機械学習(3) モデルの複雑さと汎化

情報科学類　佐久間　淳

再掲

線形回帰の定式化

- 例
  - (身長,体重)から平均寿命を予測する
  - (年齢,年収)から年間支出額を予測する
- 線形回帰

$$t = w_0 + \sum_{i=1}^{D} w_i x_i$$

先頭に1を付け加える

- データを

$$\begin{pmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \vdots \\ \boldsymbol{x}_N^T \end{pmatrix} = \begin{pmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1D} \\ 1 & x_{21} & x_{22} & \cdots & x_{2D} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{N1} & x_{N2} & \cdots & x_{ND} \end{pmatrix}$$

とすると

$$t = \sum_{i=0}^{D} w_i x_i = \boldsymbol{w}^T \boldsymbol{x}$$

切片を気にせず
内積一発で書けて便利

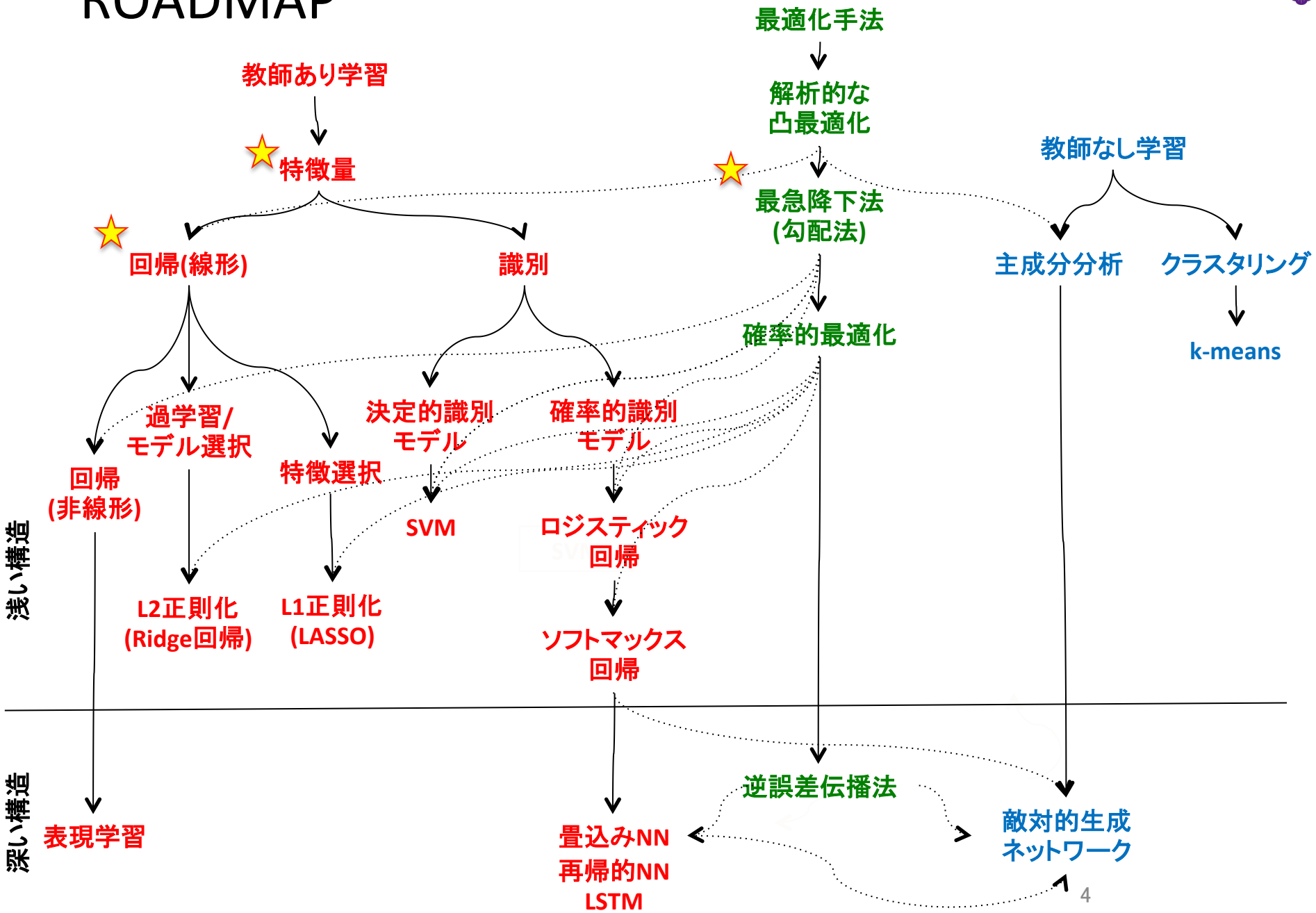
再掲

二乗誤差の最小化

- 二乗誤差 $E(\boldsymbol{w}) = \sum_{i=1}^{N} (t_i - \boldsymbol{w}^T \boldsymbol{x}_i)^2$

- 二乗誤差の最小化
  - $E(\boldsymbol{w})$は$\boldsymbol{w}$ に関して下に凸な関数
    - 下に凸であることは前提としていい(説明は省略)
  - $\frac{\partial E}{\partial \boldsymbol{w}} = 0$ になるような$\boldsymbol{w}$を求めればよい
  - $\boldsymbol{w} = (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{t}$ (導出は演習)

ROADMAP


教師あり学習
特徴量
回帰(線形)
識別
最適化手法
解析的な
凸最適化
最急降下法
(勾配法)
教師なし学習
主成分分析
クラスタリング
k-means
確率的最適化
過学習/
モデル選択
回帰
(非線形)
特徴選択
決定的識別
モデル
確率的識別
モデル
SVM
ロジスティック
回帰
L2正則化
(Ridge回帰)
L1正則化
(LASSO)
ソフトマックス
回帰
浅い構造
深い構造
表現学習
畳込みNN
再帰的NN
LSTM
逆誤差伝播法
敵対的生成
ネットワーク

質問・コメント(特徴量)

- スケーリングをすると、影響の強さに差が生じてしまうことが分かったが、精度などに影響が出ることはあるか?
  - 手法にもよるのですが、線形回帰の場合、影響はでません。多項式回帰の場合は、場合によってはスケーリングしないほうがいい場合があるのかもしれません
- 特徴と特徴量は同じですか?
  - 同じです。あえて言えば、特徴は変数、特徴量はその値をさしている雰囲気があります

質問･コメント(特徴量)

- HLACは位置不変性はあるが回転不変性はないのでは？
  - 位置不変性と加法性があります。回転不変性はありません。スライドにはかいてありませんが、口頭でそのように言ってしまった？
- Bag of words「推定したい文書すべてについてBoWを出した後、BoWを使った各文書の単語出現回数ベクトルを出す」という感覚です。本来のBoWの定義はこれと異なる？
  - 質問と授業で説明した定義の違いがよくわからないですが、普通はBoWといった場合は出現回数はあまり気にしないようです

質問・コメント(回帰・誤差)

- 重回帰は単回帰よりMSEを小さくすると思うのですが、重回帰よりもMSEを小さくする方法ってありますか?
  - 単にMSEを小さくするだけなら、あります（今回やります）。今日の話題は、「それだけではダメ」です
- 誤差を表すときに、授業では二乗誤差の和を考えていたが、自然対数や累乗のような誤差の表しかたをすることは許されているのか?
  - もちろんやるのは自由です。ただそれが簡単に解けるかどうかは別です。回帰が二乗誤差を考える理由は今日やります。それ以外の誤差(損失関数と言いいます)については、来週から2,3週にわたって扱います

質問・コメント(最急降下法)

- $\boldsymbol{x}^{t+1} \leftarrow \boldsymbol{x}^{t} - \nabla_{\boldsymbol{x}} f$　　$\nabla_{\boldsymbol{x}} f$は、　　によって進む大きさが変わりそうですが大丈夫なのですか?　　$\boldsymbol{x}^{t+1} \leftarrow \boldsymbol{x}^{t} - \eta \nabla_{\boldsymbol{x}} f$
  - 正しくは　　ですね
  - ステップサイズηの適切な決め方は、fの性質にも依存し、簡単には結論が出ない問題です (第四回で少しやります)
  - 更新毎に、勾配方向にどれだけ進むかを最適化する方法や、tに依存してηを減少させる方法などがあります

質問・コメント（凸関数）

- 凸関数の理解は出来たが、凸集合の概念が分からない
  - 黒板で
- 凸関数となるモデルの見つけ方はありますか?
  - 「あるfが凸関数かどうか？」を示すには、「そのfが凸関数の定義を満たすかどうか？」を証明することになります
  - fが特殊な形をしている場合は、もう少し楽にわかることがあります。$f(\boldsymbol{x}) = \boldsymbol{x}^T \boldsymbol{Q}\boldsymbol{x} + c\boldsymbol{x}$ の場合、Qの固有値が全て正なら、fは(狭義)凸関数です

質問・コメント（その他）

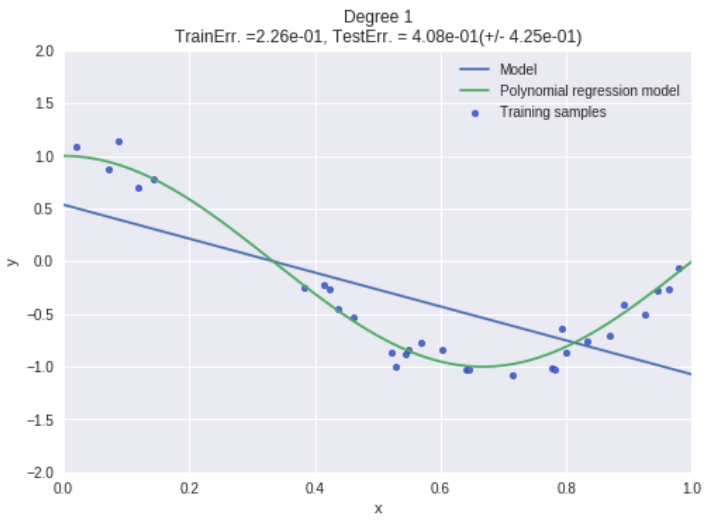
- 行列を太文字で書いてないタイプの数学の本を見たのですが、そこは自由なのでしょうか?
- スライドの文字の表記もベクトルとスカラーを分けてほしいです
  - 定義さえすれば、自由です。この授業では、太文字にしてください。スライドも、必ず書き分けています
- スライド25枚目の勾配にカンマは必要なのか?
  - 行ベクトルの場合カンマを入れることが多いようです(が必須でない)

