T-61.3050 Machine Learning: Basic Principles

Supervised Learning

Kai Puolamäki

Laboratory of Computer and Information Science (CIS)
Department of Computer Science and Engineering
Helsinki University of Technology (TKK)

Autumn 2007

Outline

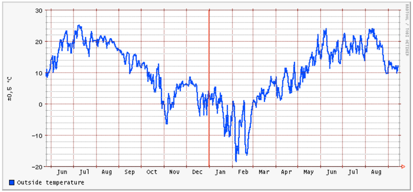
1. Learning a Class from Examples
   - Introduction
   - Aldo and Family Car
   - PAC Learning and VC Dimension
2. Noise and Regression
   - Noise
   - Regression
   - Validation
3. Conclusion
   - About Supervised Learning
   - Better Basis Functions

Learning a Class from Examples

- What follows is some theory of classification into two classes.
- We assume there is no noise (results can be generalized to noise, though).
- What you should learn:
  - Learning can be seen as pruning out possible hypothesis.
  - Learning is generalization (we want to predict classes of new examples).
  - Learning is impossible if the hypothesis space is too large (in other words: we need some prior information, we need to select a model family)
  - The complexity of the hypothesis space (model family) can be characterized using the VC dimension.
  - More complex model, bigger the training data needed.

Independent and Identically Distributed (iid) Data

- We assume that we have a training data $\mathcal{X}$ that contains N data points drawn independently from the identical distribution.
- In other words: ordering of the data points does not matter.
- Usually a good approximation.
- Notable exception: time series.
- Example: today's temperature is not independent of the yesterday's temperature, in fact, there is a strong correlation.



Outside temperature in Otaniemi from `http://outside.hut.fi/`.

Outline

1. Learning a Class from Examples
   - Introduction
   - Aldo and Family Car
   - PAC Learning and VC Dimension
2. Noise and Regression
   - Noise
   - Regression
   - Validation
3. Conclusion
   - About Supervised Learning
   - Better Basis Functions

Does Aldo Enjoy Sport?

- Question: Does Aldo enjoy sport, given weather conditions?
- Assumption: we have sufficient information (6 weather attributes) that fully determine Aldo's enjoyment of sports (no “noise”, Aldo is deterministic).

			$\mathbf{x}^t$				$r(\mathbf{x}^t)$
t	*Sky*	*AirTemp*	*Humidity*	*Wind*	*Water*	*Forecast*	*EnjoySport*
1	Sunny	Warm	Normal	Strong	Warm	Same	1
2	Sunny	Warm	High	Strong	Warm	Same	1
3	Rainy	Cold	High	Strong	Warm	Change	0
4	Sunny	Warm	High	Strong	Cool	Change	1

Table: Aldo's observed sport experiences in different weather conditions.

Does Aldo Enjoy Sport?

Hypothesis Class

- Hypothesis h is a function from weather attributes $\mathbf{x}$ to $\{0, 1\}$.
- Hypothesis class $\mathcal{H}$ is the chosen set of hypothesis.
- The goal of the learner is to find a hypothesis $h \in \mathcal{H}$ such that $h(\mathbf{x}) = r(\mathbf{x})$ for every possible $\mathbf{x}$.
- One possible hypothesis class in Aldo's case is a vector of six weather attributes. For each attribute, the hypothesis will be either:
  - ?: any value is acceptable for this attribute.
  - single value (e.g., "Warm"): required value for this attribute.
  - $\emptyset$: no value is acceptable.
- If an instance $\mathbf{x}$ satisfies the constraints then h classifies this as a positive example, $h(\mathbf{x}) = 1$.
- Example: Aldo enjoys the sport only on cold days with high humidity (independent of other attributes), this would be represented with $(?, Cold, High, ?, ?, ?)$.

Does Aldo Enjoy Sport?

General and Specific Hypothesis

Definition

Let h and g be hypothesis on X. h is more general than or equal to g (written $h \succeq g$) if and only if

$$\forall \mathbf{x} \in X : g(\mathbf{x}) = 1 \Rightarrow h(\mathbf{x}) = 1.$$

Examples:

- The most general hypothesis is represented by $(?, ?, ?, ?, ?, ?)$ (every day is a positive example).
- The most specific hypothesis is represented by $(\emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset)$ (no day is a positive example).
- $h = (Sunny, ?, ?, ?, ?, ?)$ is more general than $g = (Sunny, ?, ?, Strong, ?, ?)$, or $h \succ g$.

Does Aldo Enjoy Sport?

Consistent hypothesis

Definition (Consistent Hypothesis)

A hypothesis h is consistent with a set of training examples $\mathcal{X}$ if and only if $h(\mathbf{x}) = r(\mathbf{x})$ for each example $(\mathbf{x}, r) \in \mathcal{X}$.

