .

Idea: (1) understand how we do the 2D case. (2) generalize!

A Simple Linear Model

Polynomial: $f(x) = w_0 + w_1 x + w_2 x^2 + \cdots + w_n x^n$

Slight generalization:

$$x \quad \longrightarrow \quad \Phi(x) = \begin{bmatrix} \phi_0(x) \\ \phi_1(x) \\ \cdots \\ \phi_n(x) \end{bmatrix} \quad \longrightarrow \quad f(x) = [w_0, w_1, \ldots, w_n] \times \begin{bmatrix} \phi_0(x) \\ \phi_1(x) \\ \cdots \\ \phi_n(x) \end{bmatrix}$$

Equivalently: $f(x) = w^\top \Phi(x)$

Lets choose a basis Φ and use the data to determine w.

Linear Least Squares

Input : x_i
Output : $w^\top \Phi(x_i)$
Desired Output : y_i

Difference : $y_i - w^\top \Phi(x_i)$

Minimize : $C(w) = \sum_{i=1}^{n} \left(y_i - w^\top \Phi(x_i) \right)^2$

Quadratic convex function in w.
The minimum exists and is unique.
But it could be reached for multiple values of w.

A little bit of Linear Algebra

At the optimum, $$\frac{dC}{dw} = \sum_{i=1}^{n} 2 \left(y_i - w^\top \Phi(x_i) \right) \Phi(x_i)^\top = 0$$

Therefore we must solve the system of equations :

$$\left[\sum_{i=1}^{n} \Phi(x_i) \Phi(x_i)^\top \right] \times w \; = \; \left[\sum_{i=1}^{n} y_i \Phi(x_i) \right]$$

Shorthand form : $$\boxed{\left(X^\top X \right) w = \left(X^\top Y \right)}$$

Singularities

Almost the same as $\boldsymbol{w} = (\,\boldsymbol{X}^\top \boldsymbol{X}\,)^{-1}\,(\,\boldsymbol{X}^\top \boldsymbol{Y}\,)$.

You should never solve a system by inverting a matrix.

Who said $X^\top X$ is invertible?

Consider the case where $\phi_1(x) = \phi_8(x)$
– the matrix $X^\top X$ is singular.
– but the minimum is unchanged.
– the minimum is reached by many $\boldsymbol{w}$,
as long as $w_1 + w_8$ remains constant.

Among the $\boldsymbol{w}$ that minimize $\boldsymbol{C}(\boldsymbol{w})$,
compute the one with the smallest norm.

Numerical Procedures

Diagonalization of $X^\top X$

$$Q^\top D\, Q\, w = X^\top Y \quad \Longleftarrow \quad w = Q^\top D^+\, Q\, X^\top Y$$

Traditional methods: SVD or QR decomposition of X

$$V\, D\, U^\top\, U\, D\, V^\top\, w = V\, D\, U^\top\, Y \quad \Longleftarrow \quad w = V\, D^+\, U^\top Y$$

$$R^\top Q^\top Q\, R\, w = R^\top Q^\top Y \quad \Longleftarrow \quad R\, w = Q^\top Y$$

and solve using back-substitution.

Simple and Fast: Regularization + Cholevsky

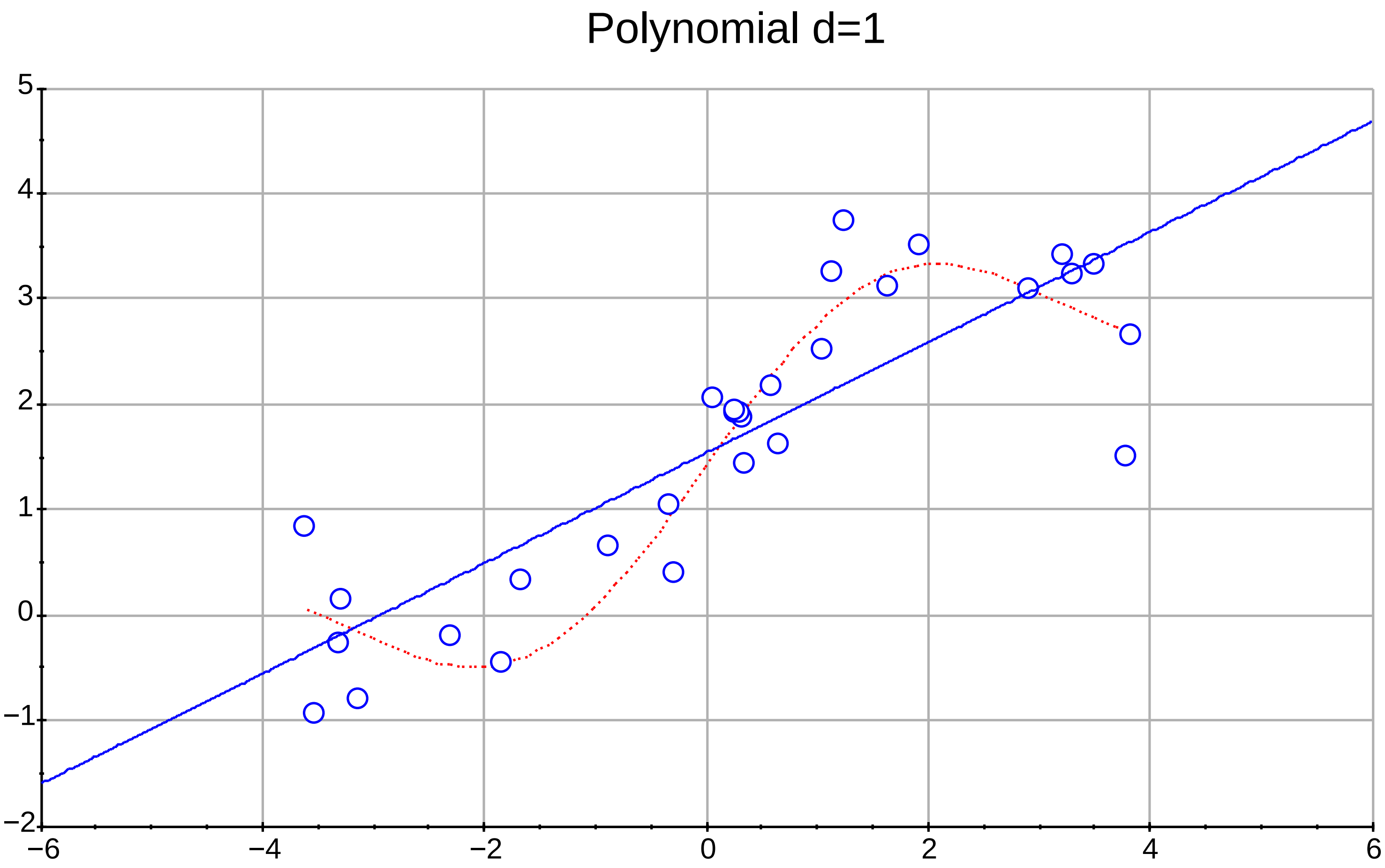
$$\min\ C(w) + \varepsilon w^2 \quad \Longleftrightarrow \quad (\, X^\top X + \varepsilon I\,)\ w = (\, X^\top Y\,)$$

$$\Longleftrightarrow \quad U\ U^\top w = (\, X^\top Y\,)$$

and solve using two rounds of back-substitution.