質問・コメント（その他）

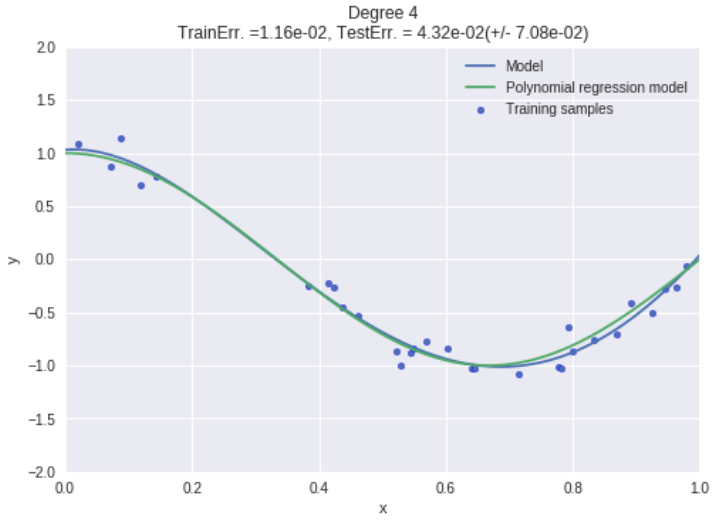
- 3B棟など、第3エリア内の教室にできませんか?
  - 総合講義とかぶっているので、大教室がないそうです
- 課題を返却する際、可能ならば2018,2017,2016,2015ぐらいで分けてほしいです。
  - その時間中に課題を返すために、TAさんは授業時間中に採点しています。時間的余裕があればそうします
- 本日の授業も楽しかったです。いつか文書を解析して、本の面白さなどを定量化出来たら良いなと思いました。
- 手書き必須は辛いです。

データの非線形性

- データの分布が非線形なら？
- 線形モデルより高い表現能力による予測を行いたい　(緑線：正解モデル、青線：学習したモデル)



単回帰



多項式回帰
(多項式特徴による単回帰)

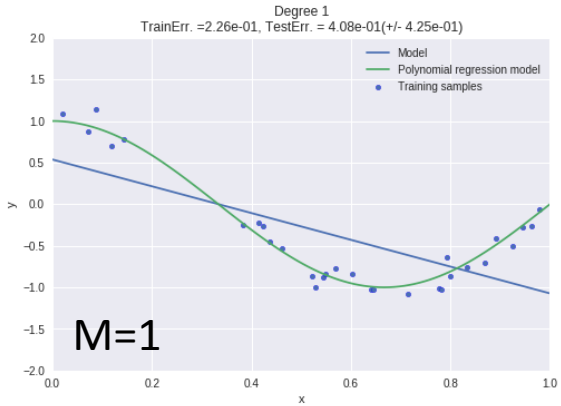
多項式回帰

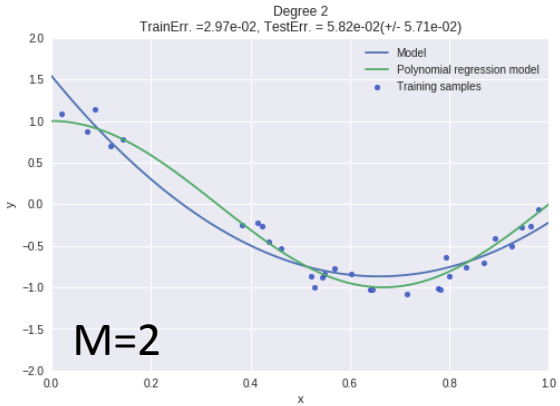
- 線形回帰モデル
  - 1次元特徴　$t = w_0 + w_1 x$
  - D次元特徴　$t = w_0 + w_1 x_1 + \ldots + w_D x_D = w_0 + \sum_{d=1}^{D} w_d x_d$
- M次多項式回帰モデル
  - 1次元特徴　$t = w_0 + w_1 x + w_2 x^2 + \ldots + w_M x^M$
  - D次元特徴　$t = w_0 + \sum_{m=1}^{M} \sum_{i=d}^{D} w_{md} x_d^m$

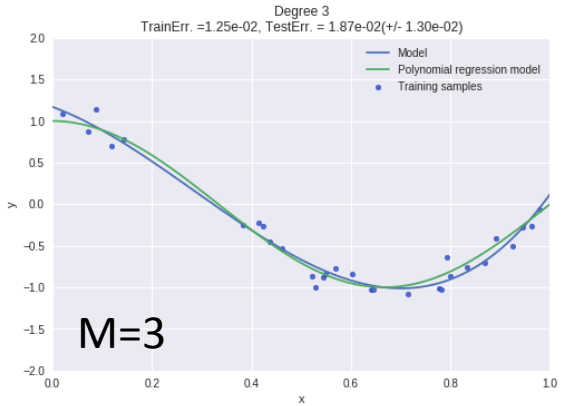
1次元多項式特徴量による回帰

- 特徴量関数　$\boldsymbol{\phi} : \mathbb{R} \to \mathbb{R}^{M+1}$
  - 多項式特徴量 $\boldsymbol{\phi}(x) = (x^0, x^1, x^2, \ldots, x^M)$
- 多項式特徴量による線形回帰モデル

$$t = \boldsymbol{w}^T \boldsymbol{\phi}(x)$$







多次元多項式回帰

- 特徴量ベクトル⇒(φによる)特徴量ベクトル

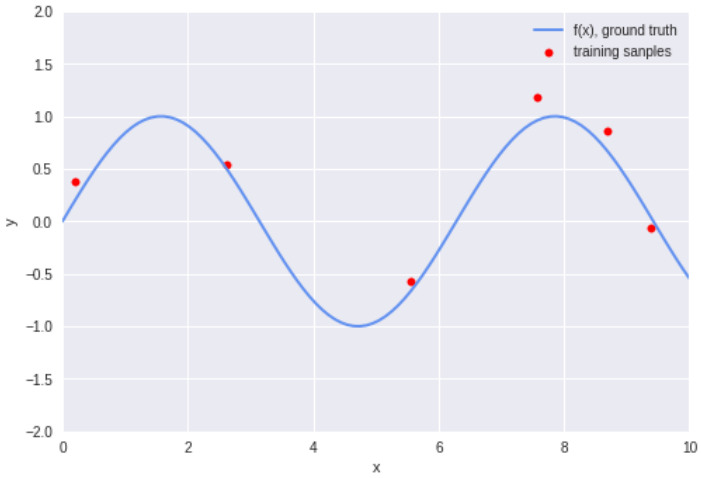
$$\boldsymbol{x}^T = (1, 2, 5) \Rightarrow \boldsymbol{\phi}(\boldsymbol{x})^T = (1, 2, 5, 1^2, 2^2, 5^2, 1^3, 2^3, 5^3)$$

- $\boldsymbol{x}$ の代わりに特徴量ベクトル$\boldsymbol{\phi}$ で回帰

$$\boldsymbol{X} = \begin{pmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \vdots \\ \boldsymbol{x}_N^T \end{pmatrix} \qquad \boldsymbol{\Phi} = \begin{pmatrix} \boldsymbol{\phi}(\boldsymbol{x}_1)^T \\ \boldsymbol{\phi}(\boldsymbol{x}_2)^T \\ \vdots \\ \boldsymbol{\phi}(\boldsymbol{x}_N)^T \end{pmatrix}$$

$$\boldsymbol{w} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{t} \qquad \boldsymbol{w} = (\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^T\boldsymbol{t}$$

演習:
多項式単回帰の二乗誤差



3.1　多項式単回帰の二乗誤差

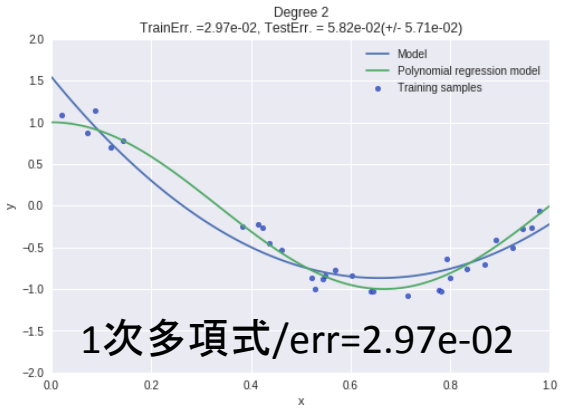
polynomialRegSquaredError.ipynb を実行して以下の問いに答えなさい。

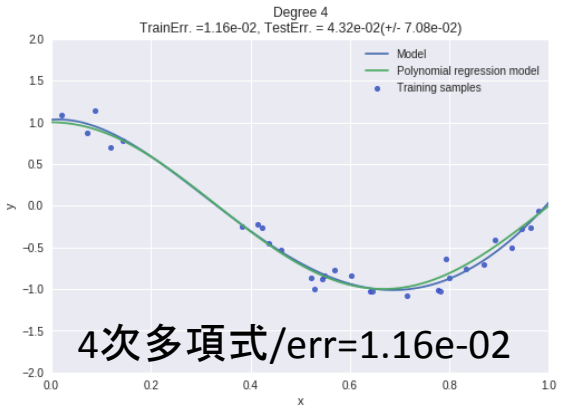
$y = f(x) + N(0,1)$ に従い $i = 1, \ldots, 6$ の 6 点のデータを生成した。ここで, $f(x) = \sin(x)$ であり, $N(0,1)$ は平均 0 分散 1 の正規分布から生成した正規乱数である。$i = 1, ..., 6$ のサンプル (y_i, x_i) から多項式回帰によって関数 f を推定することを考える。