Definition (Version Space)

The version space is the set of all hypothesis that are consistent with the training examples.

Does Aldo Enjoy Sport?

Maximally general and specific hypothesis

- Question 1: What are the most general hypothesis that are consistent with the training data (4 days of observation of Aldo)? (general boundary G)
- Question 2: What are the most specific hypothesis that are consistent with the training data? (specific boundary S)

Theorem (Version Space Representation Theorem)

Let G and S the most general and most specific hypothesis that are consistent with the training data. Then all hypothesis that are consistent with the training data (version space) are given by

$$\{h \in \mathcal{H} \mid (\exists s \in S)(\exists g \in G) : g \succeq h \succeq s\} .$$

Does Aldo Enjoy Sport?

All consistent hypothesis

$$\begin{tikzcd}
 & G = \{(Sunny, ?, ?, ?, ?, ?), \arrow[dl] \arrow[dr] & & (?, Warm, ?, ?, ?, ?)\} \arrow[dl] \arrow[dr] & \\
(Sunny, ?, ?, Strong, ?, ?) \arrow[drr] & & (Sunny, Warm, ?, ?, ?, ?) \arrow[d] & & (?, Warm, ?, Strong, ?, ?) \arrow[dll] \\
 & & S = \{(Sunny, Warm, ?, Strong, ?, ?)\} & &
\end{tikzcd}$$

	$\mathbf{x}^t$						$r(\mathbf{x}^t)$
t	*Sky*	*AirTemp*	*Humidity*	*Wind*	*Water*	*Forecast*	*EnjoySport*
1	Sunny	Warm	Normal	Strong	Warm	Same	1
2	Sunny	Warm	High	Strong	Warm	Same	1
3	Rainy	Cold	High	Strong	Warm	Change	0
4	Sunny	Warm	High	Strong	Cool	Change	1

See Mitchell (1997) and CANDIDATE-ELIMINATION algorithm for details.

Does Aldo Enjoy Sport?

- One of the consistent hypothesis could be the "truth". For others we get some error:

Definition (Error of Hypothesis)

$$E\left(h \mid \mathcal{X}\right) = \frac{1}{N}\sum_{t=1}^{N} \mathbf{1}\left(h(\mathbf{x}^t) \neq r^t\right)$$

- Given enough training samples, we might be able to end up with only one consistent hypothesis.
- Given enough training samples, we might end up with no consistent hypothesis if:
  - If none of the hypothesis in the hypothesis class is correct. (For example, if Aldo would enjoy sport only if (sky is sunny and wind is strong) or (sky is rainy and wind is light).)
  - If there is noise (e.g., some positive examples are incorrectly observed as negative examples).

Does Aldo Enjoy Sport?

Inductive bias

- If none of the hypothesis in the hypothesis class is correct we might end up with no consistent hypothesis.
- “Solution”: include all possible hypothesis into the hypothesis class! In the Aldo’s case, there are $2^{2^6} = 1.8 \times 10^{19}$ possible hypothesis (number of boolean functions with 6 inputs).
- This does not work (even if we could compute): we could not say anything of the unseen cases.
- Inductive bias: we must restrict the allowed hypothesis to be able to generalize (predict classes of new instances).
- The selection of hypothesis space is called model selection.
- Underfitting: the hypothesis space is too simple.
- Overfitting: the hypothesis space is too complex.

A Family Car

- Question 1: Is car $\mathbf{x}$ a family car, given car properties?
- Question 2: What do people expect from a family car?
- Car properties: $\mathbf{x} = (\text{price}, \text{engine power})$.
- Hypothesis: $h(\mathbf{x}) = 1$ if car is a family car.

A Family Car

Training set

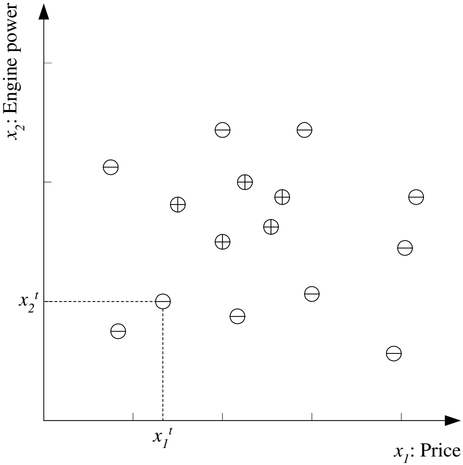


Figure 2.1 of Alpaydin (2004).