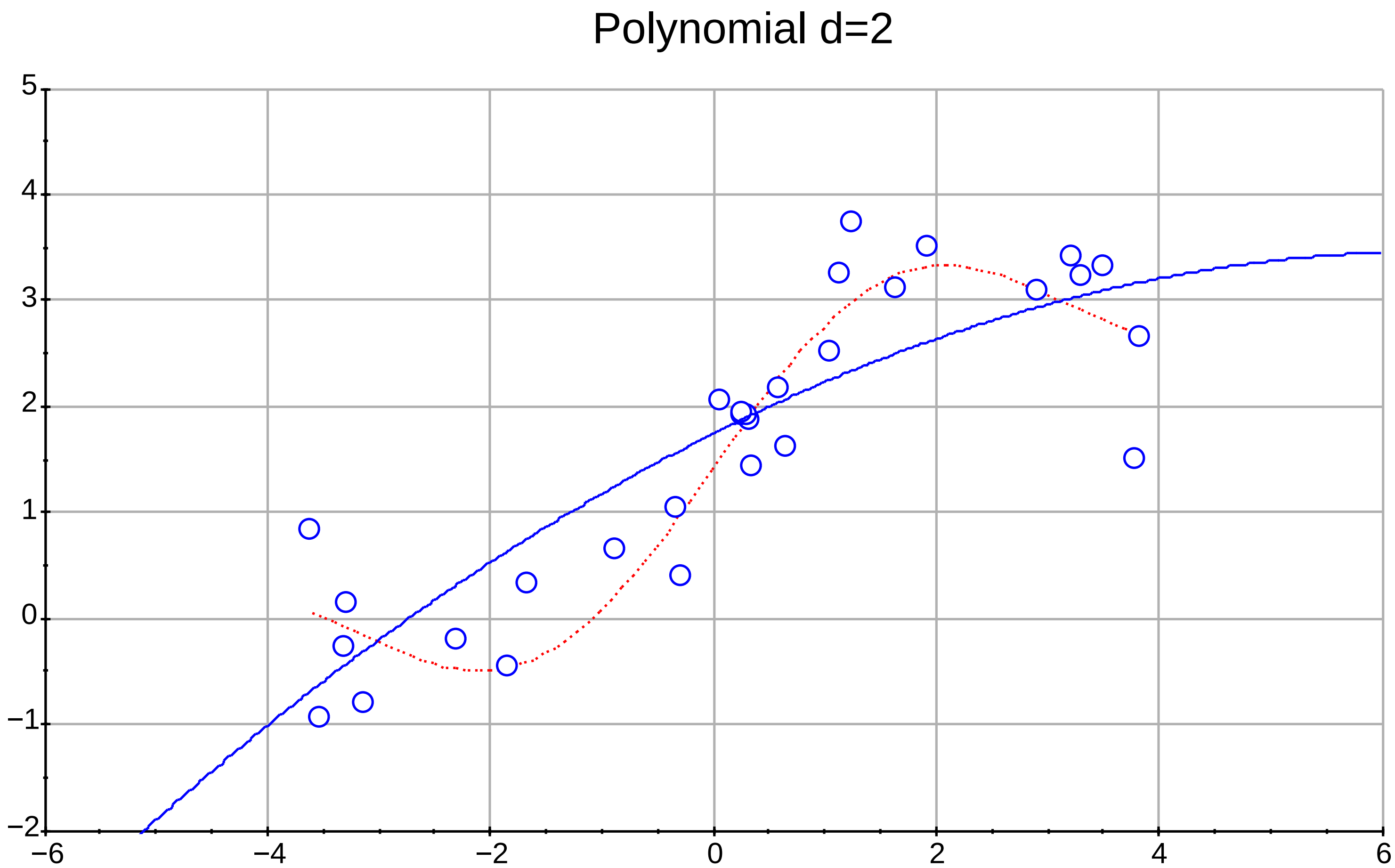
Polynomial degree 1

$\Phi(x) = 1, x$



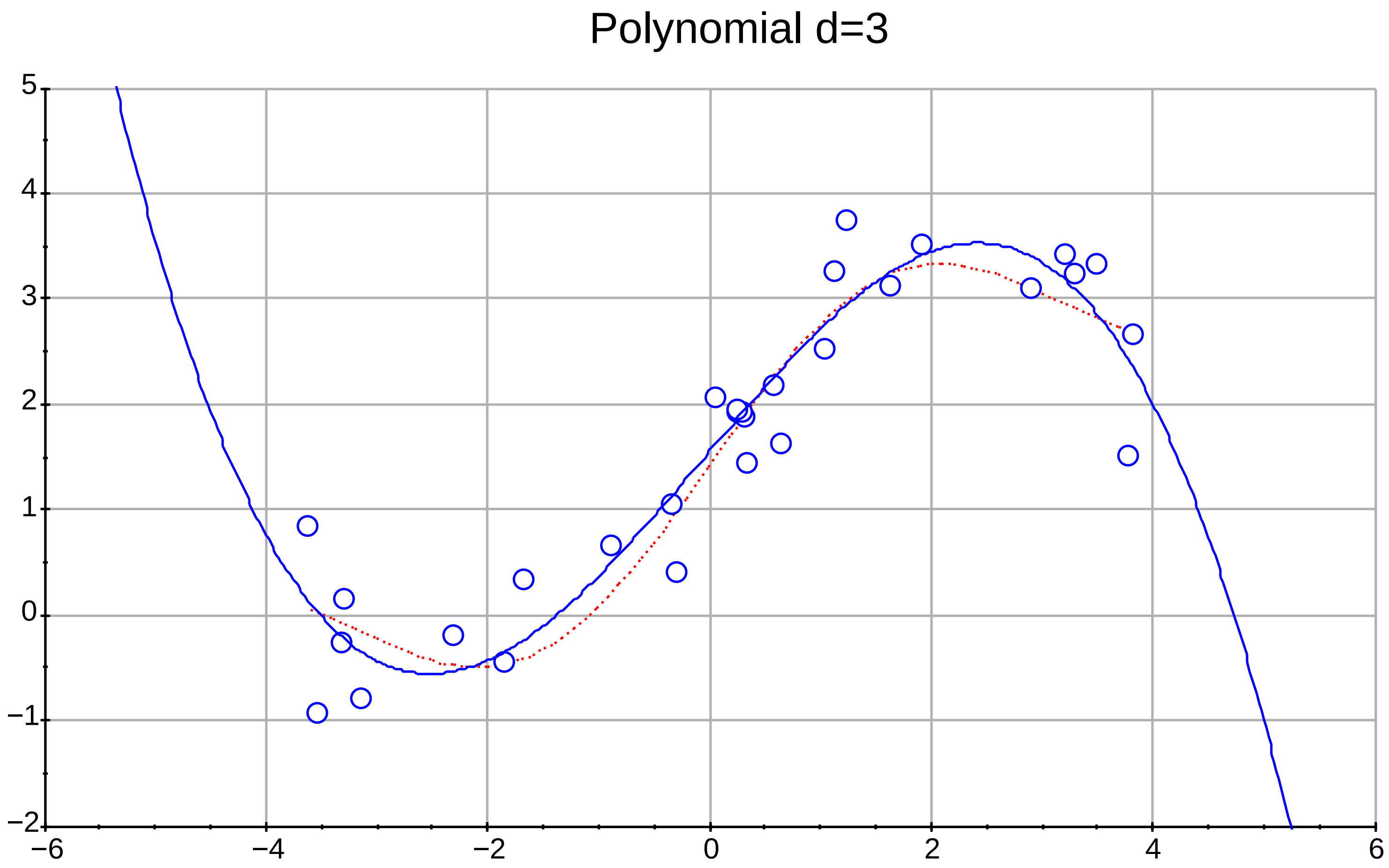
Polynomial degree 2

$$\Phi(x) = 1,\ x,\ x^2$$



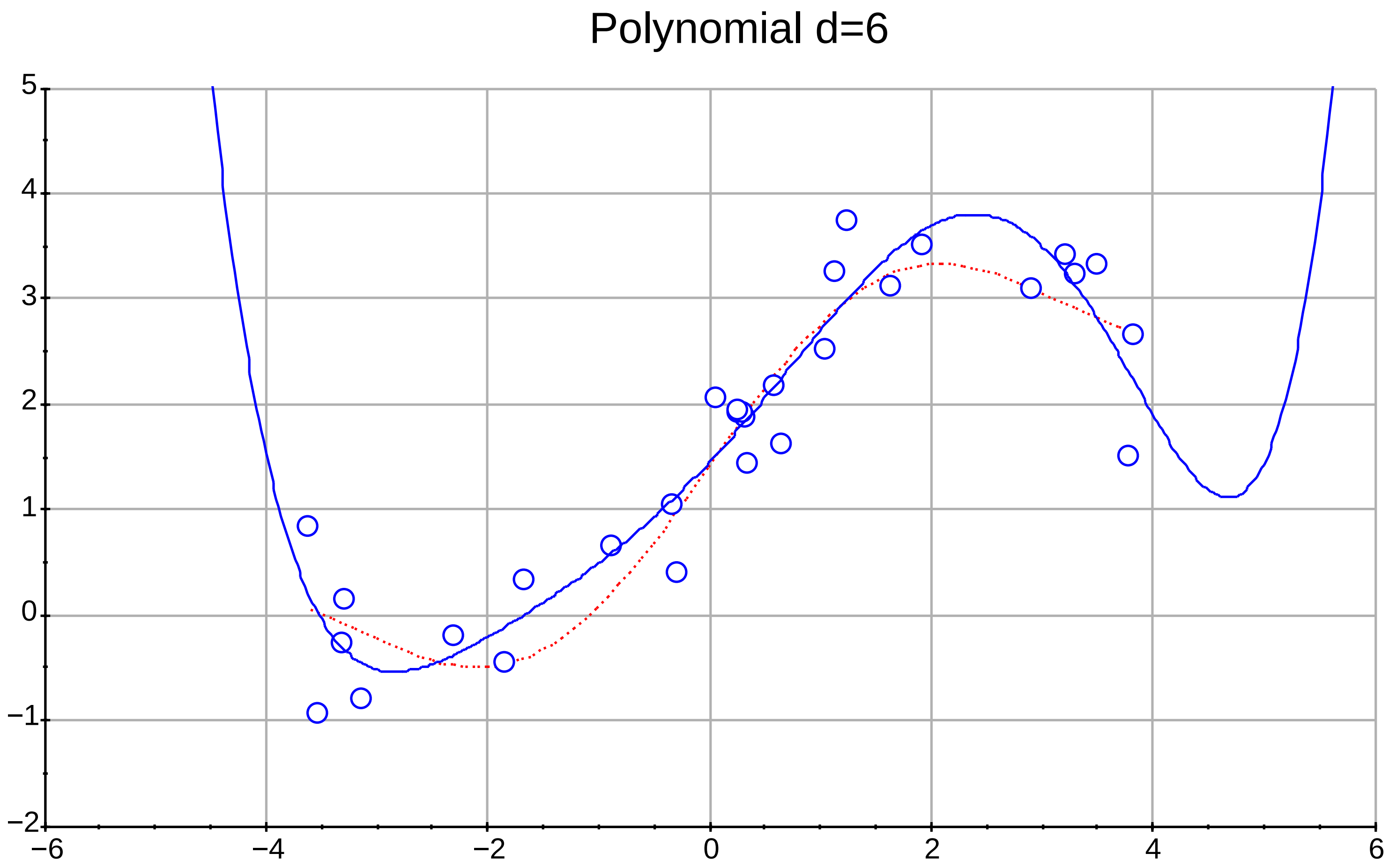
Polynomial degree 3

$$\Phi(x) = 1,\ x,\ x^2,\ x^3$$



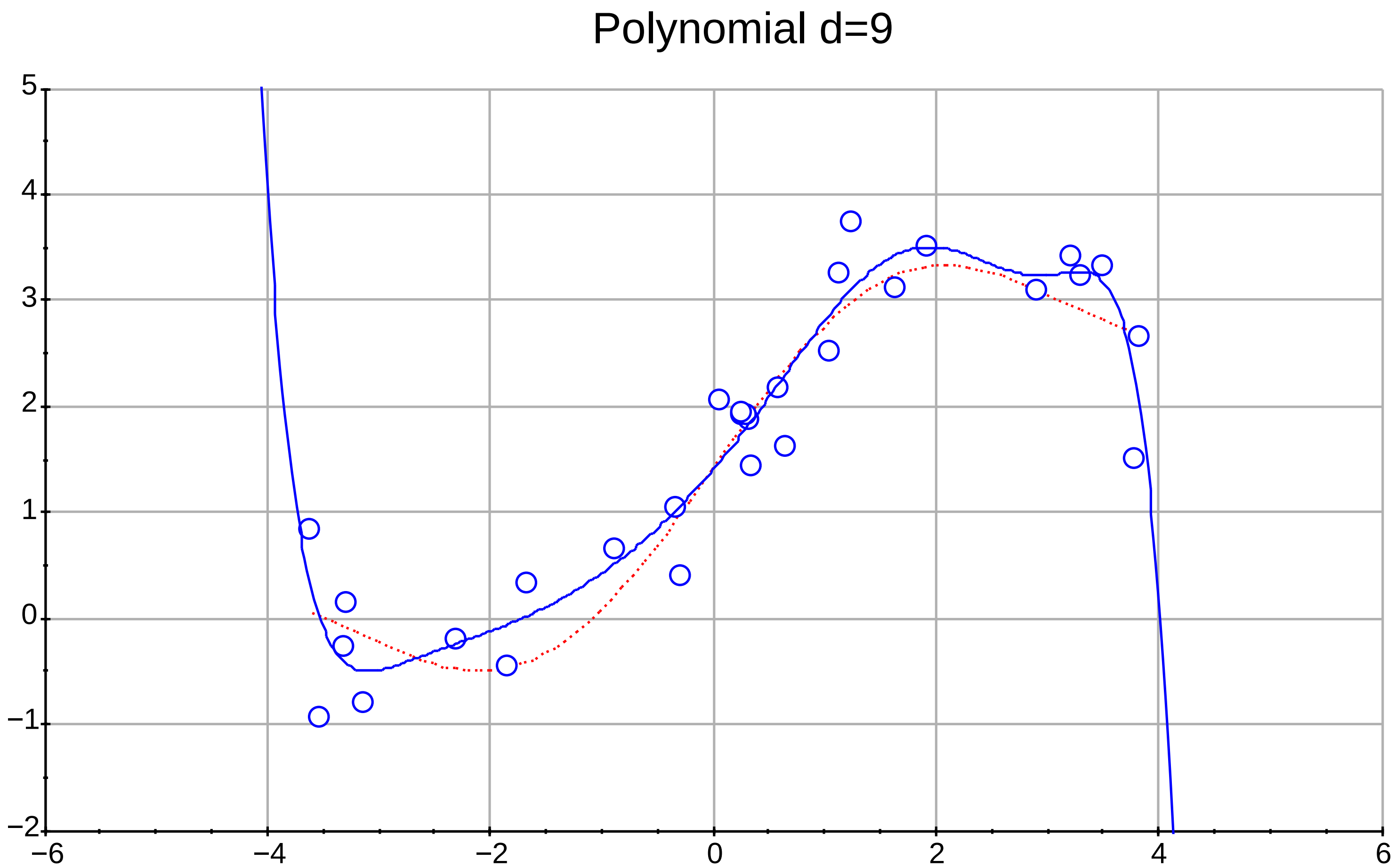
Polynomial degree 6

$$\Phi(x) = 1,\ x,\ x^2,\ x^3,\ x^4,\ x^5,\ x^6$$



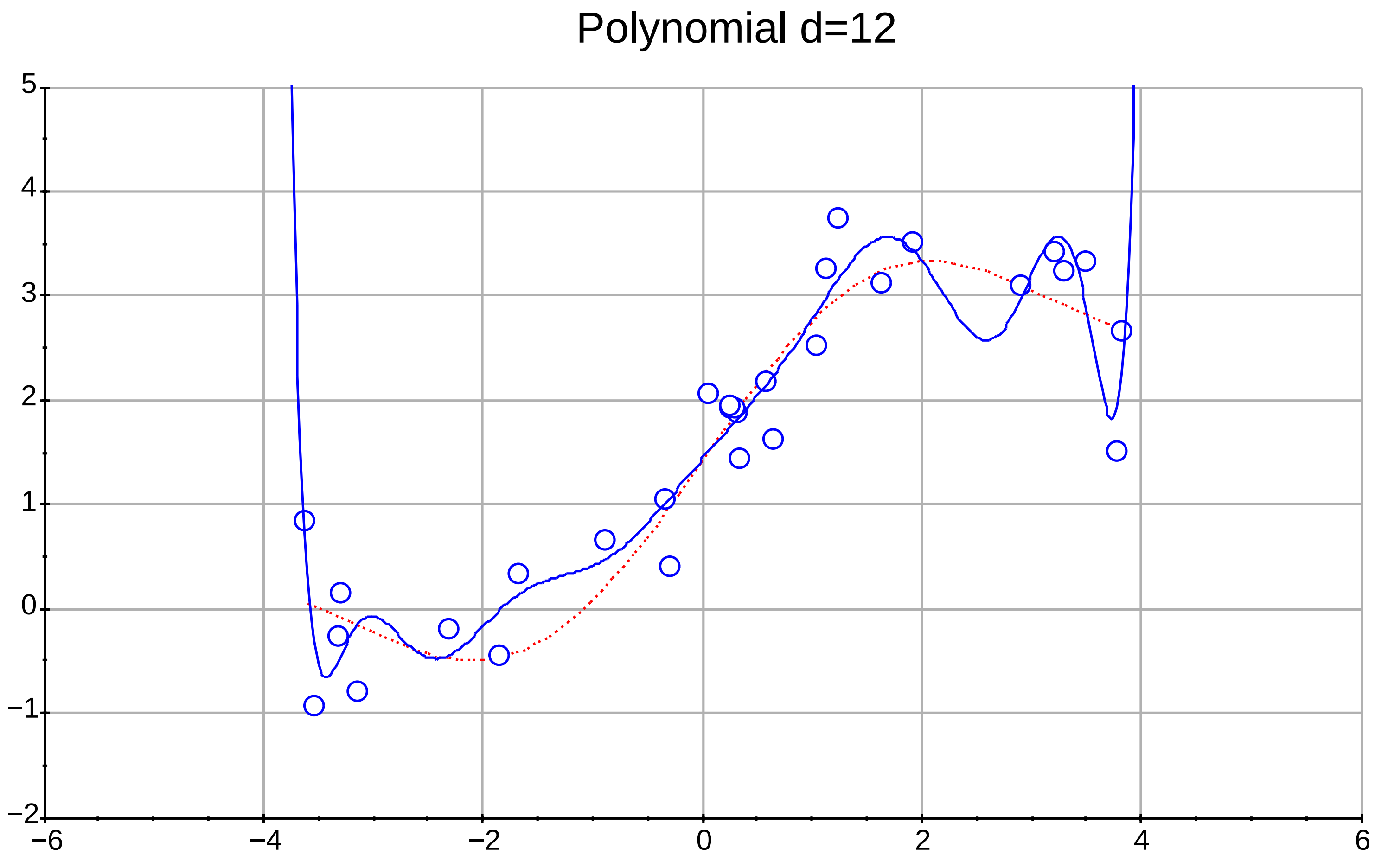
Polynomial degree 9