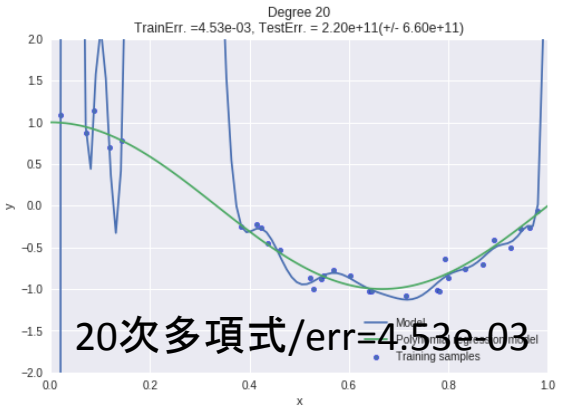
1. プログラムは、生成された多項式特徴量の値を表示している。多項式特徴量を求める式を示せ
2. 最小の二乗誤差を与える多項式の次数はいくつか
3. 二乗誤差が次数 d=5 でほぼゼロになるのはなぜか？

モデルの複雑さ

- 多項式特徴量の導入⇒リッチな表現の回帰を実現
- どの程度リッチにすればいいのか？



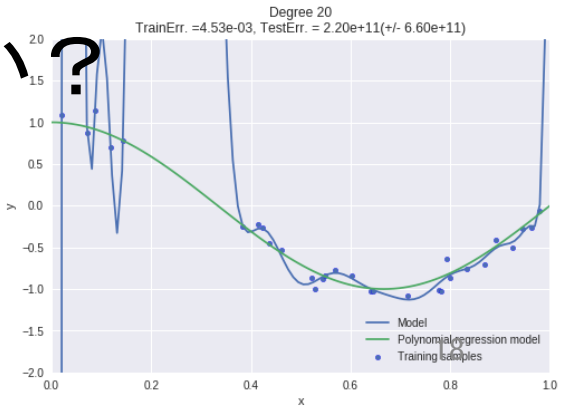




- 20次多項式回帰の二乗誤差は4次多項式回帰より一桁小さい
- でもこれが最良とは思えない
- どうすれば良いのか？

オッカムの剃刀

- ある事柄を説明するためには、必要以上に多くの実体を仮定するべきでない（14世紀の哲学者 Occam)
  - Entities should not be multiplied beyond necessity
  - (ML的には…) あるデータが与えられたとき、モデルを複雑にすればするほど、そのデータをうまく説明できる。しかし、そのようなモデルは、不必要に複雑なモデルであり、計算が困難であるばかりでなく、過去のデータに過剰に適合してしまい、未来のデータを説明できなくなってしまう
- 適切な複雑さをどのように選ぶか？



確率変数、確率分布

- 確率変数:ある変数の値をとる確率が存在する変数
- 確率分布: 確率変数の各値に対して、その値の起こりやすさ(確率)への対応
- 離散確率変数の例: コインの裏(=0)、表(=1) x
  - $P(x=0)=0.5,\ P(x=1)=0.5$
  - 確率変数xは$p=0.5$のベルヌーイ分布に従う

$$f(x;p) = p^x(1-p)^{(1-x)}$$

- 連続確率変数の例: 落ちた木の葉の木からの距離 x

$$P(1 \leq x \leq 2) = \int_1^2 N(x;0,1)$$

  - $N(x;0,1)$は平均0, 分散1の正規分布　(確率密度関数)
  - 確率変数xは正規分布$N(x;0,1)$に従う

期待値

- 期待値: 確率変数の値を, その確率$p(x)$で重み付き平均した値

離散確率変数Xの場合

$$E[X] = \sum_{i=1}^{\infty} x_i P(X = x_i)$$

連続確率変数xの場合

$$E[x] = \int_{-\infty}^{\infty} x p(x) dx$$

- 無限個のサンプルが得られれば、期待値を計算できる
- しかし通常は有限個のサンプルしか得られない
- 有限個のサンプルによる標本平均を期待値の代わりに用いる

$$\bar{X}_N = \frac{1}{N} \sum_{i=1}^{N} x_i$$

- ある母集団から無作為抽出されたサンプルによる標本平均はサンプル数を大きくすると母集団の期待値に近づく(大数の法則)

$$\epsilon > 0, \quad \lim_{N \to \infty} \Pr[|\bar{X}_N - E[X]| > \epsilon] = 0$$

我々は何を知ろうとしているのか

- 世界には…
  - データを生成する分布が存在 $p(\boldsymbol{x}, t)$
    - ワインの成分とそのワインのqualityの分布
  - モデルが存在 $f : \mathbb{R}^D \to \mathbb{R}$
    - ワインの成分からワインのqualityへの写像
  - しかしこれらは全て未知
- 我々が観測できるのは…
  - ワインの成分とそのワインのqualityのサンプル
    $(\boldsymbol{x}_1, t_1), (\boldsymbol{x}_2, t_2), \ldots, (\boldsymbol{x}_N, t_N),$
  - 仮定：qualityには(未知の)ノイズが乗る $t_i = f(\boldsymbol{x}_i) + \epsilon$
- この条件で、f に近いモデル$\hat{f}$ を推定したい

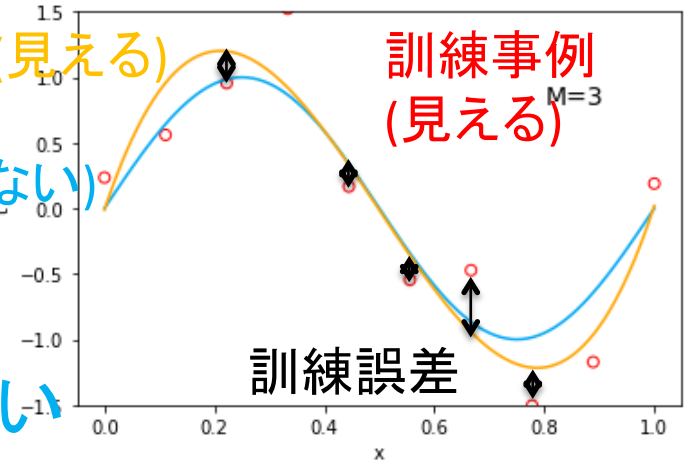
訓練サンプルとテストサンプル

- 手持ちのN個のサンプル $(\boldsymbol{x}_1, t_1), (\boldsymbol{x}_2, t_2), \ldots, (\boldsymbol{x}_N, t_N),$
- 非常に正確だが意味のない学習
  - x_iを与えられたらt_iを返すハッシュ関数を構築する
  - 二乗誤差は常にゼロ
  - しかし手持ちサンプル以外の問い合わせに対応できない
- 訓練用とテスト用のサンプルに分ける

$$\underbrace{(\boldsymbol{x}_1, t_1), (\boldsymbol{x}_2, t_2), \ldots}_{\text{訓練事例 } X_{\mathrm{tr}}}, \underbrace{(\boldsymbol{x}_N, t_N)}_{\text{テスト事例 } X_{\mathrm{ts}}},$$

  - 訓練事例で学習して、テスト事例で性能をチェック
  - 訓練事例＝解答付き演習問題 (丸暗記すれば100点がとれる)
  - テスト事例=期末試験 (演習問題とは異なる問題で理解をチェック)

訓練誤差



- モデル：$t = f(\boldsymbol{x})$　，知りたい
- 訓練事例：$X_{tr} = \{(\boldsymbol{x}_i, t_i)\}$，見える
- 学習した回帰モデル：$\hat{t}_i = \hat{f}(\boldsymbol{x}_i)$　，見える
- 訓練誤差 (=これまで二乗誤差と呼んでいたもの)
  - 訓練事例で得た回帰モデルの，訓練事例における誤答率(解答付き練習問題の正答率)
  - 訓練事例数で割ることに注意

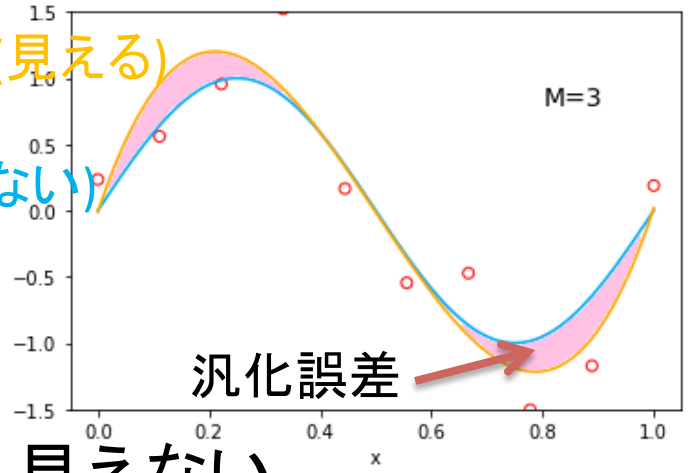
$$\mathrm{TrainingErr} = \frac{1}{|X_{\mathrm{tr}}|} \sum_{(\boldsymbol{x}_i, t_i) \in X_{\mathrm{tr}}} (t_i - \hat{f}(\boldsymbol{x}_i))^2$$

訓練サンプルの目標値　予測

汎化誤差

学習した回帰曲線(見える)

真の回帰曲線(見えない)



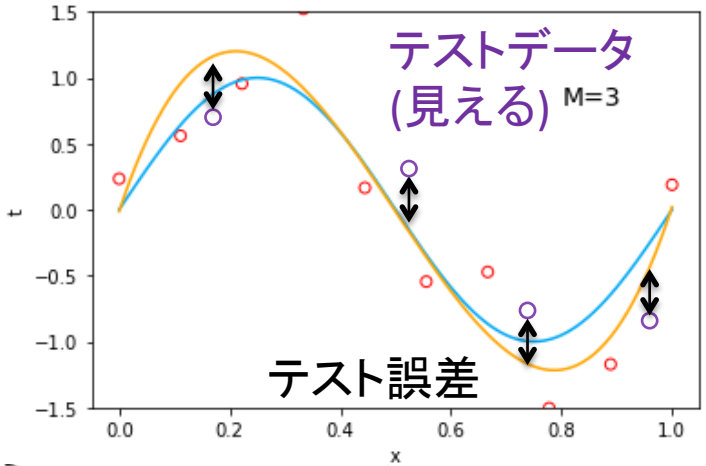
- 汎化誤差:
  - $p(\boldsymbol{x}, t)$　　データ生成分布, 見えない
  - 真のモデル(推定対象), $t = f(\boldsymbol{x})$,見えない
  - 学習した回帰モデル: $\hat{t}_i = \hat{f}(\boldsymbol{x}_i)$　　, 見える

$$\mathrm{GeneralizationErr} = \iint \underbrace{(t - \hat{f}(\boldsymbol{x}))^2}_{\text{予測}} \underbrace{p(\boldsymbol{x}, t)}_{\text{データ生成分布(未知)}} d\boldsymbol{x} dt$$