$$\mathcal{X} = \left\{\mathbf{x}^t, r^t\right\}_{t=1}^N$$

$$r = \begin{cases} 1 \text{ if } \mathbf{x} \text{ is positive} \\ 0 \text{ if } \mathbf{x} \text{ is negative} \end{cases}$$

$$\mathbf{x} = \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{pmatrix}$$

A Family Car

True class

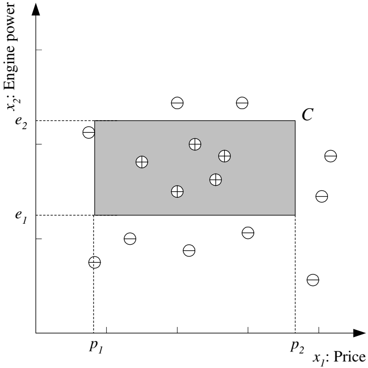


Figure 2.2 of Alpaydin (2004).

$$r(\mathbf{x}) = \begin{cases} 1 & p_1 \le \text{price} \le p_2 \wedge e_1 \le \text{engine power} \le e_2 \\ 0 & \text{otherwise} \end{cases}$$

A Family Car

Hypothesis class H

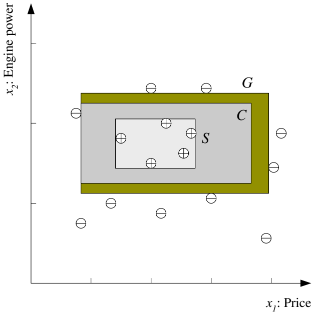


Figure 2.4 of Alpaydin (2004).

Error of h in $\mathcal{X}$:

$$E(h \mid \mathcal{X}) = \frac{1}{N} \sum_{t=1}^{N} \mathbf{1}\left(h(\mathbf{x}^t) \neq r^t\right).$$

$h \in \mathcal{H}$ between S and G is consistent and make up the version space (error in $\mathcal{X}$ is zero). Notice that if S and G are close the error on new data will be small!

- The hypothesis class H is the set of all rectangles.
- The cars between the most general (G) and most specific (S) hypothesis may be classified incorrectly. C is the correct hypothesis.

What did we learn from Aldo and Family Cars?

- We must choose some hypothesis to be able to predict anything (unless we observe all possible data values). (model selection)
- This causes *inductive bias* (the choice of hypothesis space affects your results).
- All consistent hypothesis can be found between the most general and most specific hypothesis.
- There may be no consistent hypothesis due to too simple hypothesis space (underfitting) or noise. These must be taken into account in practical applications.

Outline

1. Learning a Class from Examples
   - Introduction
   - Aldo and Family Car
   - PAC Learning and VC Dimension
2. Noise and Regression
   - Noise
   - Regression
   - Validation
3. Conclusion
   - About Supervised Learning
   - Better Basis Functions

Probably Approximately Correct (PAC) Learning

- How many training examples N should we have, such that with probability of at least $1 - \delta$, any consistent hypothesis h has error at most ϵ?

Probably Approximately Correct (PAC) Learning

Theorem

The probability that version space has no hypothesis with error greater than ϵ is at most $|\mathcal{H}|e^{-\epsilon N}$.(Assume finite hypothesis class $\mathcal{H}$.)

Proof.

The probability that a hypothesis that has an error greater than ϵ is consistent with one randomly drawn example is at most $1 - \epsilon$. Therefore, the probability that this hypothesis is consistent with N independently drawn examples is at most $(1-\epsilon)^N$. There are at most $|\mathcal{H}|$ hypothesis that have an error greater than ϵ. The probability that there is at least one hypothesis in the version space with an error greater than ϵ is at most $|\mathcal{H}|(1-\epsilon)^N \leq |\mathcal{H}|e^{-\epsilon N}$. □

It follows that $|\mathcal{H}|e^{-\epsilon N} \leq \delta$, or $N \geq \frac{1}{\epsilon}\left(\ln|\mathcal{H}| + \ln(1/\delta)\right)$.

Probably Approximately Correct (PAC) Learning

Theorem (Probably Approximately Correct (PAC) Learning)

We should have N training examples to have an probability of at least $1 - \delta$ that any consistent hypothesis h has error at most ϵ, where

$$N \geq \frac{1}{\epsilon} \left(\ln |\mathcal{H}| + \ln \frac{1}{\delta} \right)$$

If we accept that the best hypothesis might have a non-zero training error (often case in practice) the limit becomes

$$N \geq \frac{1}{\epsilon^2} \left(\ln |\mathcal{H}| + \ln \frac{1}{\delta} \right) ,$$

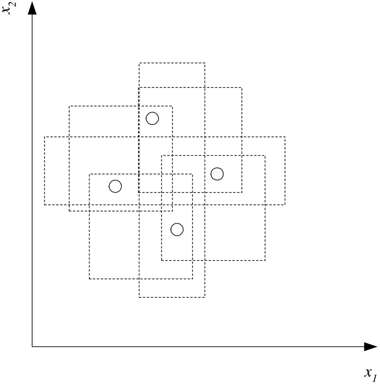
where the obtained error will be with probability $1 - \delta$ no more than $E(h_{best} \mid \mathcal{X}) + \epsilon$, where $E(h_{best} \mid \mathcal{X})$ is the error of the best hypothesis.