$\Phi(x) \;=\; 1,\; x,\; x^2, \ldots,\; x^9$



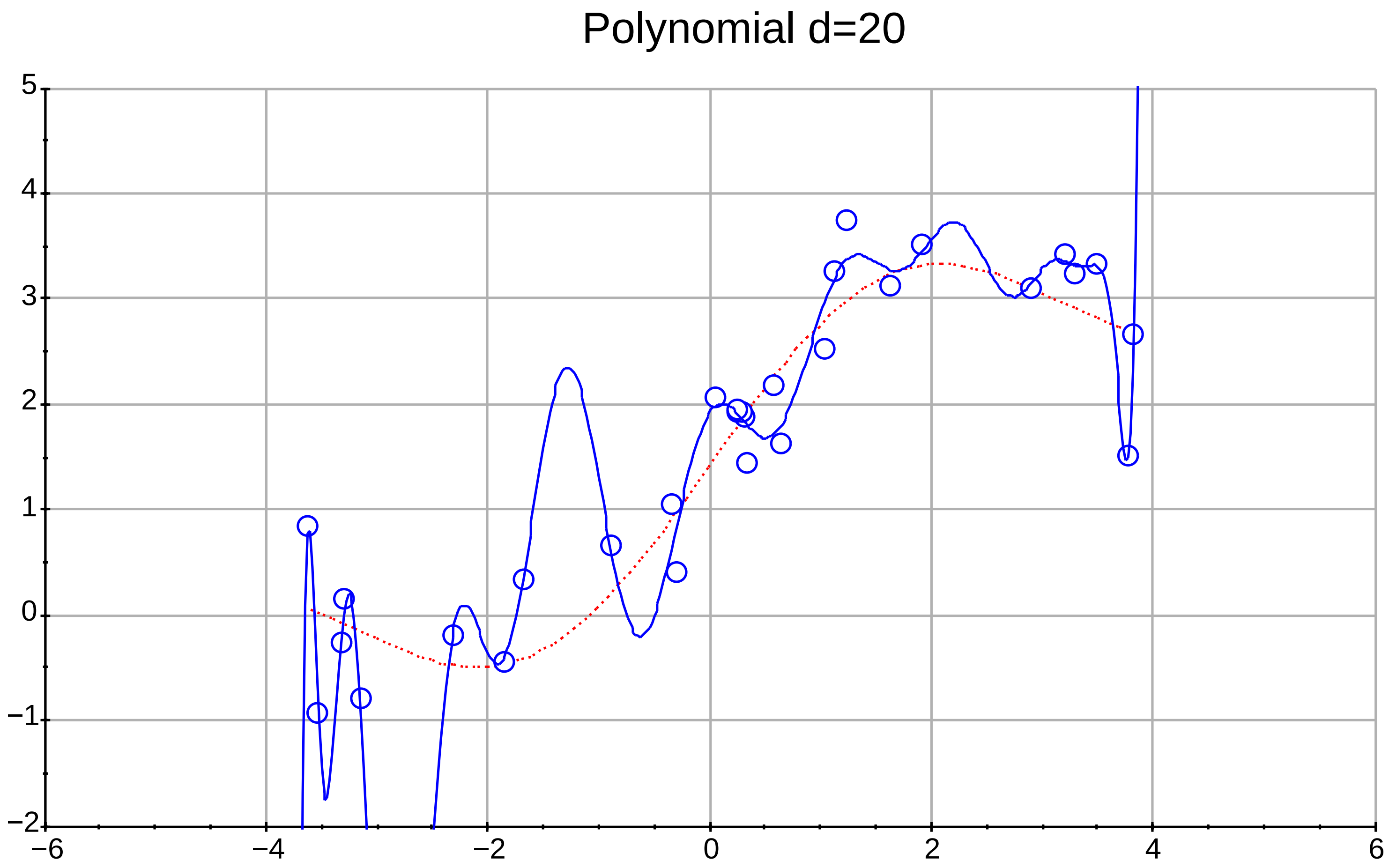
Polynomial degree 12

$$\Phi(x) \;=\; 1,\; x,\; x^2, \ldots,\; x^{12}$$

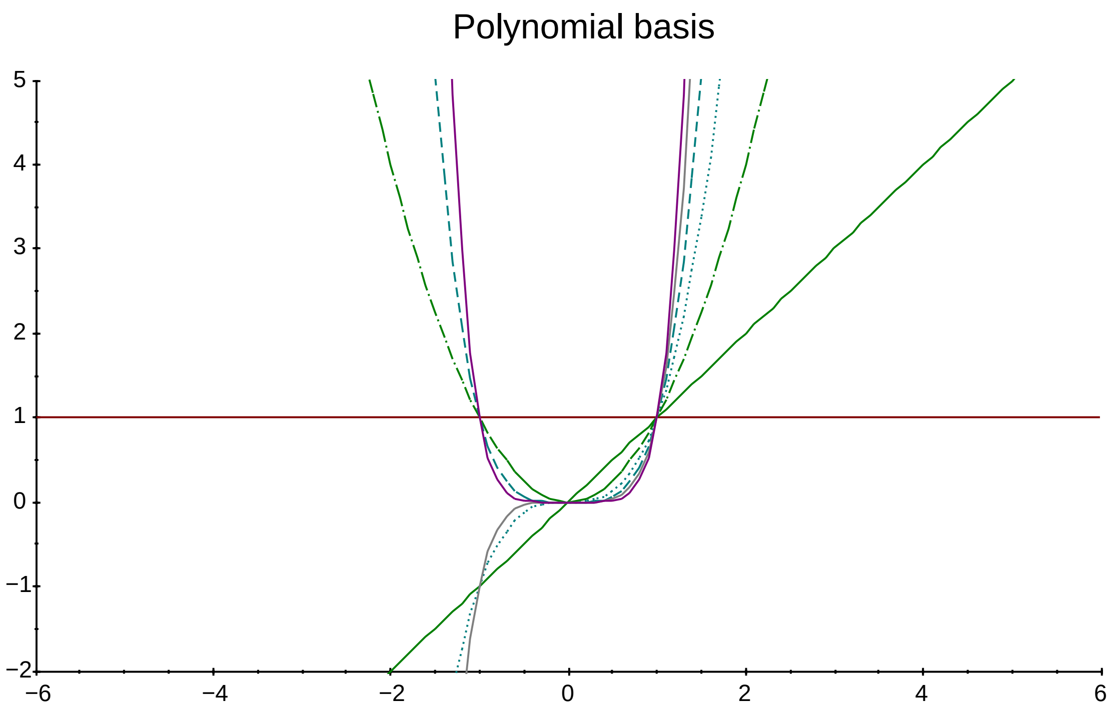


Polynomial degree 20

$$\Phi(x) \;=\; 1,\; x,\; x^2, \ldots,\; x^{20}$$



Polynomial Basis



Polynomials of the form x^k quickly become very steep.

There are much better polynomial bases : e.g. Chebyshev, Hermite, . . .