正解(未知)

  - 本当はこの期待値が欲しい　　,
  - しかし、$p(\boldsymbol{x}, t), f(\boldsymbol{x})$ が得られないので評価できない

テスト誤差



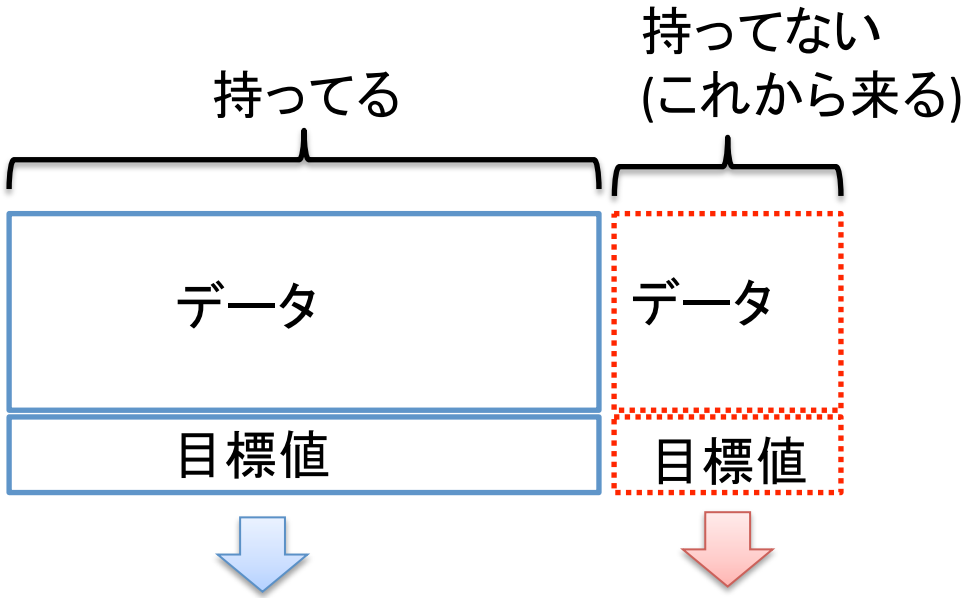
- モデル: $t = f(\boldsymbol{x})$, 知りたい
- テスト事例: $X_{ts} = \{(\boldsymbol{x}_i, t_i)\}$, 見える
- 学習した回帰モデル: $\hat{t}_i = \hat{f}(\boldsymbol{x}_i)$, 見える
- テスト誤差:
  - 訓練事例で学習した回帰モデルの，テスト事例における誤答率(期末テストの正答率)
  - テスト事例数で割ることに注意

$$\mathrm{TestErr} = \frac{1}{|X_{\mathrm{ts}}|} \sum_{(\boldsymbol{x}_i, t_i) \in X_{\mathrm{ts}}} (t_i - \hat{f}(\boldsymbol{x}_i))^2$$

有限個のサンプル平均

テストサンプルの目標値　予測

誤差の評価

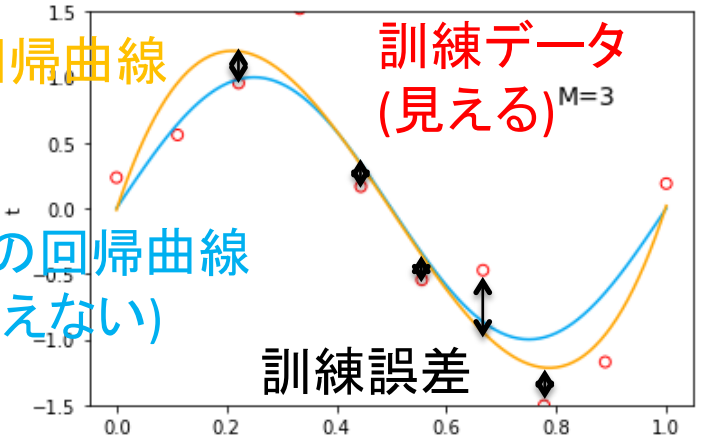


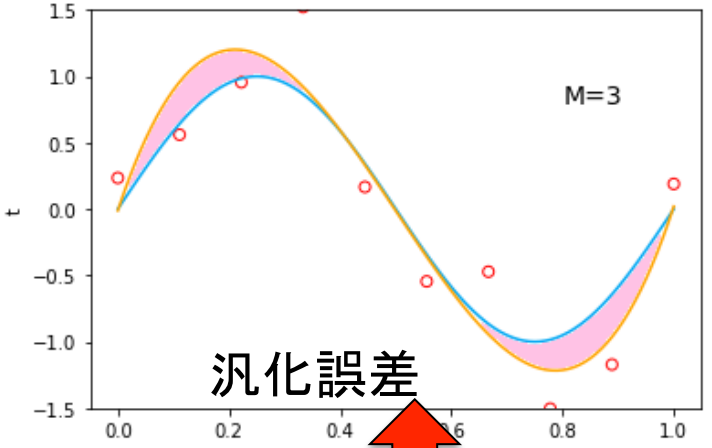
訓練誤差　　テスト誤差

$$\mathrm{GeneralizationErr} = \iint (t - \hat{f}(\boldsymbol{x}))^2 p(\boldsymbol{x}, t) d\boldsymbol{x} dt$$

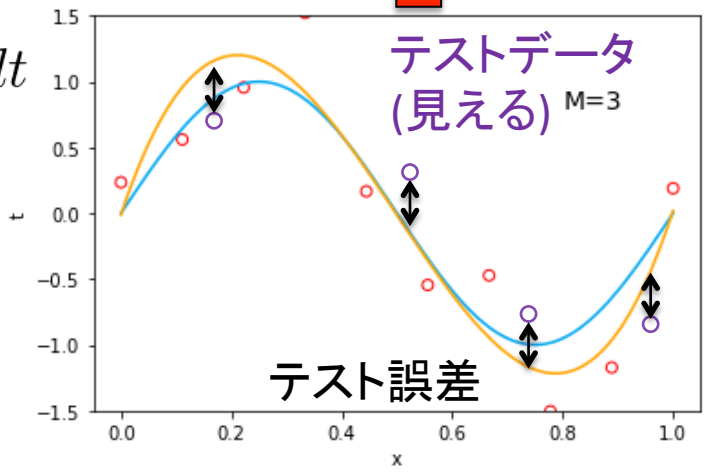
汎化誤差の有限サンプル近似=テスト誤差

$$\mathrm{TestErr} = \frac{1}{|X_{\mathrm{ts}}|} \sum_{(\boldsymbol{x}_i, t_i) \in X_{\mathrm{ts}}} (t_i - \hat{f}(\boldsymbol{x}_i))^2$$



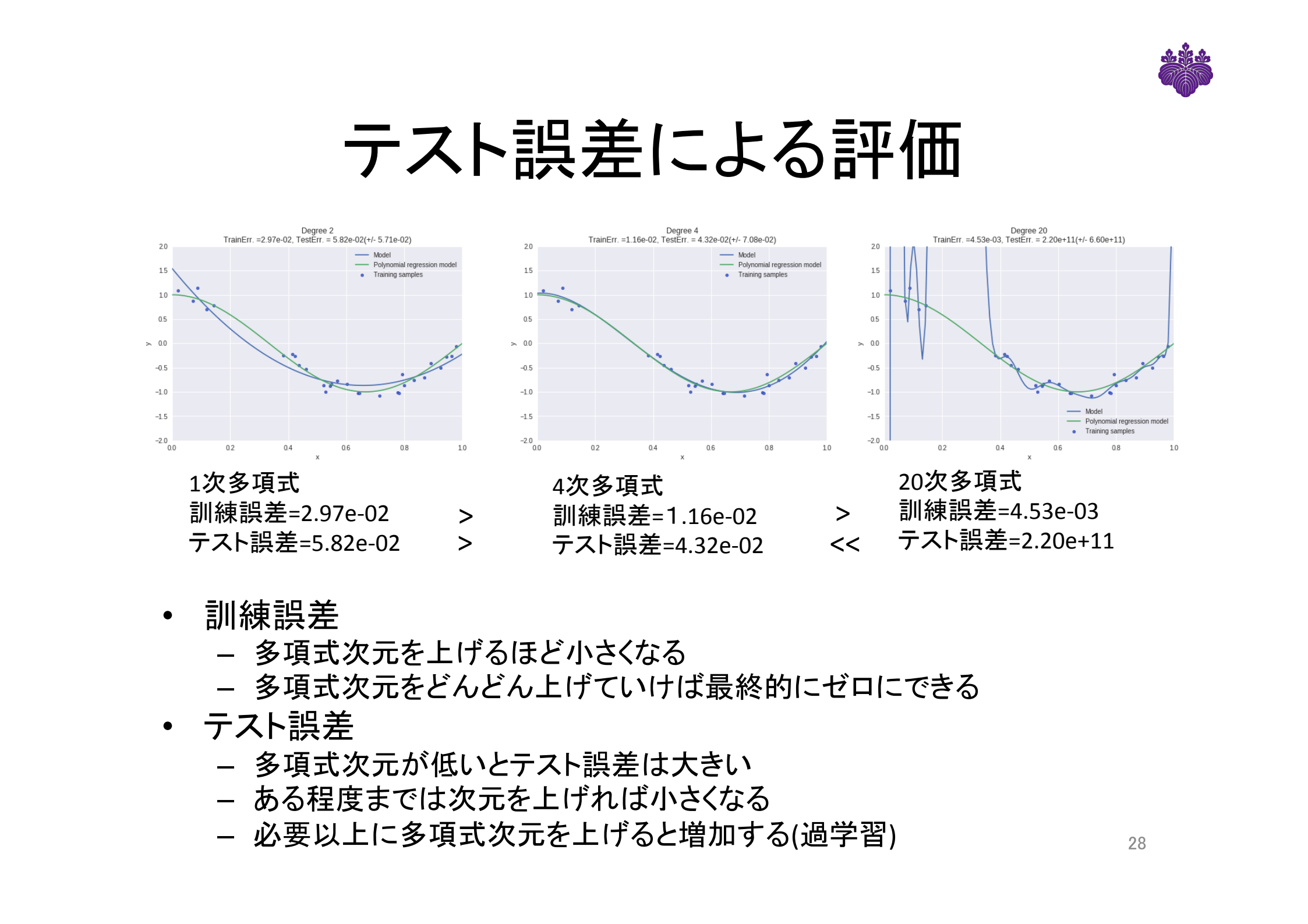


サンプル近似



誤差Q&A

- 事例の発生分布p(x,t)は具体的にはどういうものか？,汎化誤差のp(x,t)は確率密度関数？
  - 事例(x,t)がどのような確率分布から発生したか，を表しています
- 汎化誤差のサンプル近似がテスト誤差なのはなぜ？
  - テストデータは，データの発生分布p(x,t)から生成されていると考えているので，p(x,t)における期待値を，サンプル近似していることになります
- 汎化誤差を実際に用いる局面はどのようなものか？汎化誤差自体に，「何かの概念を説明するのに用いる」以外の用途はないのか
  - 汎化誤差を直接計算することはできないのですが，汎化誤差の上限を評価することはできる場合があって，これはある種の学習機の性能保証として使えます．汎化誤差の上界は機械学習のコアになる理論的概念ですが，この授業では範囲をこえるため扱いません．興味があれば研究室まで。