Vapnik-Chervonekis (VC) Dimension

- N points can be labelled $r^t = 0/1$ in 2^N ways.
- $\mathcal{H}$ shatters N points if there exists $h \in \mathcal{H}$ consistent for all 2^N labellings.

Definition (VC Dimension)

VC Dimension is the largest number N of points that can be shattered by $\mathcal{H}$.



Rectangles can shatter four points, $VC = 4$. Figure 2.5 of Alpaydin (2004).

PAC Bound using VC Dimension

Theorem

We should have N training examples to have an probability of at least $1 - \delta$ that any consistent h has error at most ϵ, where

$$N \geq \frac{1}{\epsilon} \left(4 \log_2 \frac{2}{\delta} + 8 VC(\mathcal{H}) \log_2 \frac{13}{\epsilon} \right) .$$

- We can use the VC dimension instead of $\ln |H|$ as a measure of model complexity.
- Lesson: larger VC dimension, more complex model, more training samples are needed.
- (See Mitchell (1997), chapter 7, for details.)

What Did We Learn of PAC Learning and VC Dimension?

- Hypothesis class complexity (or model complexity) can be evaluated using the VC dimension.
- More complex model, more data you need to learn (learning is ability to describe the true hypothesis with a given confidence).
- PAC bounds are extremely conservative, in practice (when we also have noise) we usually need significantly smaller data sets.

Outline

1. Learning a Class from Examples
   - Introduction
   - Aldo and Family Car
   - PAC Learning and VC Dimension
2. Noise and Regression
   - Noise
   - Regression
   - Validation
3. Conclusion
   - About Supervised Learning
   - Better Basis Functions

Noise and Model Complexity

- Noise is unwanted anomaly of data.
- Because of the noise, we may never reach zero error.
- Noise may be caused by:
  - Errors in measurements of input attributes or class labels.
  - Unknown or ignored (hidden or latent) attributes.
- Noise is best treated probabilistically (next lectures).
- Why to use simpler model:
  - simpler to use
  - easier to train
  - easier to explain
  - generalizes better

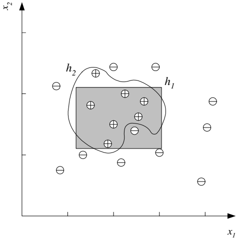


Figure 2.7 of Alpaydin (2004).

Outline

1. Learning a Class from Examples
   - Introduction
   - Aldo and Family Car
   - PAC Learning and VC Dimension
2. Noise and Regression
   - Noise
   - Regression
   - Validation
3. Conclusion
   - About Supervised Learning
   - Better Basis Functions

Regression

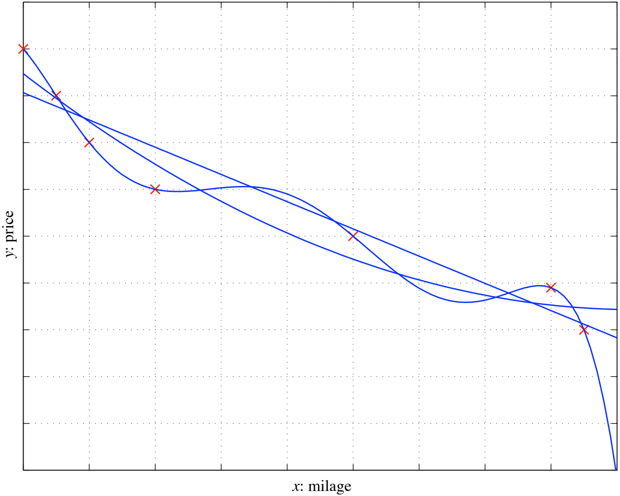


Figure 2.9 of Alpaydin (2004).

Regression

- Classification is the prediction of a 0–1 class, given attributes.
- Regression is the prediction of a real number, given attributes. (Usually with noise.)
- The training set is given by $\mathcal{X} = \{\mathbf{x}^t, r^t\}_{t=1}^N$, where $r^t \in \mathbb{R}$.
- We imagine that the r^t are given by some function $r^t = f(\mathbf{x}^t, \mathbf{z}^t)$, where $\mathbf{z}^t$ are some unknown hidden variables.
- The role of hypothesis is taken by the model $g(\mathbf{x})$. We would like to find a model such that $g(\mathbf{x}^t) \approx r^t$ for all items in the training set.
- Usually, we want to minimize a quadratic error function,

$$E(g \mid \mathcal{X}) = \frac{1}{N} \sum_{t=1}^{N} \left(r^t - g(\mathbf{x}^t)\right)^2 .$$

Linear Regression

- The simplest case is linear regressor: $g(\mathbf{x}) = w_0 + w_1 \mathbf{x}$.
- Optimization task: find w_0 and w_1 such that the error $E(g \mid \mathcal{X}) = \frac{1}{N} \sum_{t=1}^{N} \left(r^t - (w_0 + w_1 \mathbf{x}^t)\right)^2$ is minimized.