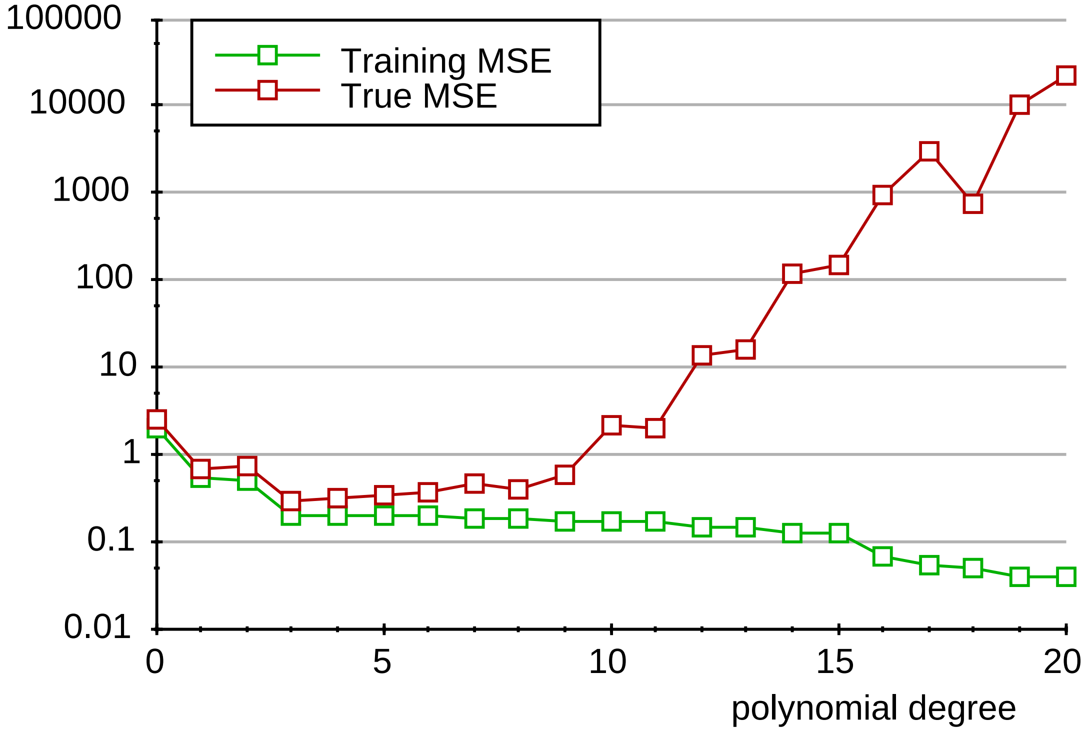
Mean squared error for polynomial models

Training set MSE:

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2$$

True MSE:

$$\frac{1}{8}\int_{-4}^{+4} \sigma^2_{\text{true}} + (\,f_{\text{true}}(x) - \hat{f}(x))^2 dx$$



Is MSE a good measure of the error ?

Why integrating on $[-4, +4]$?

About Error Measures

Domain

– should be related to the input data distribution.

Metric

– Uniform metric: L_∞

– Averaged with a L_p norm, e.g. MSE.

Derivatives

– Very close functions can have very different derivatives.

– Sobolev metrics.

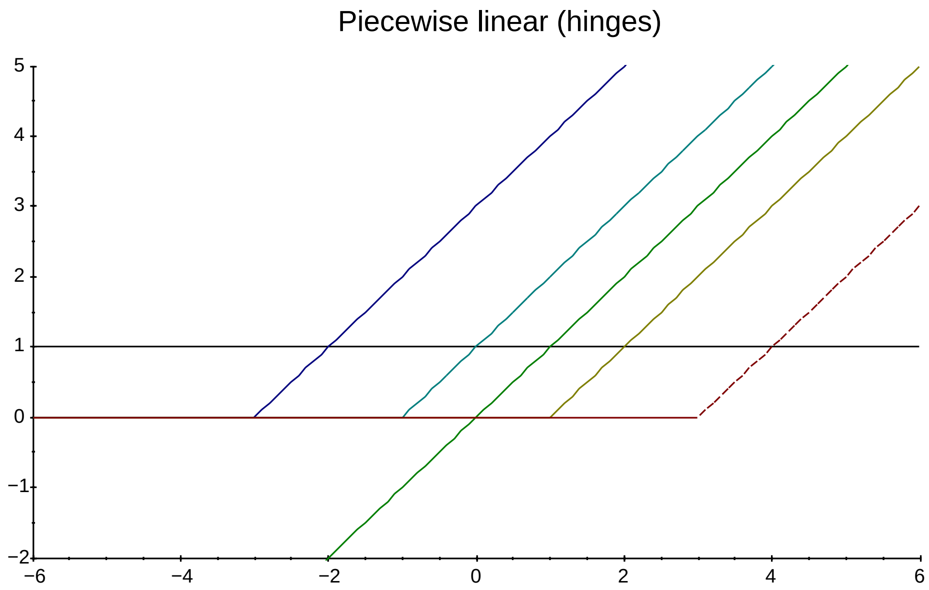
Integrals

– Conversely, very close functions always have very close integrals.

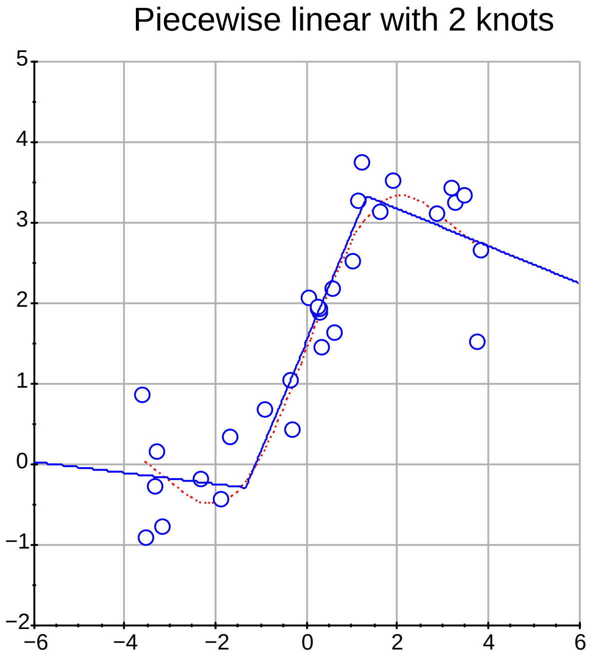
Piecewise Linear Basis

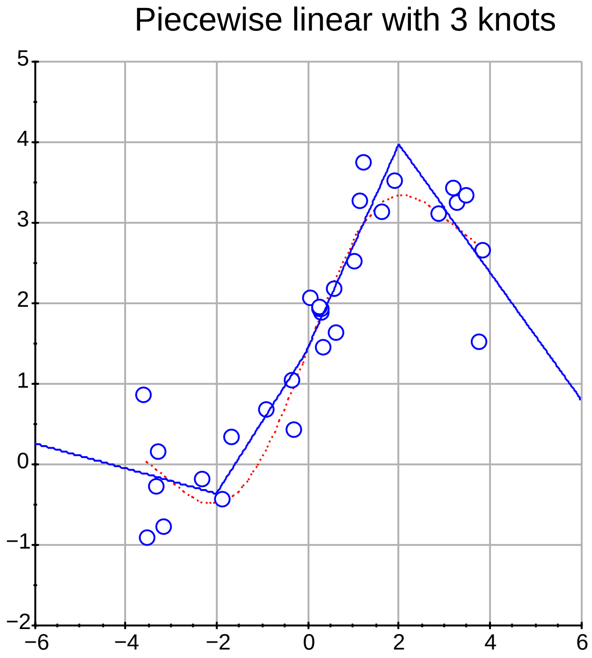
Choose knots $r_1 \dots r_k$

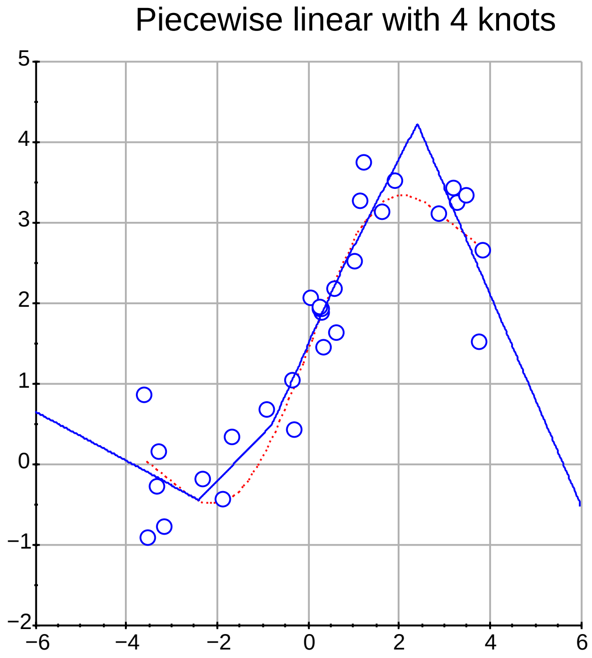
$$\begin{aligned}
\phi_0(x) &= 1 \\
\phi_1(x) &= x \\
\phi_2(x) &= \max(0, x - r_1) \\
&\dots \\
\phi_j(x) &= \max(0, x - r_{j-1})
\end{aligned}$$