テスト誤差による評価

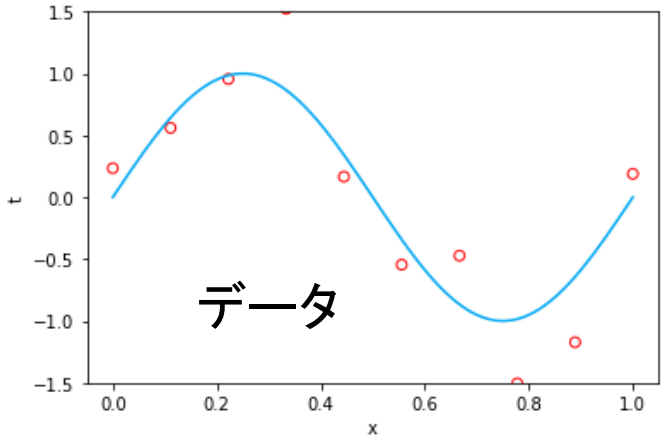
1次多項式
訓練誤差=2.97e-02 >
テスト誤差=5.82e-02 >

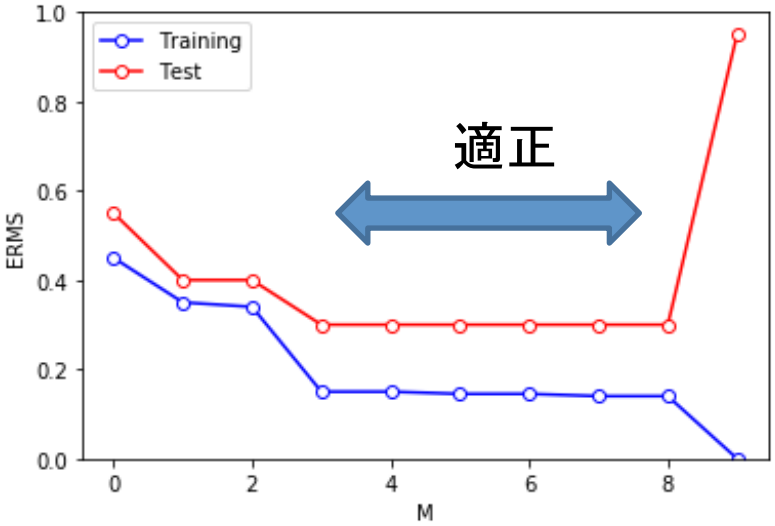
4次多項式
訓練誤差=1.16e-02 >
テスト誤差=4.32e-02 <<

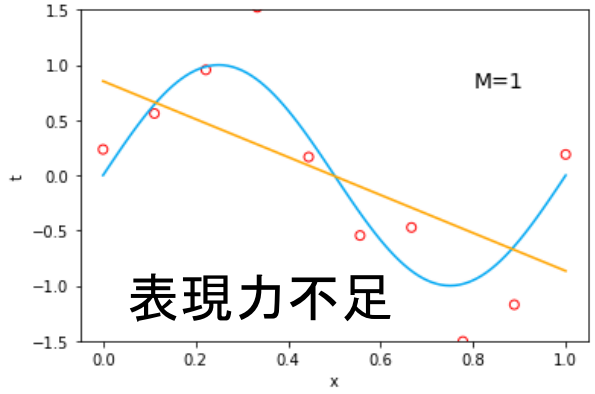
20次多項式
訓練誤差=4.53e-03
テスト誤差=2.20e+11

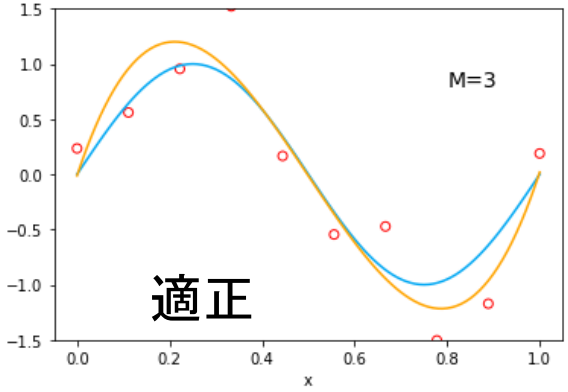
- 訓練誤差
  - 多項式次元を上げるほど小さくなる
  - 多項式次元をどんどん上げていけば最終的にゼロにできる
- テスト誤差
  - 多項式次元が低いとテスト誤差は大きい
  - ある程度までは次元を上げれば小さくなる
  - 必要以上に多項式次元を上げると増加する(過学習)

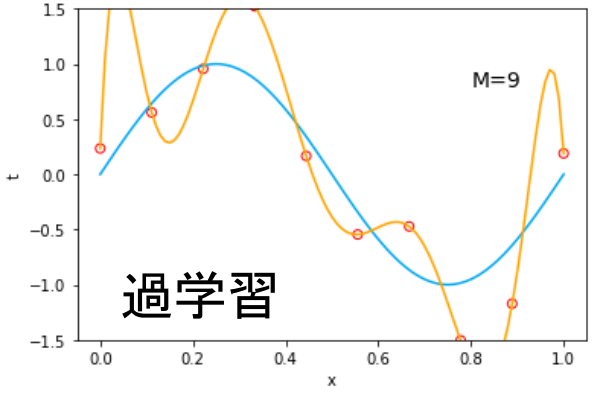
過学習











訓練誤差大 ← → 訓練誤差小

汎化誤差大 ← 汎化誤差小 → 汎化誤差大

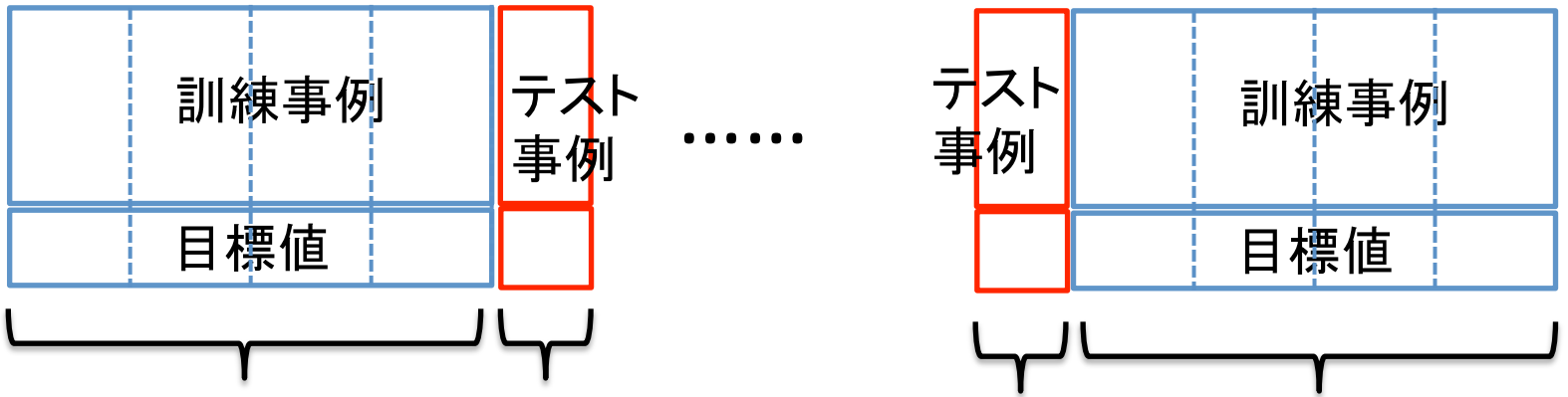
モデルの複雑さを適切に制御する必要がある

モデルの複雑さをどう制御するか

1. モデル選択
   - 様々な複雑さを持つモデルを用意しておく
   - それらのモデルを全部学習させ、最小のテスト誤差を与えるモデルを選択する
   - 結果として適度な複雑さのモデルが選択される
2. 正則化
   - 十分に複雑なモデルを用意しておく
   - モデルの複雑さを抑制する仕組みを誤差関数に埋め込む
   - モデルの複雑さを変化させながら学習させ、最小のテスト誤差を与える複雑さを選択する
   - 結果として適度な複雑さのモデルが学習される

- 何れにせよ、テスト誤差で評価することが大事

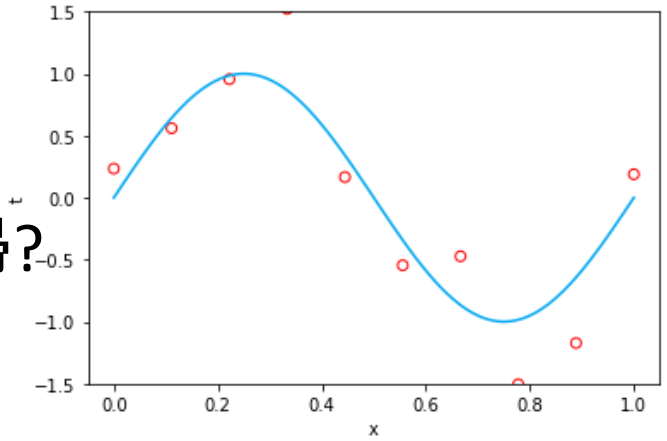
K-fold交差検証によるテスト誤差の評価



- 事例をk個に分割する
- For i=1,…,k
  - i番目の分割をテスト事例、残る $K-1$ 個を訓練事例としテスト誤差を評価
- 全foldのテスト誤差を平均し、これを汎化誤差の推定値として出力
- 単に2分割し、片方(訓練事例)で学習、片方(テスト事例)でテスト誤差評価するだけではダメなのか？
  - 学習結果は訓練事例に依存する
  - 「たまたま」偏りの大きい事例が訓練事例に集中すると、学習モデルも偏りが大きくなる
  - 様々な分割において訓練とテスト誤差評価を行い偏りをなるべく少なくする

交差検証を使ったモデル選択

- モデル選択
  - データがある
  - M=1,2,…,9, 何次多項式で回帰?