Analytic solution:

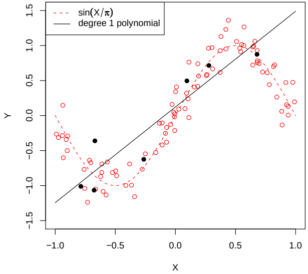
$$\begin{aligned} w_1 &= \frac{\sum_t x^t r^t - \overline{x}\overline{r}N}{\sum_t (x^t)^2 - N\overline{x}^2}, \\ w_0 &= \overline{r} - w_1 \overline{x}, \end{aligned}$$

where $\overline{x} = \sum_t \mathbf{x}^t / N$ and $\overline{r} = \sum_t r^t / N$.

Linear Regression

Toy data

- Toy data: we have generated 100 data points using $\sin(X/\pi)$ in interval $[-1, 1]$, added with Gaussian random noise.
- We randomly selected 7 data points to act as the training data (shown in black).
- Solution: $g(\mathbf{x}) = 0.12 + 1.37\mathbf{x}$.
- Error on training data: $E(g \mid \mathcal{X}) = 0.0032$.
- Error on the remaining 93 points: 0.21 (much larger than on training data!)



Linear Basis Functions

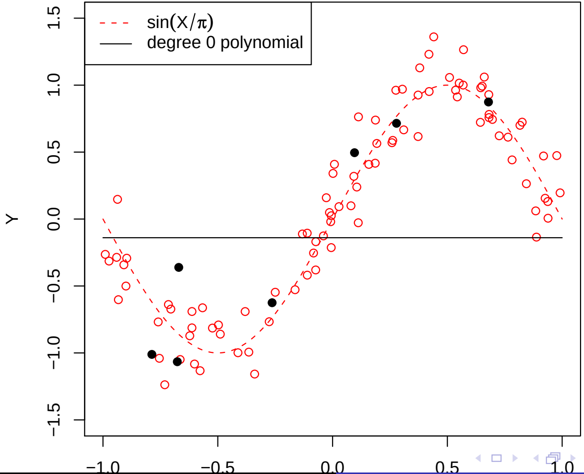
We can generalize linear regression using k basis functions $\phi_i(\mathbf{x})$,

$$g(\mathbf{x}) = \sum_{i=0}^{k} w_i \phi_i(\mathbf{x}),$$

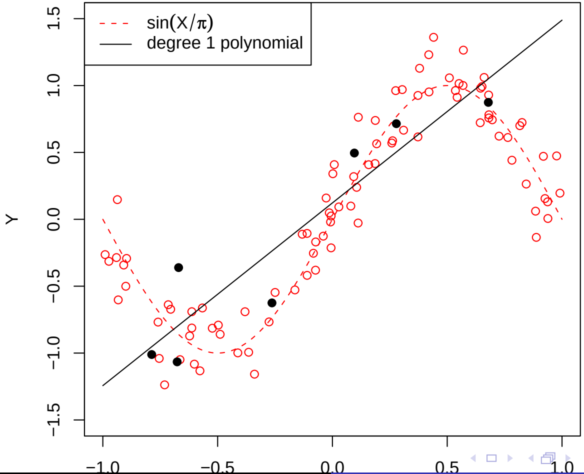
where usually $\phi_0(\mathbf{x}) = 1$.

- A common choice: $\phi_i(\mathbf{x}) = \mathbf{x}^i$ (polynomial basis).
- $\phi_i(\mathbf{x}^t)$ can be computed beforehand and w_i can be solved using linear algebra.
- In practice, there are lots of good software packages available that do the solving for you.
- Clearly, a high degree polynomial can represent a lower degree polynomial as a special case.
- Higher degree polynomial means larger hypothesis space or model family.