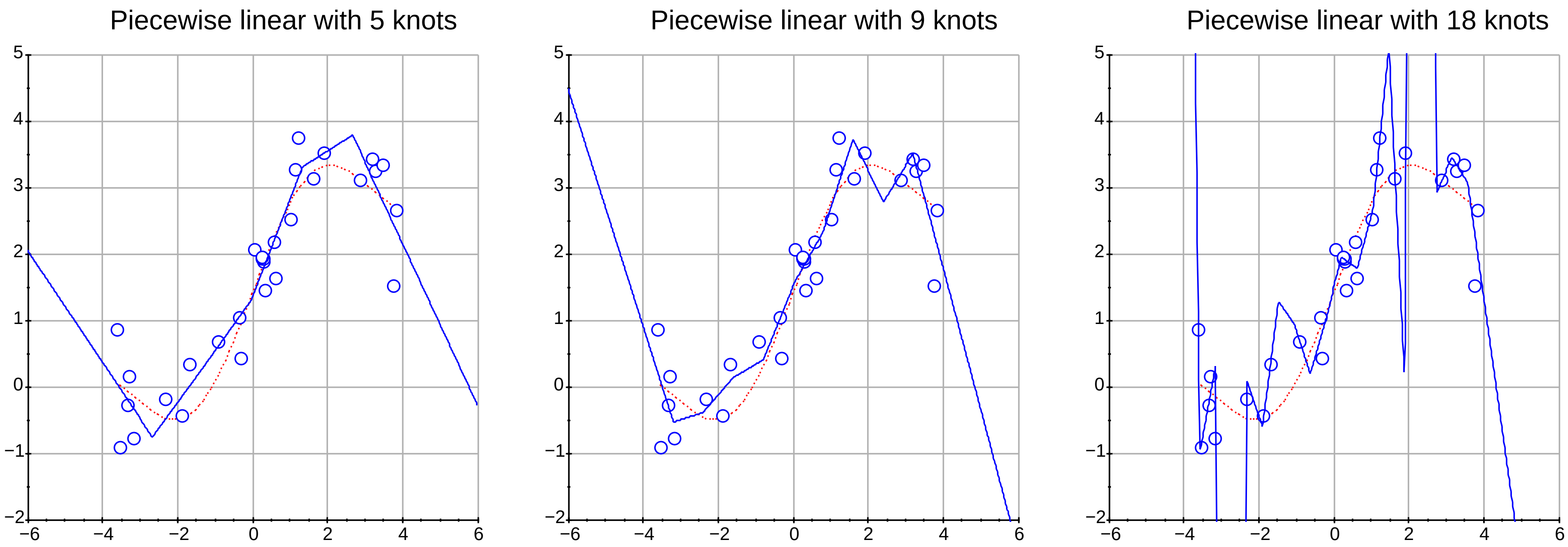
Piecewise Linear Models







Piecewise Linear Models



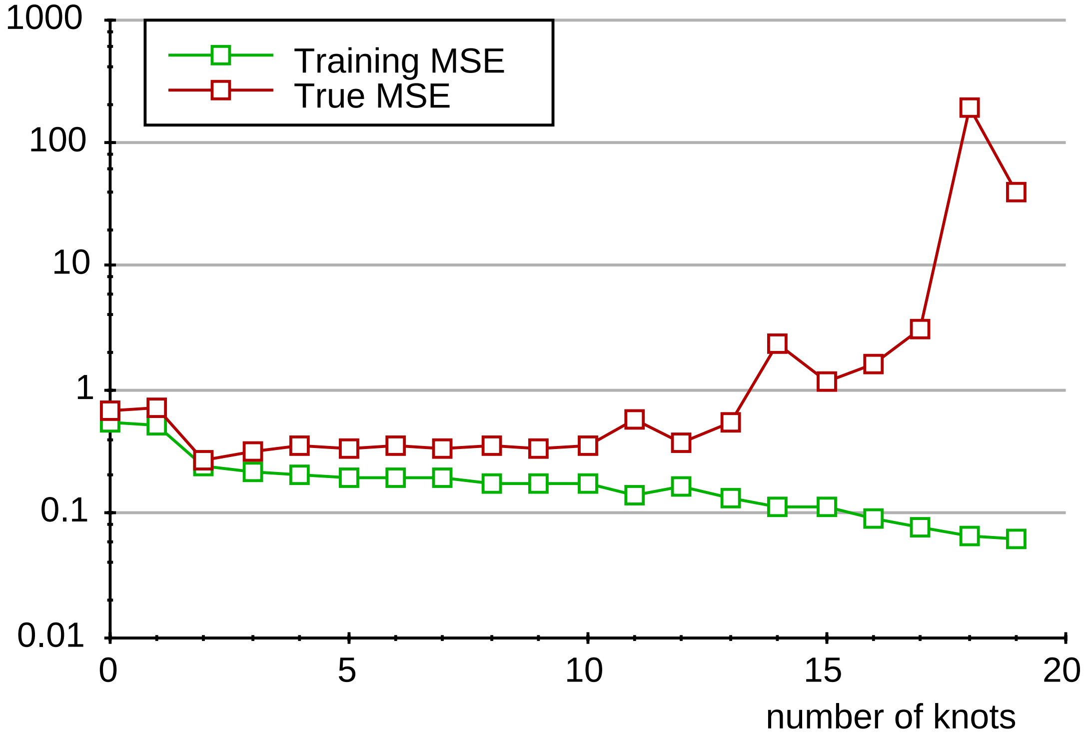
MSE for Piecewise Linear Models

Training set MSE:

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2$$

True MSE:

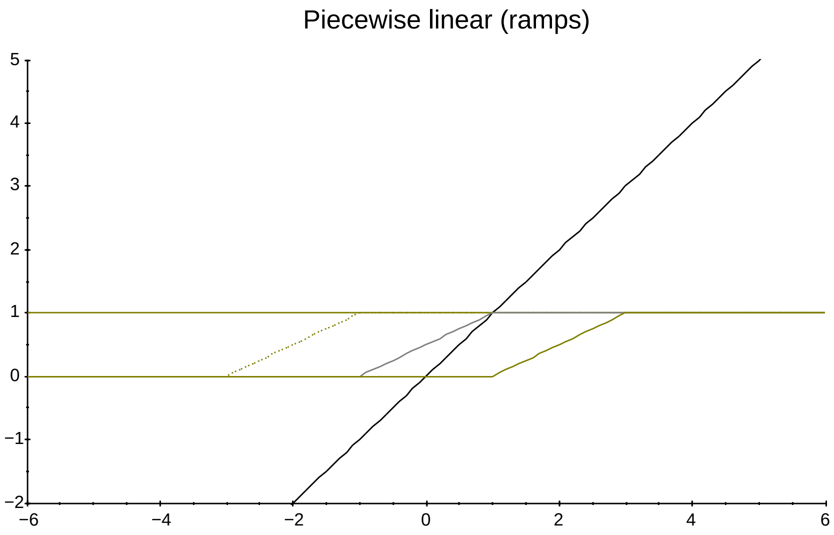
$$\frac{1}{8}\int_{-4}^{+4} \sigma_{\text{true}}^2 + (f_{\text{true}}(x) - \hat{f}(x))^2 dx$$

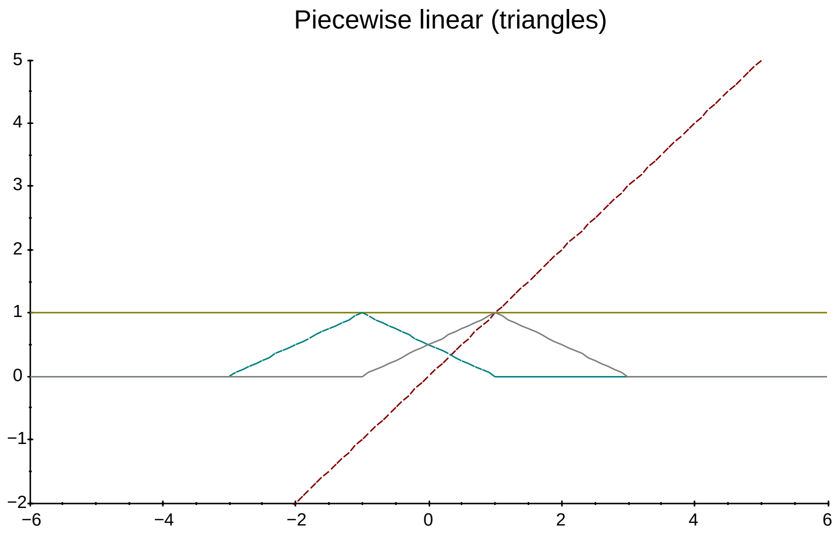


Piecewise Linear Variants

Counting the dimensions

- Linear functions on $K+1$ segments: $2K+2$ parameters.
- Continuity constraints: K constraints.
- Other constraints: 0 (hinges), 1 (ramps), 2 (triangles).