- 交差検証によるモデル選択
  1. For M=1,…,9
     1. M次多項式回帰でモデリング
     2. k-fold交差検定でテスト誤差を評価
  2. 最小のテスト誤差を与えるMを採用
  - 全体でMK回の学習・テスト誤差評価が必要

演習　多項式単回帰の訓練/テスト誤差

3.2　多項式単回帰の訓練誤差とテスト誤差

polynomialReg_testErr.ipynb を実行して以下の問いに答えなさい。$y = \cos(1.5\pi x) + 0.1 * N(0,1)$ に従い 30 の訓練事例を生成した。ここで $N(0,1)$ は平均 0 分散 1 の正規分布から生成した正規乱数である。プログラムの実行結果は、これらの訓練事例について、次数 degree を 1 から 20 まで変化させながら、多項式単回帰を行い、訓練誤差 TrainErr. およびテスト誤差 TestErr（とその標準偏差）を表示している。

1. 訓練誤差を最小にする多項式特徴量は何次元のものか
2. テスト誤差を最小にする多項式特徴量は何次元のものか
3. 訓練誤差は小さいが、テスト誤差が大きくなっている状態をなんと呼ぶか
4. もっとも優れた多項式特徴量は何次元か

演習

3.3　バイト君とコンビニ店長

あるコンビニエンスストアの店長は、ビールの売り上げ本数 $y_i, (i = 1, ..., T)$ が毎日変化するため、これを予測し、仕入れの数量を調整したいと考えた。予測のために、i 日の日付 $x_{i1}(1, 2, \ldots, 30)$, 気温 x_{i2}(度)、湿度 x_{i3}(%)、と, その日に売れたビールの本数 y_i を記録した．　$\boldsymbol{x}_i^T = (x_{i1}, x_{i2}, x_{i3})$ と定義する．$(\boldsymbol{x}_i^T, t_i) = (15, 21, 35, 60)$ は、ある月の 15 日、気温 21 度、湿度 35%の時、ビールは 60 本売れたということである。

店長 「バイト君は大学で機械学習の授業取ってるんだって？これまでのデータはあるから毎日のビールの売れ行きを予測してみてくれないか？」

バイト君　「余裕っすよ．これまでのデータ$_1$ をもらえますか?多項式重回帰$_2$ で予測モデルを作成して, 過去のデータでまずどれだけ予測が当たるか$_3$ 評価してみます.」

店長 「結果はどうだい？」

バイト君　「すごいっすよ. 過去のどの日の予測も, ほとんど誤差なく$_4$ 売れ行きを当てることができています．機械学習すごいっす.」

(店長心の声)　商売はそんなに甘いものではない．誤差なく当たることなどあるのだろうか．しかしバイト君は機械学習を勉強したといっているし...

店長 「そうか．では明後日 24 日の天気予報では，気温 22 度，湿度 65%だそうだから，明後日のビールの売り上げをさっそく予測してもらおうか.」

バイト君 「多項式特徴量を使った回帰による予測だと, 24 日はバカ売れっす! 10 ケース仕入れても売り切れです!」

しかし 24 日は給料日前日なうえ，急激な冷え込みのため，ビールは全く売れなかった$_5$．店長は在庫を抱え，バイト君はクビになりそうである．

演習

1. 下線 1 の事例を機械学習では何と呼ぶか
2. 下線 2 において, 4 次の多項式特徴量を使った重回帰モデル $f(\boldsymbol{x})$ を式で示せ
3. 下線 3 で評価した量を何と呼ぶか. $i = 1, \ldots, N$ として、その評価量を式で表せ
4. 下線 4 の現象をなんと呼ぶか
5. 下線 5 の現象が起きた理由について、考えられることを全てあげよ

6. 予測性能を向上させるために考えられるモデリング上の工夫について以下の語句を使って考察しなさい. [テスト誤差, 交差検証].
7. 予測性能を向上させるために考えられる特徴量の設計上の工夫について考えられることを書きなさい

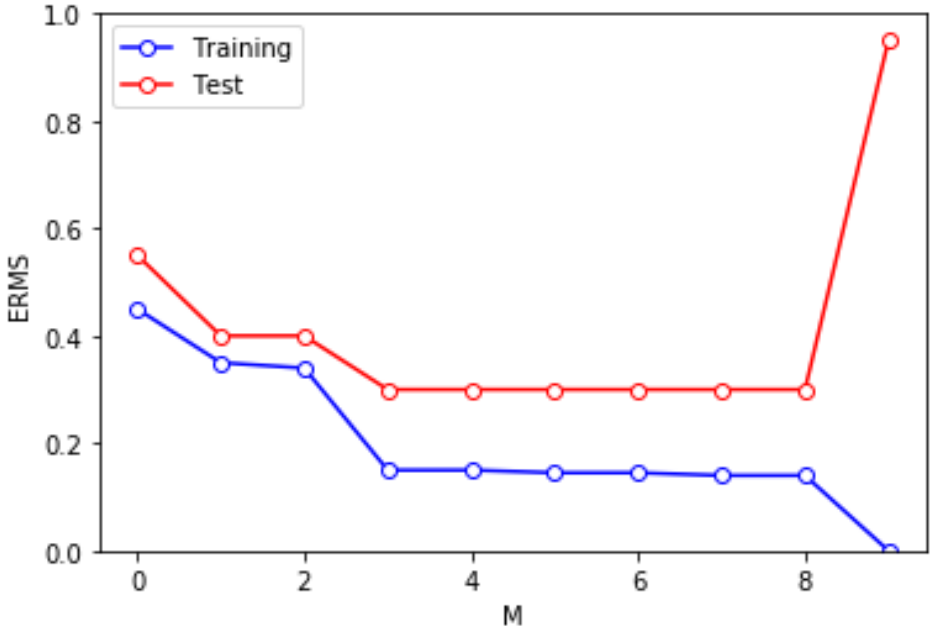
正則化

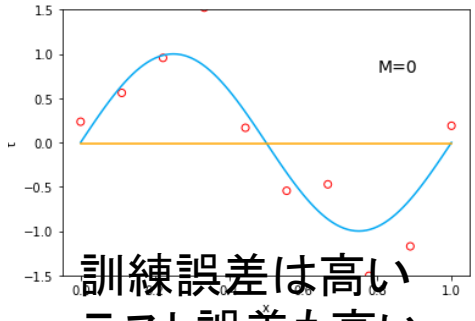
- 複雑度の高いモデル
  - 訓練誤差は低い
  - テスト誤差は高い

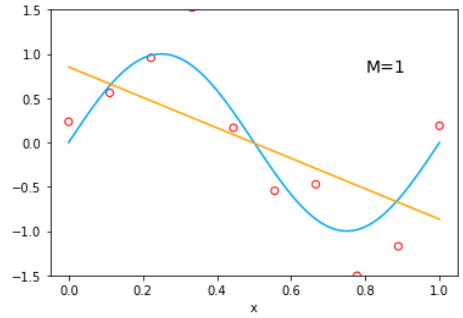
  } ⇒ 過学習
- 訓練誤差は直接最小化できるがテスト誤差は直接最小化できない
  - 学習時にテスト事例を使ったら、その事例はテスト誤差評価には使うことができない→なぜ？
- そのかわりに正則化(regularization)
  - 鋭すぎる刃を鈍らせて使う
  - モデルに対する知識で複雑さを制御
  - 複雑さを制御すればテスト誤差を制御できる(と期待)

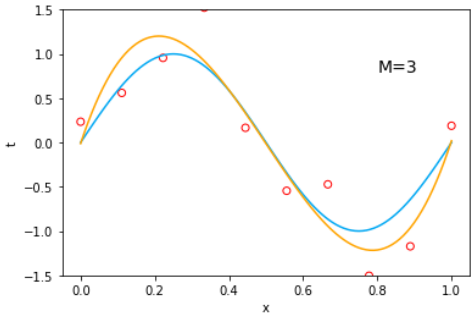
複雑なモデルとはどんなモデル？

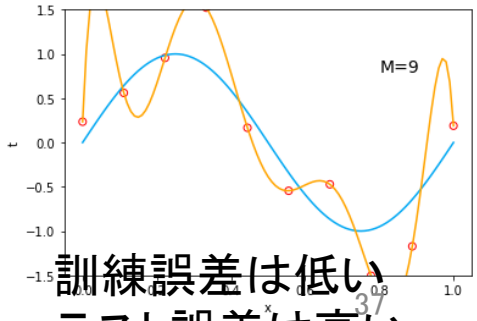
	M=0	M=1	M=3	M=9
w_0^*	0.13	0.64	-0.1	0.07
w_1^*		-1.04	10.9	356.27
w_2^*			-31.05	-812.08
w_3^*			20.53	7247.59
w_4^*				-33584.99
w_5^*				90058.44
w_6^*				-145162.27
w_7^*				138775.85
w_8^*				-72490.8
w_9^*				15932.65