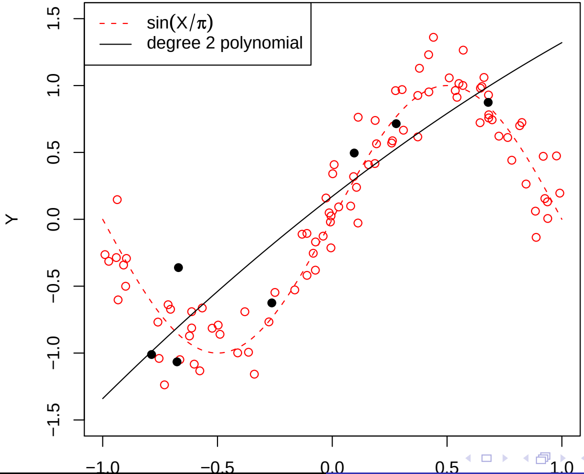
Polynomial Regressors



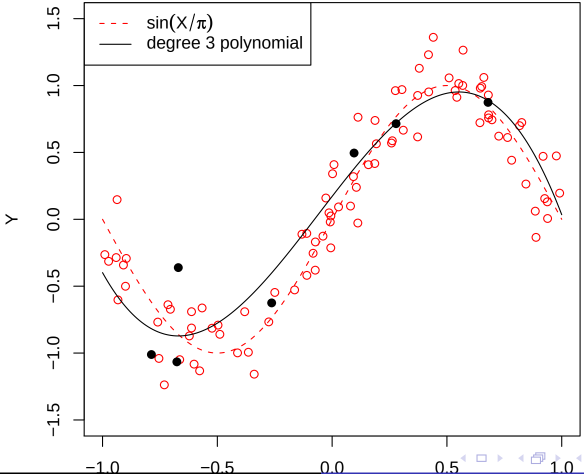
Polynomial Regressors



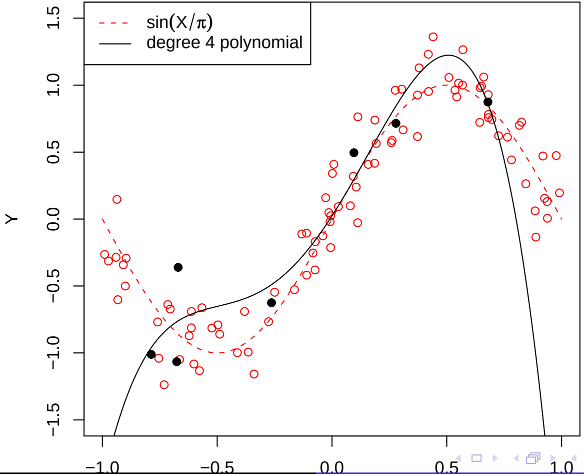
Polynomial Regressors



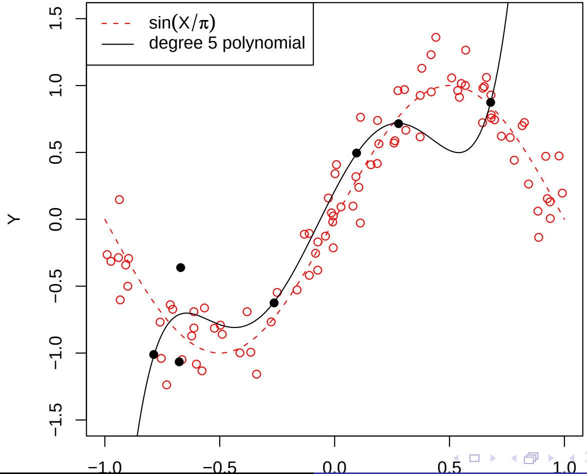
Polynomial Regressors



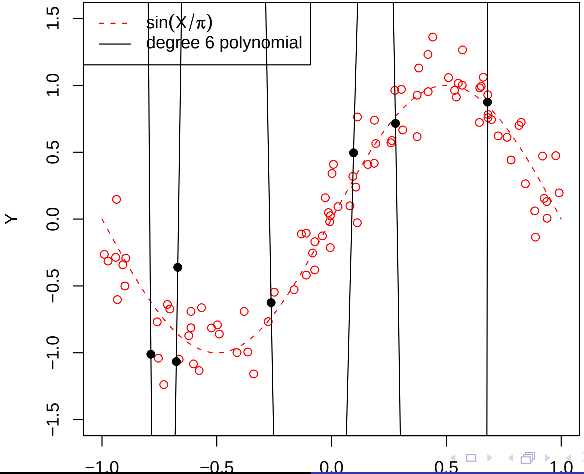
Polynomial Regressors



Polynomial Regressors



Polynomial Regressors



Polynomial Regressors

- E_{TRAIN} is the error in the training data. It decreases as model complexity increases.
- E_{TEST} is the error on the remaining 93 data points ("test set"). It has minimum at $k = 3$.

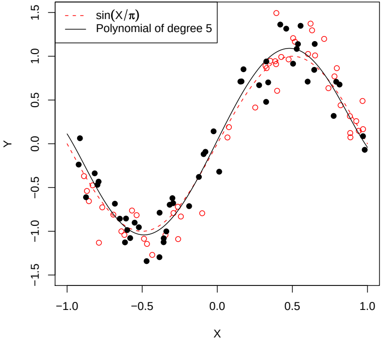
k	E_{TRAIN}	E_{TEST}	$g(x \mid w_0, \ldots, w_k) = \sum_{i=0}^{k} w_i X^i$
0	0.580	0.541	-0.14
1	0.077	0.294	$+0.12 + 1.37X$
2	0.076	0.275	$+0.17 + 1.33X - 0.18X^2$
3	0.057	0.057	$+0.17 + 2.22X - 0.35X^2 - 2.00X^3$
4	0.046	0.562	$+0.02 + 2.67X + 2.23X^2 - 3.19X^3 - 4.73X^4$
5	0.035	4.637	$+0.21 + 3.28X - 2.70X^2 - 11.88X^3 + 5.24X^4 + 15.82X^5$
6	0	10^6	$-5.86 + 57X + 186X^2 - 875X^3 - 1490X^4$ $+1634X^5 + 2412X^6$