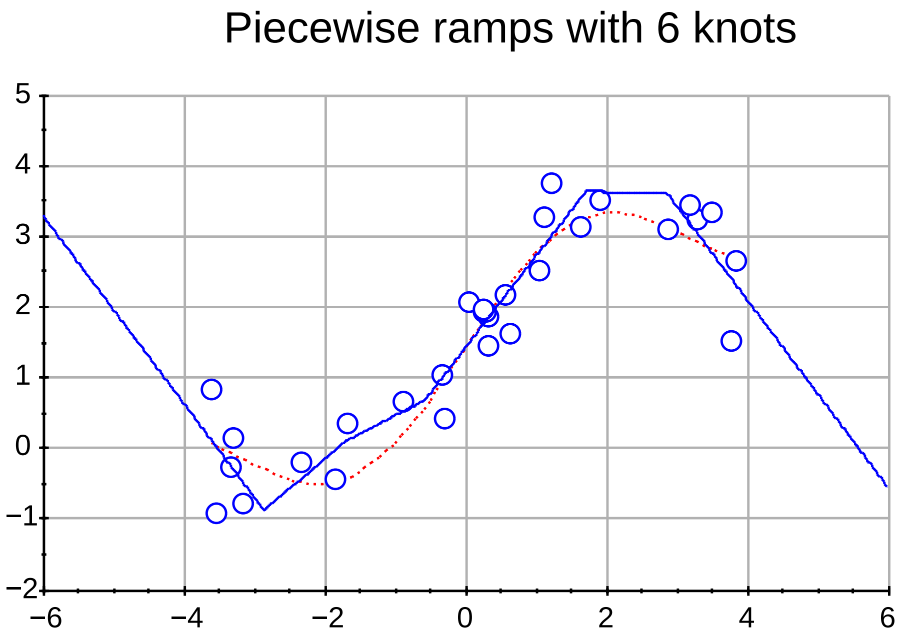
Ramps

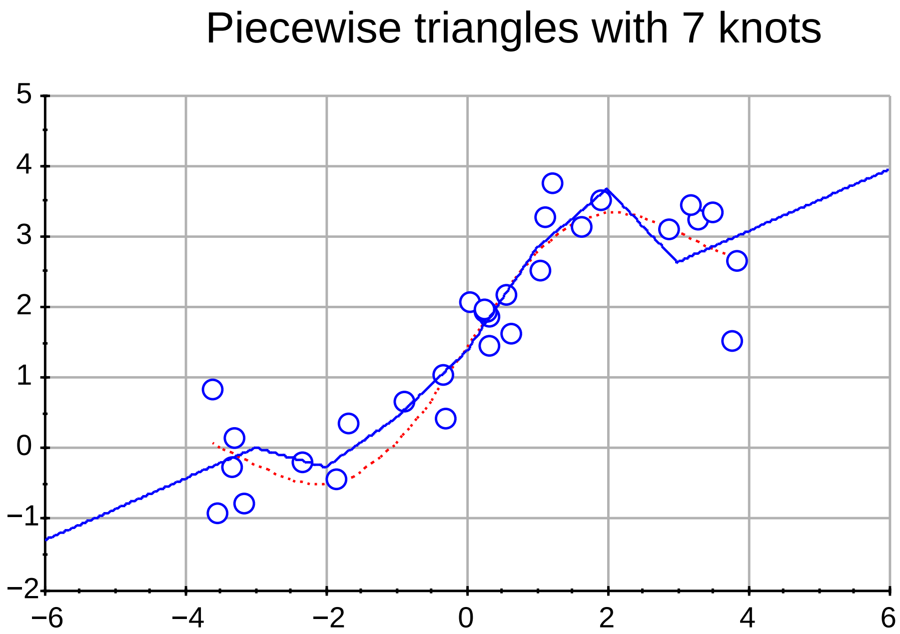
$\dim(\Phi) = \boldsymbol{K+1}$

Triangles

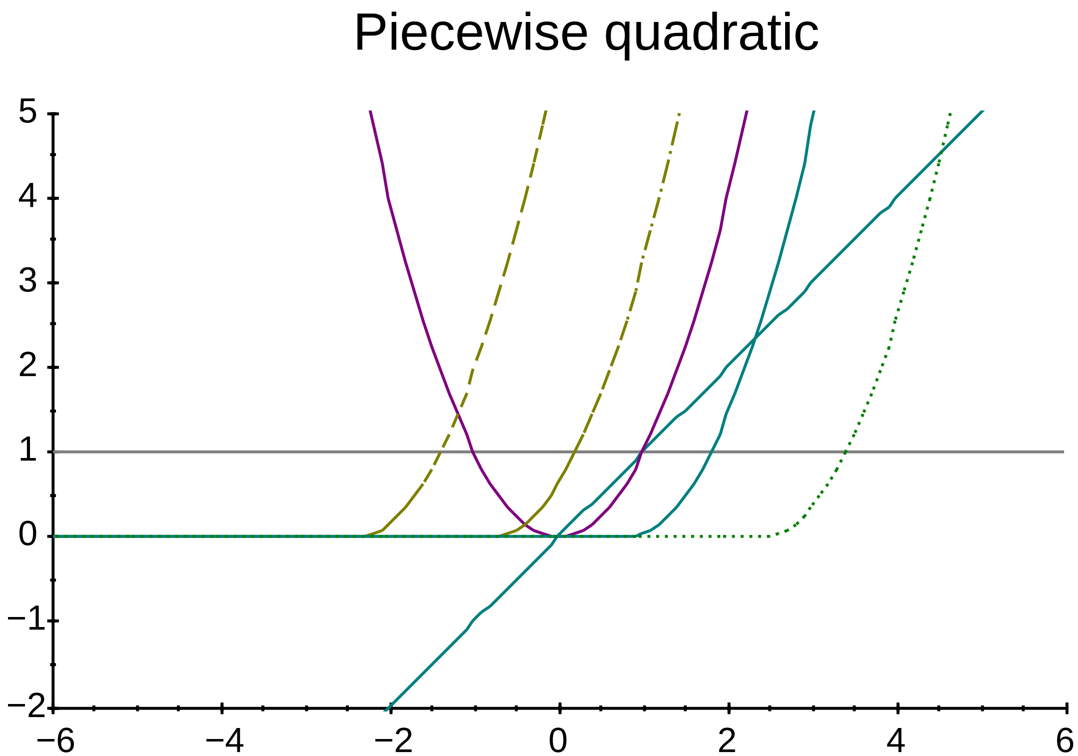
$\dim(\Phi) = \boldsymbol{K}$

Piecewise Linear Variants



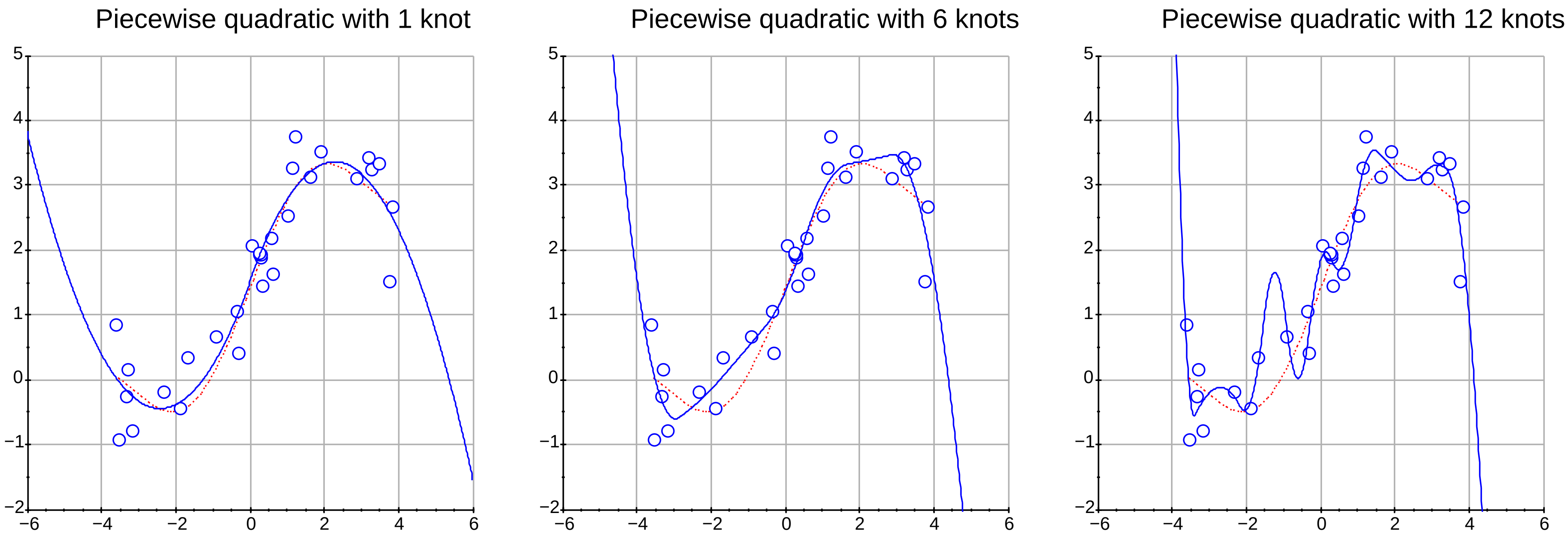


Piecewise Polynomial (Splines)



– Quadratic splines : $\Phi(x) = 1,\ x,\ x^2,\ \ldots\ \max(0, x - r_k)^2\ \ldots$
– Cubic splines : $\Phi(x) = 1,\ x,\ x^2,\ x^3,\ \ldots\ \max(0, x - r_k)^3\ \ldots$

Quadratic Splines



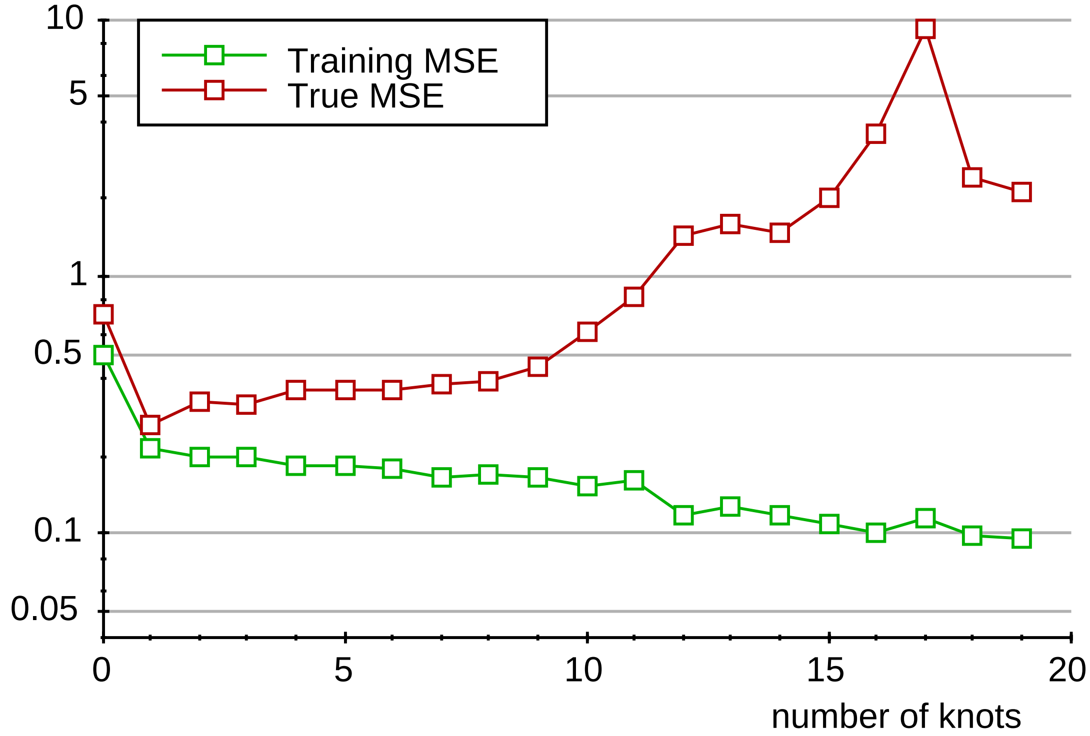
MSE for Quadratic Splines

Training set MSE:

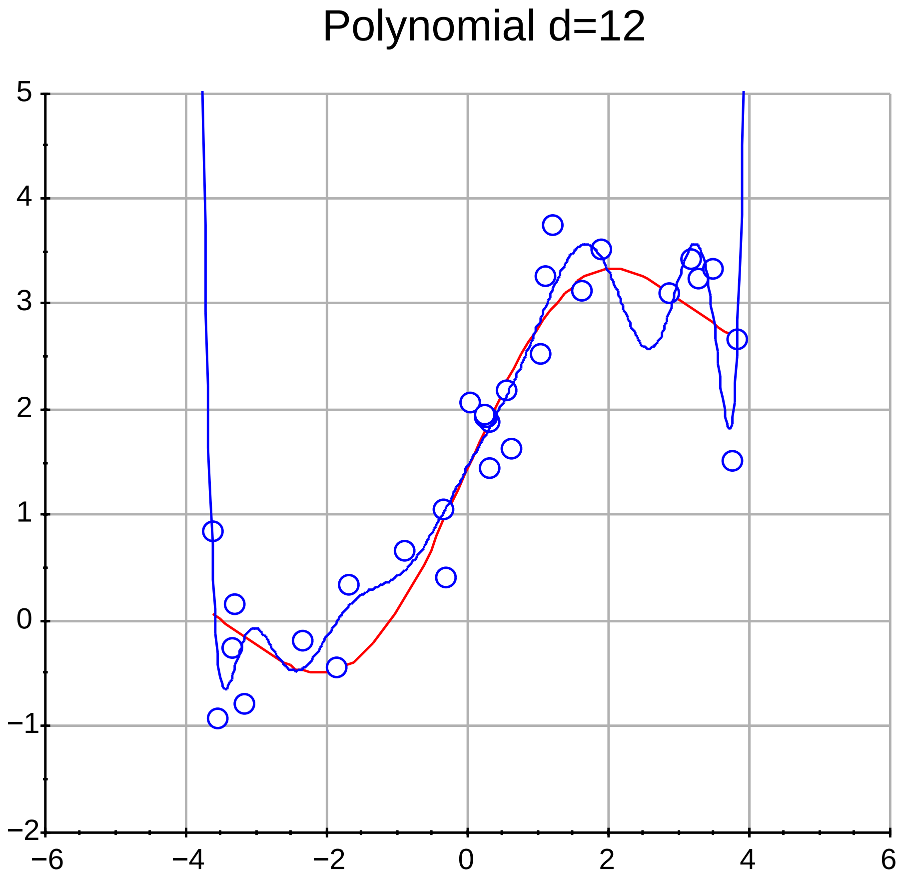
$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2$$

True MSE:

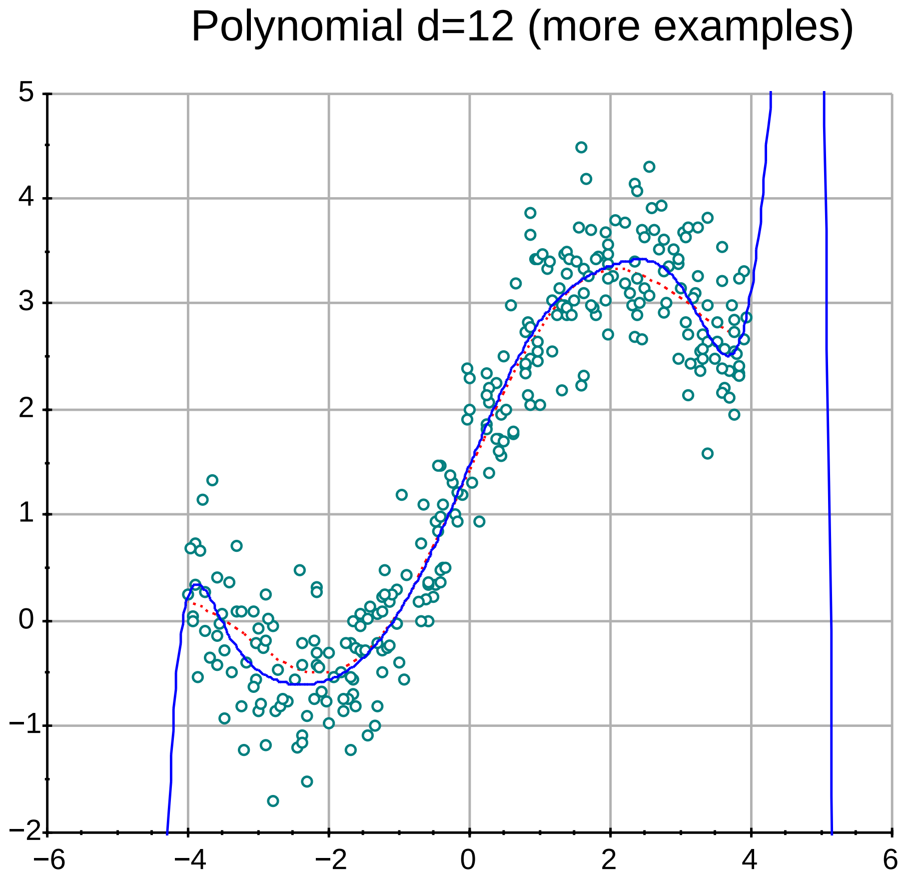
$$\frac{1}{8}\int_{-4}^{+4} \sigma_{\text{true}}^2 + (f_{\text{true}}(x) - \hat{f}(x))^2 dx$$