訓練誤差は高い
テスト誤差も高い

訓練誤差は低い
テスト誤差は高い

罰則付き最適化

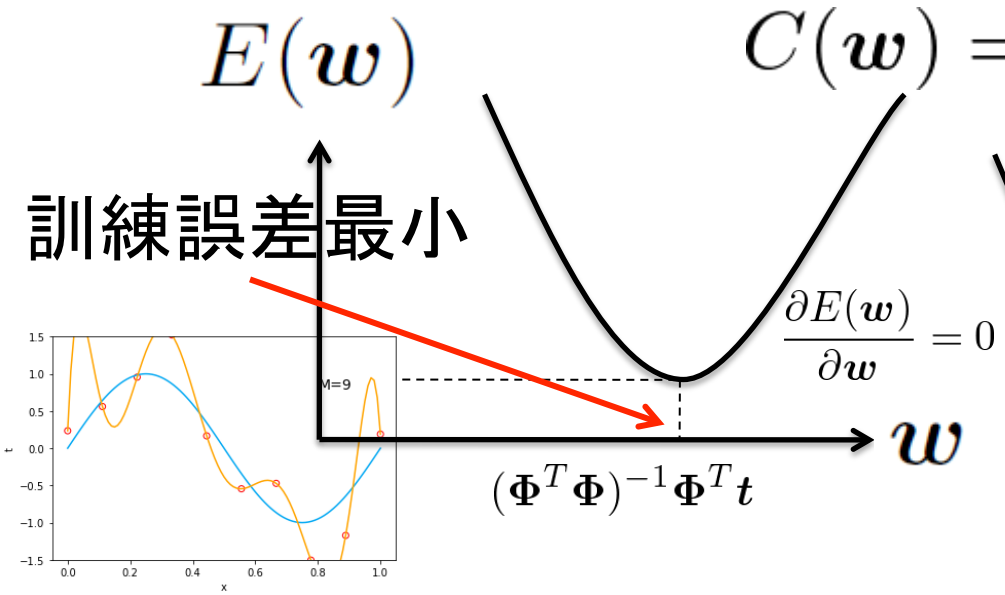
- $f(\boldsymbol{x}), g(\boldsymbol{x})$ の両方を最小化したい
  - 両者は矛盾する, e.g.
  - $f(\boldsymbol{x})$ 家賃(最小化)
  - $g(\boldsymbol{x})$ 駅からの遠さ(最小化)

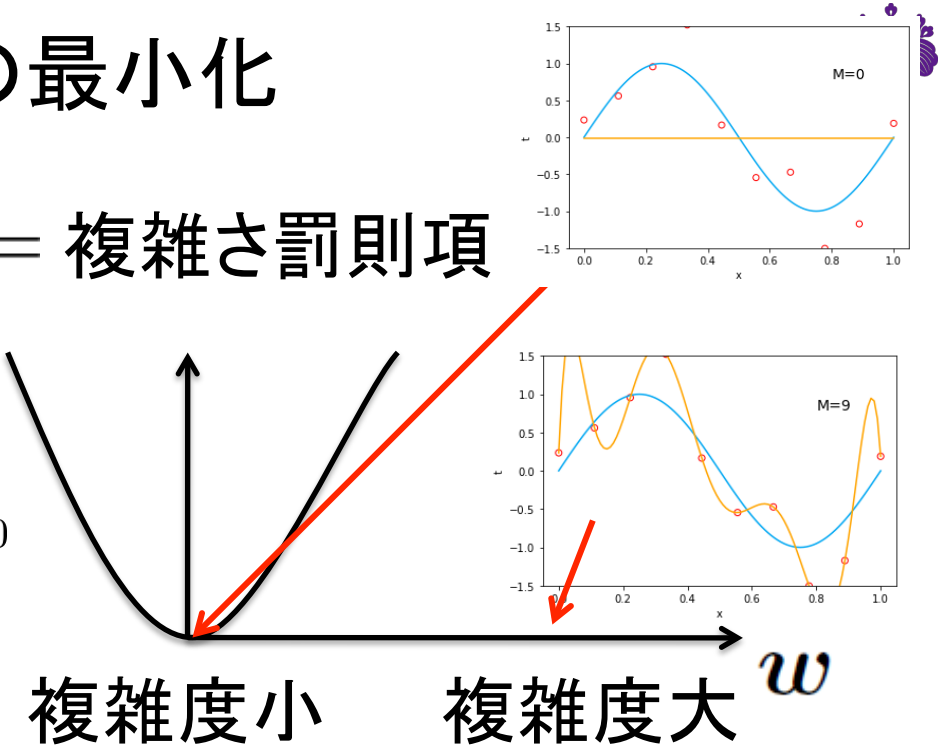
  } トレードオフ
- 罰則付き最適化
  - $\text{minimize } h(\boldsymbol{x}) = f(\boldsymbol{x}) + \lambda g(\boldsymbol{x})$

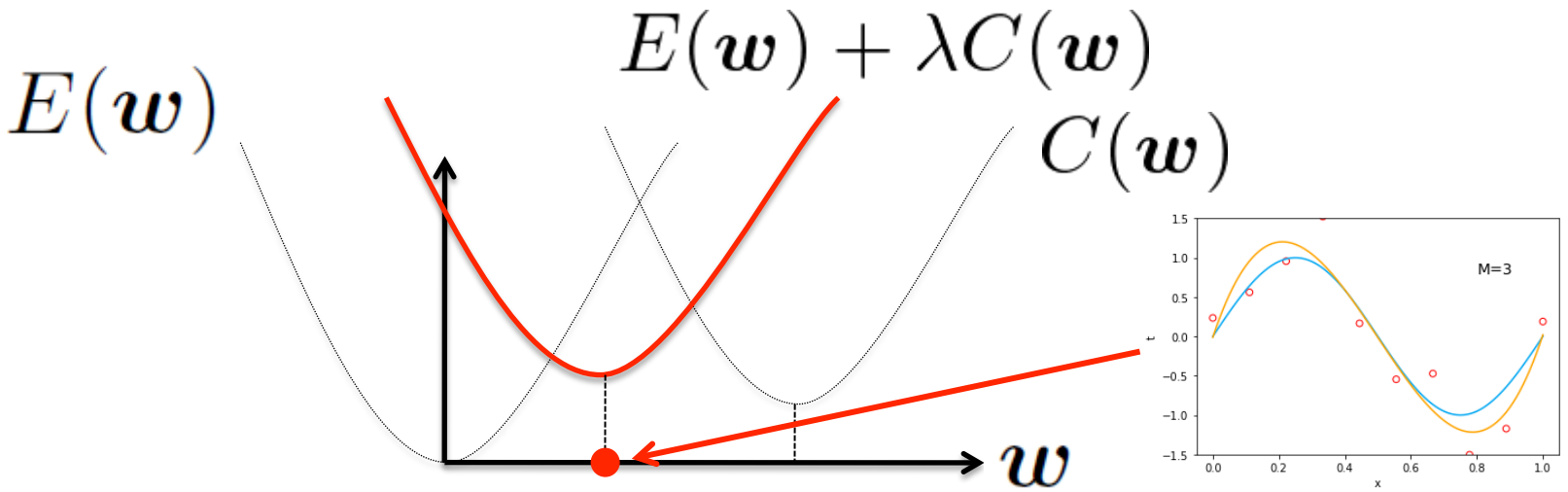
    λ: トレードオフパラメータ
    優先度を決める
- 回帰の場合は…
  - 訓練誤差を最小化
  - 複雑さを最小化

  } トレードオフ
    - 汎化誤差の低下を期待

複雑さ罰則項と訓練誤差の最小化







訓練誤差も複雑度もほどよく小さい?

ノルム

- ベクトル $\boldsymbol{x}^T = (x_1, x_2, \ldots, x_D)$
- p-ノルム

$$\|\boldsymbol{x}\|_p = \left(\sum_{i=1}^{D} |x_i|^p \right)^{1/p}$$

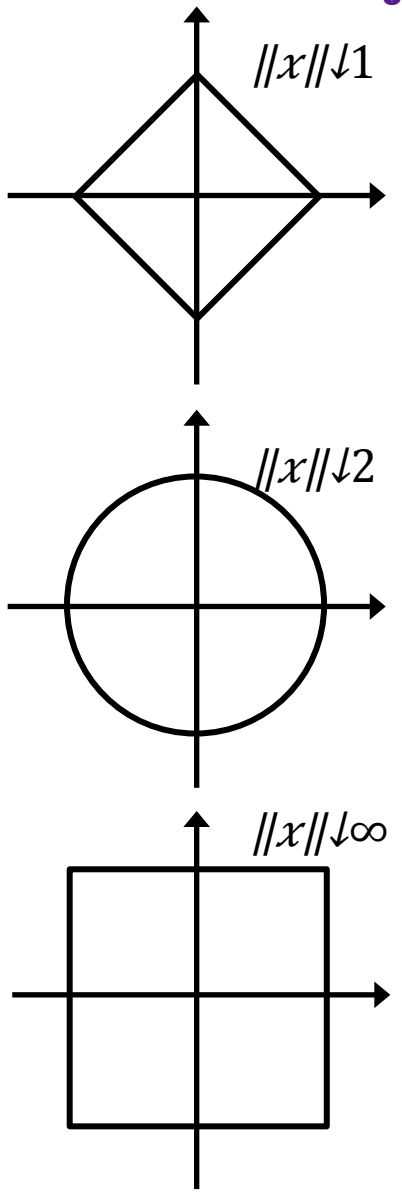
xのpノルムをこう書く

- ユークリッドノルム(2-ノルム, 距離)

$$\|\boldsymbol{x}\|_2 = \left(\sum_{i=1}^{D} |x_i|^2 \right)^{1/2}$$

- Maxノルム

$$\|\boldsymbol{x}\|_\infty = \max(|x_1|, |x_2|, \ldots, |x_D|)$$



各ノルムで定義される単位円

リッジ回帰=二乗誤差項＋L2正則化項

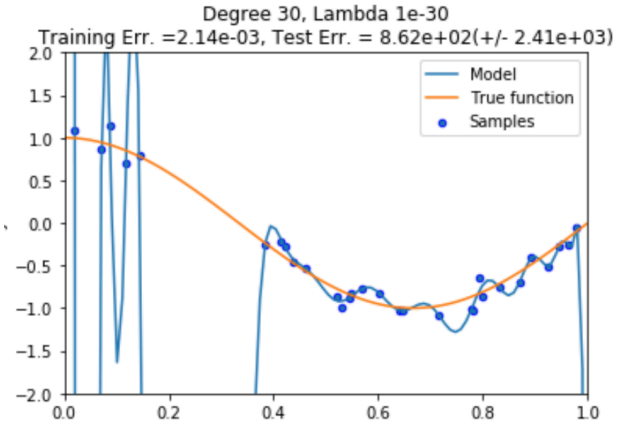
- これまでの誤差関数 $E(\boldsymbol{w}) = \sum_{i=1}^{N} (t_i - \boldsymbol{w}^T \boldsymbol{x}_i)^2$
- L2正則化項入りの誤差関数