Table: Polynomial regressors

Polynomial Regressors

N	E_{TRAIN}	E_{TEST}
7	0.0131	1.2187
10	0.0141	0.0821
15	0.0202	0.0761
20	0.0300	0.0511
25	0.0328	0.0507
30	0.0318	0.0573
35	0.0380	0.0494
40	0.0405	0.0484
45	0.0400	0.0476
50	0.0388	0.0473

Table: Effect of the size of the training data, $k = 5$.



Outline

1. Learning a Class from Examples
   - Introduction
   - Aldo and Family Car
   - PAC Learning and VC Dimension
2. Noise and Regression
   - Noise
   - Regression
   - Validation
3. Conclusion
   - About Supervised Learning
   - Better Basis Functions

Validation

- Error on training set:
    - Decreases as model becomes more complex.
    - Increases as number of data points grows.
- We want to minimize generalization error or error on test set:
    - Has a minimum at certain model complexity.
    - Decreases and approaches training set error as number of data points grows.
- How to minimize error on test set when we have no access to test set?

Validation

- To estimate generalization error, we need data unseen during training. We split the data in random as
  - training set (50%)
  - validation set (25%)
  - test set (25%)
- Train models of different complexities on training set. Pick a model complexity that gives smallest validation set error.
- Train model on combined training and validation set. Report test set error.

Validation

Example

- We are given 20 points from our sinusoidal curve data set.
- Divide the data in random to training (10), validation (5) and test (5) sets.
- Train regressors of different complexities on training set:

k	E_{TRAIN}	E_{VALID}
0	0.492	0.644
1	0.091	0.125
2	0.090	0.137
3	0.044	**0.041**
4	0.044	0.049
5	0.042	0.142
6	0.030	18.820
7	0.025	181.850
8	0.024	34.014
9	0	10^9

- Validation set error is minimized for the degree 3 polynomial ($k = 3$). Pick degree 3 polynomial.

Validation

Example

- Train degree 3 polynomial on 15 points (training+validation set) and report the results on the test set:

k	$E_{TRAIN+VALID}$	E_{TEST}
3	0.0378	0.0594

- If we would like to make predictions we should train on all 20 points (training+validation+test set). We know that the error on new data points should be approximately at most 0.0594.
- Training with all 20 points in fact gives slightly smaller error (0.0557) on 80 newly sampled data points.

Outline

1. Learning a Class from Examples
   - Introduction
   - Aldo and Family Car
   - PAC Learning and VC Dimension
2. Noise and Regression
   - Noise
   - Regression
   - Validation
3. Conclusion
   - About Supervised Learning
   - Better Basis Functions

Model Selection and Generalization

- Learning is ill-posed problem: data is not sufficient to find unique/correct solution.
- Inductive bias is needed; we need assumptions about the hypothesis class (model family) $\mathcal{H}$.
- Generalization: how well model performs on new data.
- Overfitting: $\mathcal{H}$ more complex than C or f.
- Underfitting: $\mathcal{H}$ less complex than C or f.
- Triple trade-off (Diettrich 2003):
    - complexity of $\mathcal{H}$;
    - amount of training data; and
    - generalization error on new data.

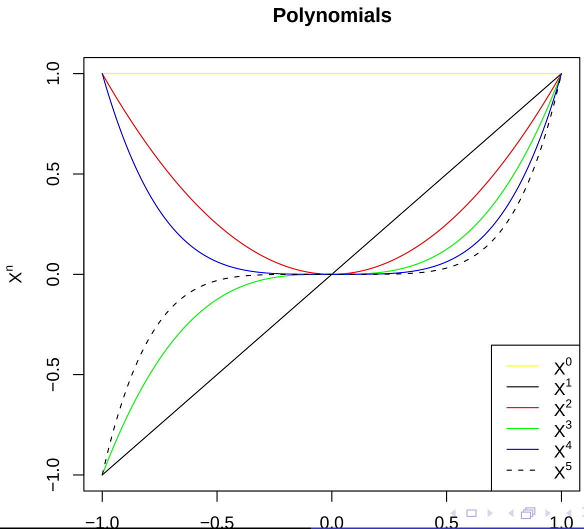
Dimensions of a Supervised Learner

1. Model: $g(\mathbf{x} \mid \theta)$.
2. Loss function: $E\left(\theta \mid \mathcal{X}\right) = \frac{1}{N}\sum_{t=1}^{N} L\left(r^t, g(\mathbf{x}^t \mid \theta)\right)$.
3. Optimization procedure: $\theta \leftarrow \arg\min_\theta E\left(\theta \mid \mathcal{X}\right)$.