Changing the training data: more examples

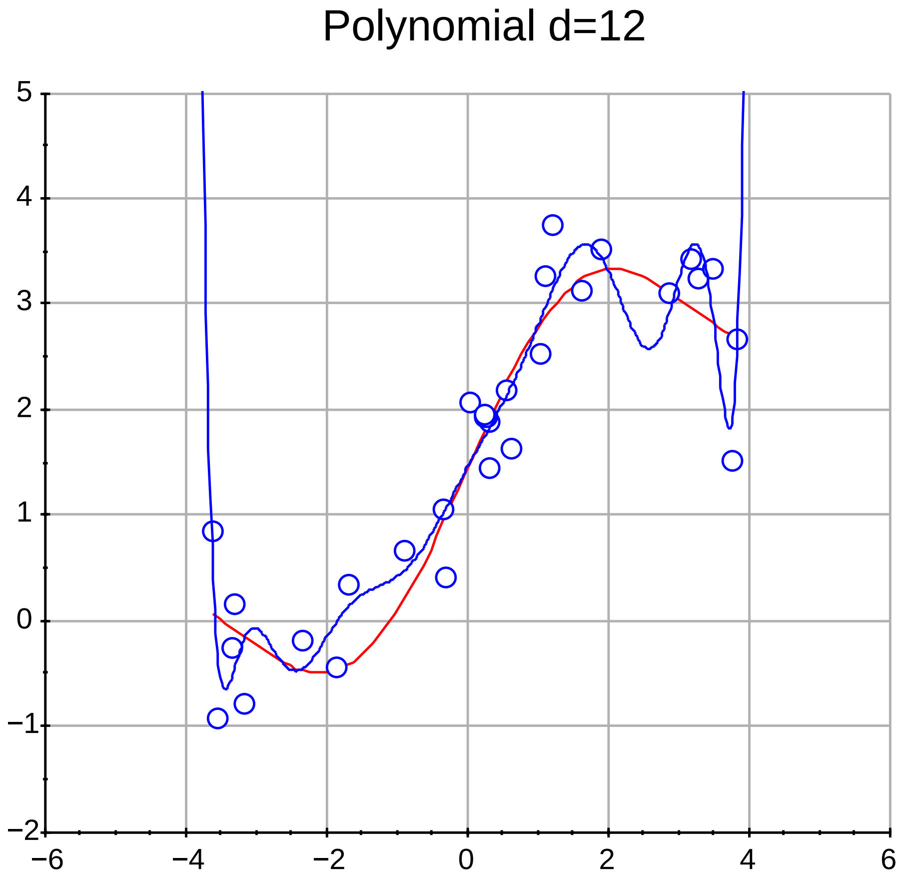


30 examples

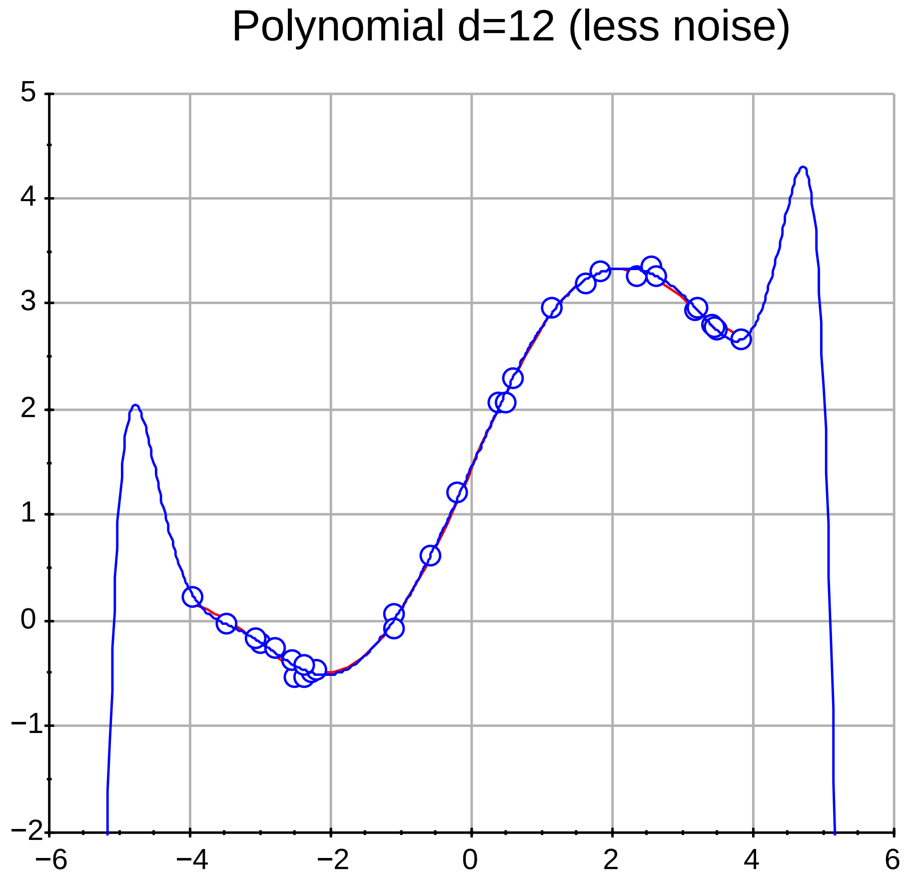


300 examples

Changing the training data: less noise



Noise sdev=0.5



Noise sdev=0.1

First Conclusions

The fancier the models, the higher the price.

– We can pay with more data.

– We can pay with better data.

In practice we do the converse.

– Changing the data is usually more costly than changing the model.

– Adapt the model "capacity" to the data.

– No shortage of methods.

The validation questions.

– We have too many options. How to choose one?

– How to estimate the quality of our work?

Estimate the quality of our work

Performance on the training data is not convincing

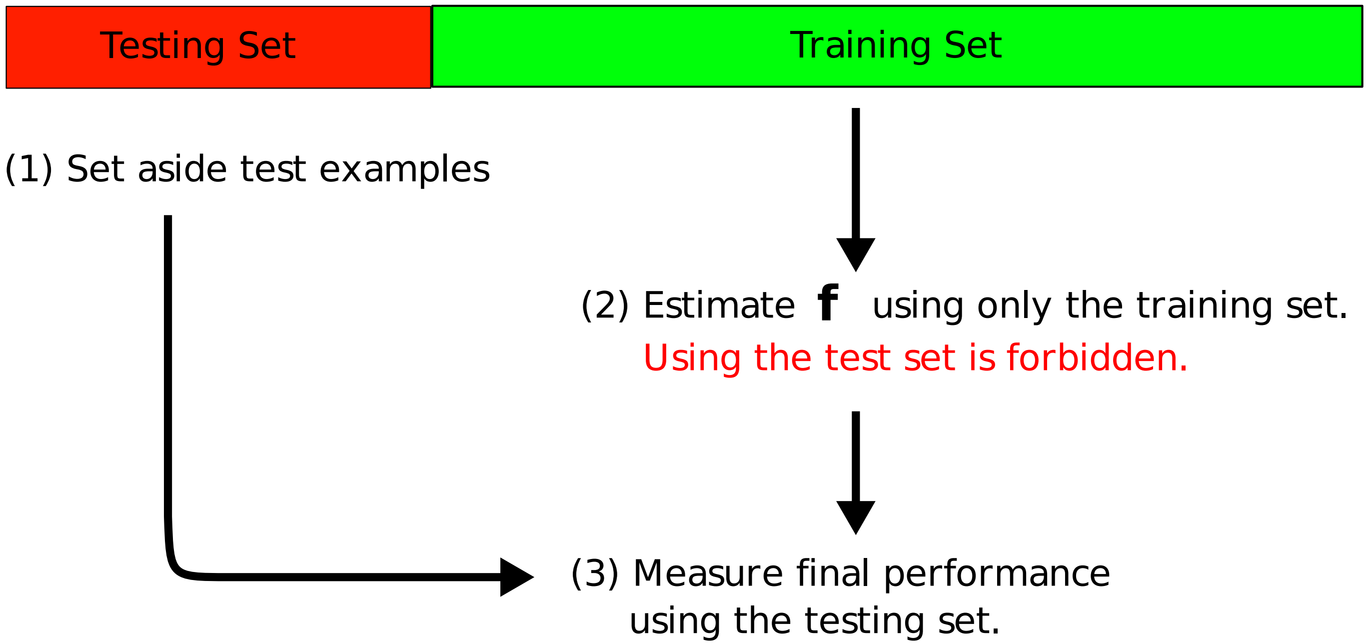
– Cannot distinguish between learning by rote and understanding.
– Understanding leads to more useful predictions than learning by rote.
– Therefore we need fresh data to evaluate our work.

- **Testing examples set aside before starting the work.**
  – Statistics work for randomly picked testing examples.
  – Real life suggests selected testing examples (e.g. time series.)
- **Testing data of a different nature.**
  – New perspective on the same phenomenon.
  – Often more instructive and convincing.

What about the “elegance” of a model ?

– Einstein: “*Make everything as simple as possible, but not simpler.*”
– How do you define “*simple*” ?

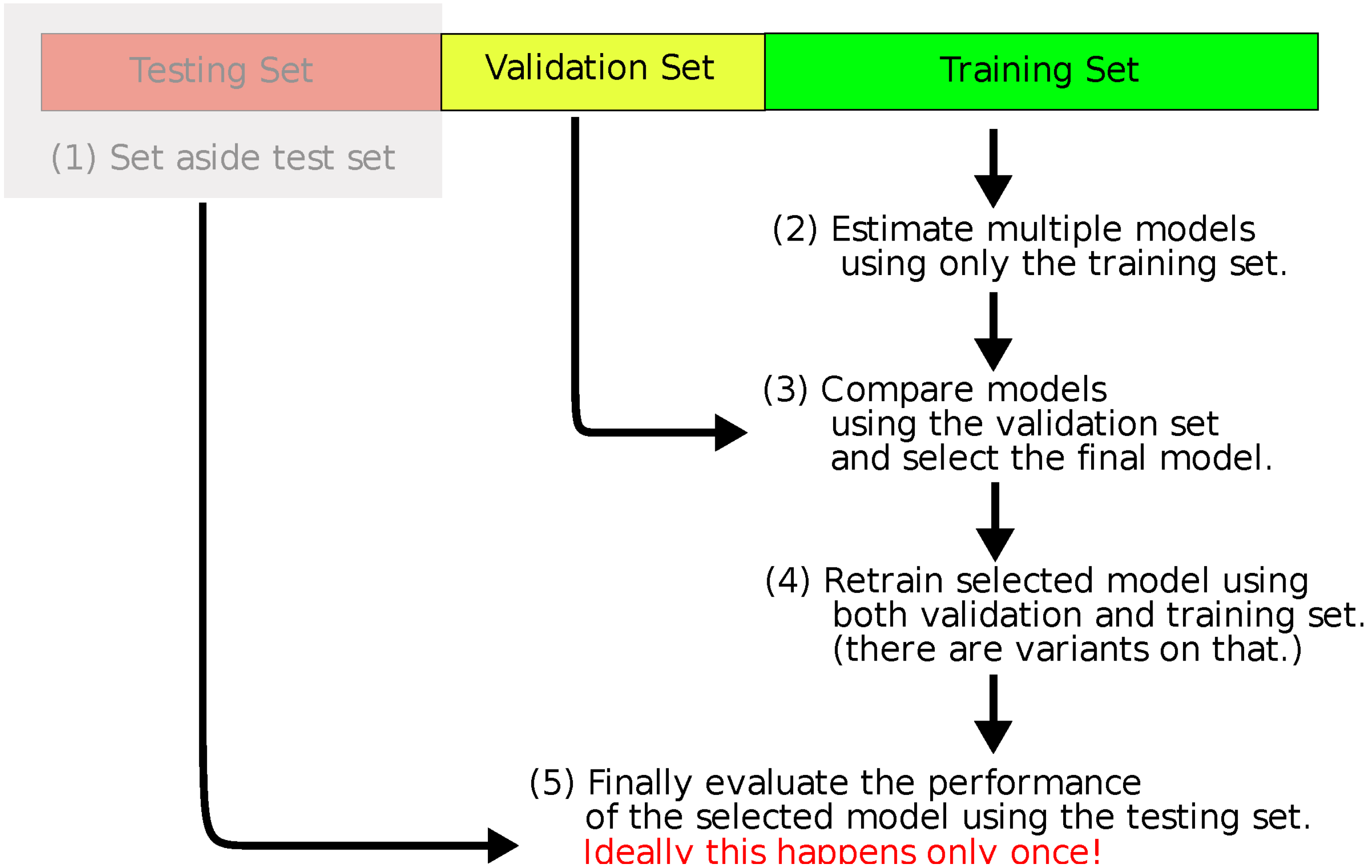
The “training set/testing set” paradigm



– One should only use the testing set once! Of course. . .
– The more we look at the testing set, the less convincing we are.
– Public benchmarks and their problems.