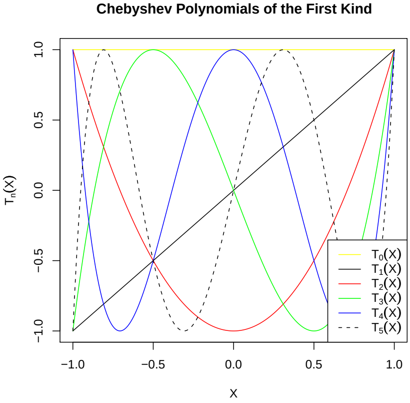
Outline

1. Learning a Class from Examples
   - Introduction
   - Aldo and Family Car
   - PAC Learning and VC Dimension
2. Noise and Regression
   - Noise
   - Regression
   - Validation
3. Conclusion
   - About Supervised Learning
   - Better Basis Functions

Polynomial Basis



Chebyshev Polynomials of the First Kind



$T_0(X) = 1$ $\qquad$ $T_3(X) = 4X^3 - 3X$

$T_1(X) = X$ $\qquad$ $T_4(X) = 8X^4 - 8X^2 + 1$

$T_2(X) = 2X^2 - 1$ $\qquad$ $T_5(X) = 16X^5 - 20X^3 + 5X$

Chebyshev Polynomials of the First Kind

- Chebyshev Polynomials are orthogonal polynomials in $X \in [-1, 1]$.
- Def.: $T_n(\cos\theta) = \cos n\theta,\ n \in \{0, 1, \ldots\}$.
- Recurrence relation: $T_{n+2}(x) = 2xT_{n+1}(x) - T_n(x)$.
- Chebyshev Polynomials are useful in numerical analysis:
    - $\max T_n(x) = +1$, $\min T_n(x) = -1$. (X^n basis also satisfies this in $X \in [-1, 1]$.)
    - The maxima and minima are spread reasonably uniformly over $[-1, 1]$. (Comparing, in X^n basis the maxima and minima are only in $X = -1$ and $X = +1$.)
    - In least squares regression, the Chebyshev basis is analytically equivalent but numerically much more robust than the commonly used X^n basis especially for larger (> 10) degrees.

$$T_0(X) = 1;\ T_1(X) = X;\ T_2(X) = 2X^2 - 1;\ T_3(X) = 4X^3 - 3X;$$
$$T_4(X) = 8X^4 - 8X^2 + 1;\ T_5(X) = 16X^5 - 20X^3 + 5X;\ \ldots$$

k	E_{TRAIN}	E_{TEST}	$g(x \mid w_0, \ldots, w_k) = \sum_{i=0}^{k} w_i T_i(X)$
0	0.580	0.541	$-0.14 T_0(X)$
1	0.077	0.294	$+0.12 T_0(X) + 1.37 T_1(X)$
2	0.076	0.275	$+0.08 T_0(X) + 1.33 T_1(X) - 0.09 T_2(X)$
3	0.057	0.057	$-0.01 T_0(X) + 0.72 T_1(X) - 0.18 T_2(X) - 0.50 T_3(X)$
4	0.046	0.562	$-0.64 T_0(X) + 0.28 T_1(X) - 1.25 T_2(X) - 0.80 T_3(X)$
			$-0.59 T_4(X)$
5	0.035	4.637	$+0.83 T_0(X) + 4.26 T_1(X) + 1.27 T_2(X) + 1.97 T_3(X)$
			$+0.65 T_4(X) + 0.99 T_5(X)$
6	0	10^6	$+282.4 T_0(X) + 422.6 T_1(X) + 478.9 T_2(X) + 291.8 T_3(X)$
			$+266.0 T_4(X) + 102.1 T_5(X) + 75.3 T_6(X)$

Table: Chebyshev regressors; compare the magnitude of the terms to the X^n basis.

$$\begin{array}{ll} T_0(X) = 1 & T_3(X) = 4X^3 - 3X \\ T_1(X) = X & T_4(X) = 8X^4 - 8X^2 + 1 \\ T_2(X) = 2X^2 - 1 & T_5(X) = 16X^5 - 20X^3 + 5X \\ & T_6(X) = 32X^6 - 48X^4 + 18X^2 - 1 \end{array}$$

Conclusion

- No problem session this week, next problem session on 28 September.
- This week's problem sheet contains a small data analysis task (for 28 September). [Will be in the web later today, hopefully.]
- Next lecture on 25 September: Bayesian Decision Theory, Alpaydin (2004) Ch 3.