The "validation set"

How to select the right model without looking at the testing set ?



Potential problems

All this consumes valuable examples!

– This is a serious problem when examples are rare!

What is the optimal size of the testing set ?

– Large enough to measure the performance with sufficient accuracy.

What is the optimal size of the validation set ?

– Large enough to justify our model selection, but not larger !

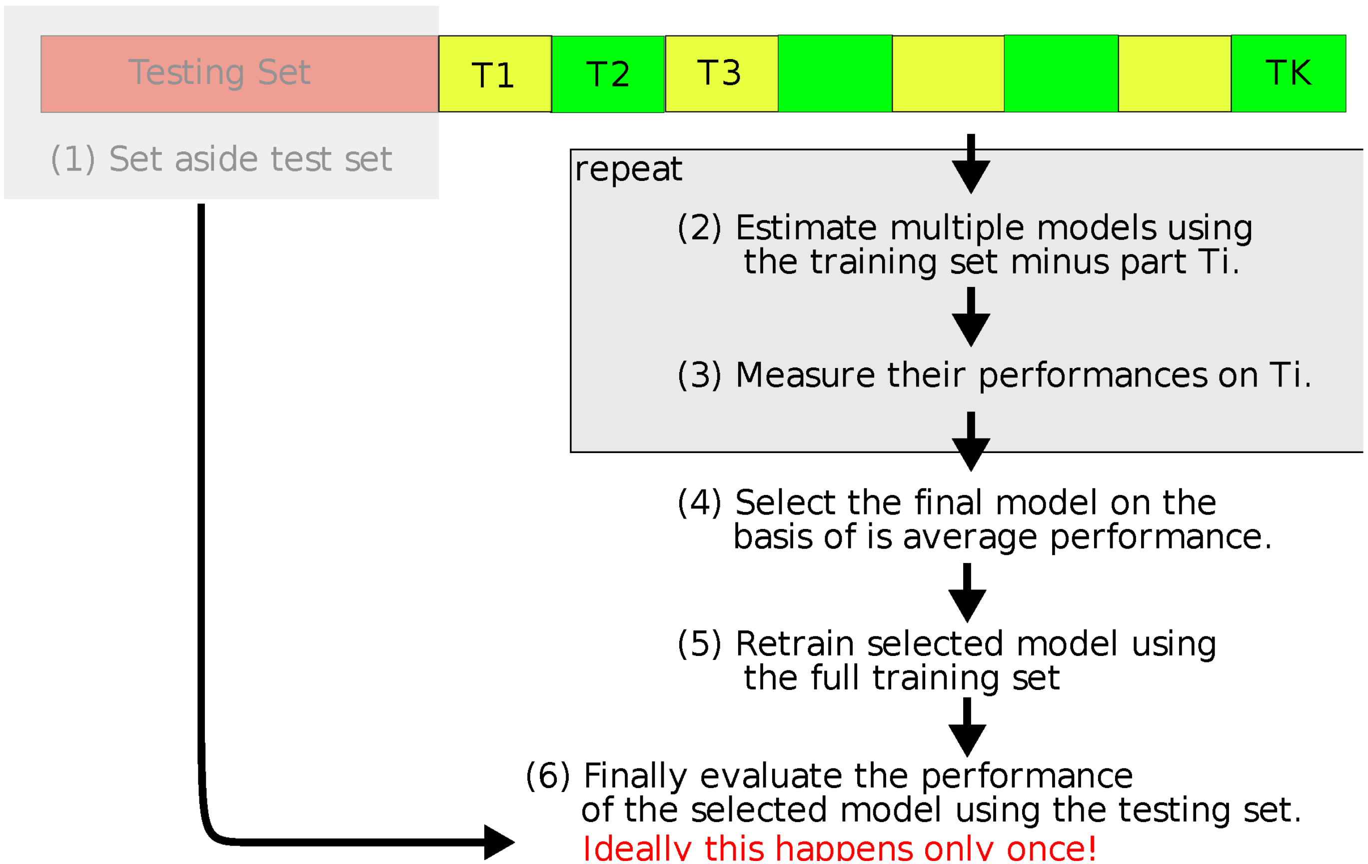
– Depends on the number of models to compare.

– Depends on the data needs of the models we compare.

– Depends on the total size of the data set.

– Trial and errors. . .

K-fold cross validation



Potential problems

All this consumes valuable computing time!

– This is a serious problem when examples are abundant.

How accurate is k-fold cross-validation?

– More than using a single parition as validation set.
– Less than using a validation set as large as the training set.
– The statistical properties of the procedure are unclear.

Suggestions

– Avoid k-fold cross validation for very large datasets.
– Observe the variations of measured performances on the folds.

Subtleties

– Evaluating the performance of a trained model.
– Evaluation the performance of a training procedure.

Beyond Curve Fitting

$$x \quad \longrightarrow \quad \Phi(x) = \begin{bmatrix} \phi_0(x) \\ \phi_1(x) \\ \dots \\ \phi_n(x) \end{bmatrix} \quad \longrightarrow \quad f(x) = [w_0, w_1, \dots, w_n] \times \begin{bmatrix} \phi_0(x) \\ \phi_1(x) \\ \dots \\ \phi_n(x) \end{bmatrix}$$

Given suitable basis functions Φ, the inputs x could be anything.

– numerical variables, e.g. 3.1415
– categorical variables, e.g. `blue`, `green`, `yellow`, `...`
– ordered variables, e.g. `small`, `medium`, `large`.
– complex data structures, such as trees, graphs, etc.
– any combination of the above.

This does not mean that constructing the features $\phi_i(x)$ will be easy.

The "adult" dataset

Predict whether income exceeds \$50K/year ($y = +1$) or not ($y = -1$).

`http://archive.ics.uci.edu/ml/datasets/Adult`

Input variables

– 6 continuous variables :

age, years of education, hours-per-week,
capital-gains, capital-losses, fnlwgt(?).

– 8 categorical variables :

workclass, education, marital status, sex,
occupation, race, relationship, native country.

Training and testing sets

– Training set: 32561 examples

– Testing set: 16281 examples

Creating $\Phi(\mathrm{x})$ for the adult dataset

Coding on 1+123 binary features $\phi_i(x)$

– First feature is always $\phi_1(x) = 1$.
– One feature for each possible value of each categorical variable.
– Five features for each continuous variable
(quantified on 5 quantiles).

copied from (Platt, 1998)

Split

– 28000 training + 4562 validation examples.
– 16281 testing examples.

Results

Experiment	Misclassification
Validation set (after training on 28K)	15.98 %
Testing set (after training on 32K)	15.47 %

A quadratic basis for the adult dataset

Coding on 1+123+7503 features

– Additional features for quadratic models.

$$\forall i \in 1 \dots 123 \quad \forall j \in 1 \dots i-1 \quad \phi_{ij}(x) = \phi_i(x)\phi_j(x)$$

Remarks

– Feature count grows quickly.

– This is slow (X is sparse, but $X^\top X$ is not.)

Results

Experiment	Misclassification
Validation set (after training on 28K)	16.40 %
Testing set (after training on 32K)	— %

Weighting the quadratic terms

Idea

Remember the regularization + cholevsky trick?

$$\min \; C(w) + \varepsilon w^2 \iff (\, X^\top X + \varepsilon I \,)\, w = (\, X^\top Y \,)$$

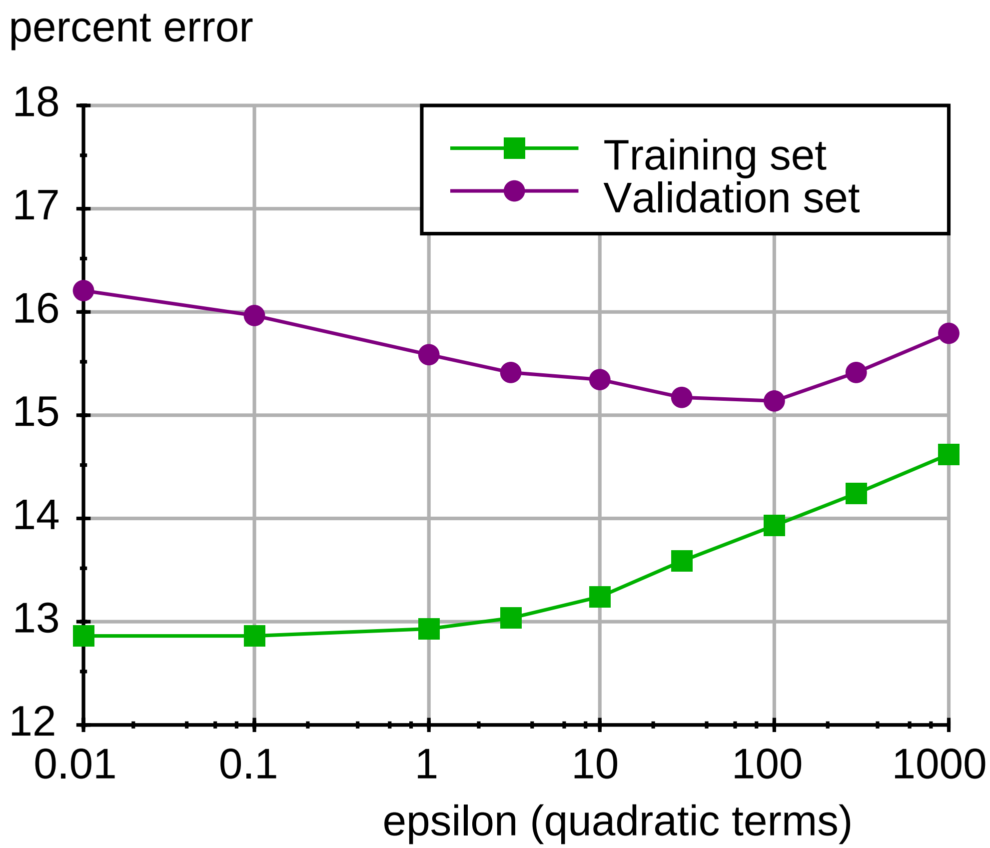
Let's penalize more the coefficients of the quadratic terms.

$$\min \; C(w) + w^\top \Lambda w \iff (\, X^\top X + \Lambda \,)\, w = (\, X^\top Y \,)$$

Details

– $\varepsilon = 10^{-5}$ for constant and linear terms.
– $\varepsilon \in [10^{-5}, 10^{5}]$ for quadratic terms.

Weighting the quadratic terms



We get the linear result when $\varepsilon \to \infty$.

We get the quadratic result when $\varepsilon \to 0$.

After retraining with $\varepsilon = 100$ on all 32K examples:
Testing set error: 14.93 %.

Coming next

Homework 1

– Due on Tue Feb 23rd.

– Something about splines.

Next lectures

– Tuesday Feb 9th:	R tutorial (Sean Gerrish)
– Thursday Feb 11th:	Review of probabilities