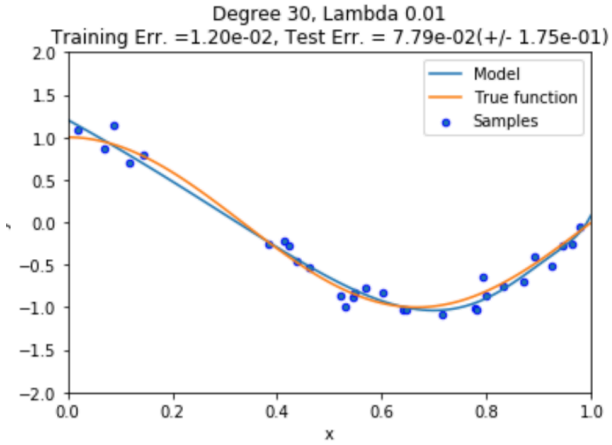
$$E(\boldsymbol{w}) = \sum_{i=1}^{N} (t_i - \boldsymbol{w}^T \boldsymbol{x}_i)^2 + \lambda \boldsymbol{w}^T \boldsymbol{w}$$

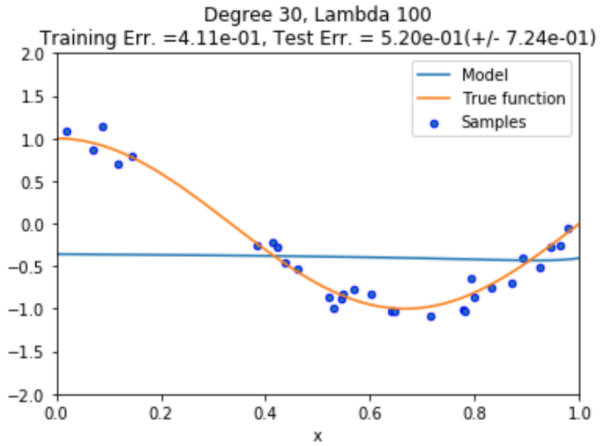
二乗誤差項　正則化パラメータ　L2正則化項

- L2正則化項
  - wの各要素が大きい値をとると大きくなる
    - 複雑さを抑制
  - 凸関数の和は凸関数→唯一の局所最適解
  - 微分可能→解析解が求まる

L2正則化の効果







正則化なし
λ=0

λ=0.01

正則化強め
λ=100

正則化パラメータ	訓練誤差	モデル複雑さ	汎化誤差
$\lambda = 0$	小さい	複雑になりやすい	大きい
$\lambda = \alpha$ 中間的	ほどほど	ほどほど	ほどほど(と期待)
$\lambda = \infty$	最小化されない	単純すぎ	結局大きい

Code: polynomialRidgeRegression.ipynb

問題　多項式単回帰の正則化

3.3　多項式単回帰の正則化

polynomialRidgeRegression.py を実行して以下の問いに答えなさい。

$y = \cos(1.5\pi x) + 0.1 * N(0,1)$ に従い、30 の訓練事例を生成した。ここで $N(0,1)$ は平均 0 分散 1 の正規分布から生成した正規乱数である。プログラムは、正則化係数を [1e-30,1e-20,1e-10,1e-5, 1e-3,1e-2,1e-1, 1,10,100] の範囲で変化させながら、これらの訓練事例を用いて、degree=30 の多項式回帰モデルによる学習を行っている。その出力は訓練誤差 TrainErr. およびテスト誤差 TestErr（とその標準偏差）を表示している。

1. $d = 30$ の多項式特徴量を用いた時に、もっとも小さい訓練誤差を与える正則化パラメータはいくらか

2. d=30 の多項式特徴量を用いた時に、もっとも小さいテスト誤差を与える正則化パラメータはいくらか

機械学習における最適化

- 解析的解法
  - 「目的関数が微分可能」かつ「勾配=0とするwがもとまる」場合に利用可能
  - (計算が終われば)正確な解が求まる
  - D*D行列の逆行列計算が必要(D＝次元数)
  - データが非常に大きい(次元数D, サンプル数N)場合、計算できない場合がある (メモリの制約)
- 近似解法 (最急降下法GD、確率的最急降下法SGD)
  - 勾配方向へのくり返し降下
  - 徐々に解を改善するため、費やした時間に応じた質の解が求まる
  - 逆行列計算が不要で、1ステップあたりの計算が軽い
  - データが大きい場合でもメモリの制約を受け難い (特にSGD)

リッジ回帰の解析的解法

- 原則ただの回帰といっしょ(導出は省略)

$$\frac{\partial E(\boldsymbol{w})}{\partial \boldsymbol{w}} = 0 \Rightarrow \boldsymbol{w} = (\boldsymbol{\Phi}^T\boldsymbol{\Phi} + \lambda \boldsymbol{I})^{-1}\boldsymbol{\Phi}\boldsymbol{t}$$

cf.　ただの回帰 $\boldsymbol{w} = (\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}\boldsymbol{t}$

- 正則化パラメータ λ のチューニングは交差検定が必要
- 逆行列計算を安定化する効果もある
- 近似解法は次回…

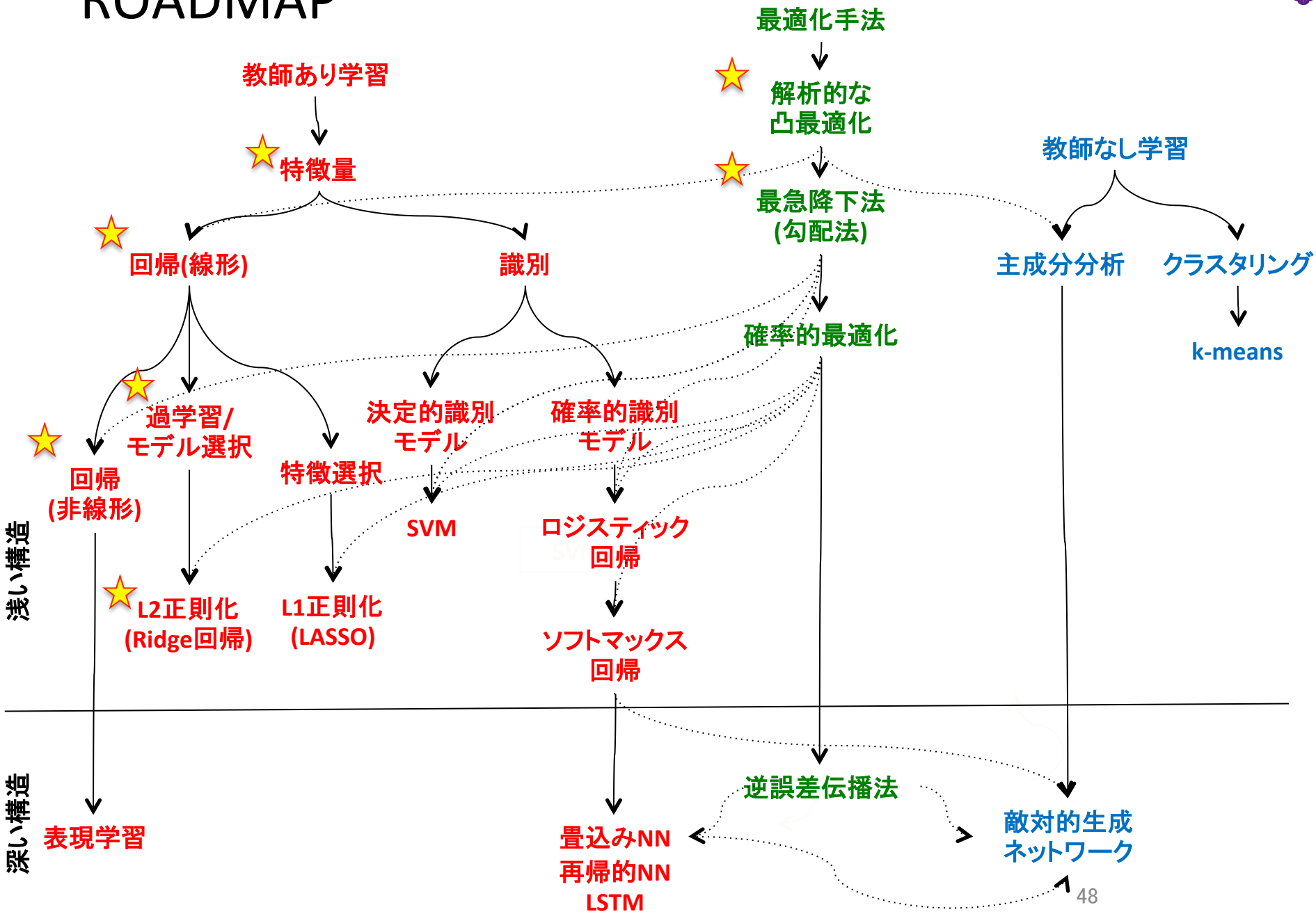
複雑さの制御まとめ

- 非線形性の導入→多項式特徴量（実際は計算法も含めて線形回帰と何も変わらない）
- 複雑すぎるモデルは低い訓練誤差を持つが高い汎化誤差を持つ(過学習)ので意味がない
- ちょうどよい複雑さのモデルを選ぶには？
  - 様々な複雑さを持つモデルを用意し、k-fold交差検定でそれぞれのテスト誤差を推定し，小さいテスト誤差を持つモデルを選べばいい
  - 正則化項を誤差関数に導入し、正則化パラメータを変化させながら、テスト誤差を推定し，小さいテスト誤差を持つ正則化パラメータによる学習結果を選べばいい

モデル選択Q&A

- 次数と正則化を組み合わせた最適化はあるか？
  - k-fold CVを使えばできますね
- k-fold CVのkはどうやって決める?単純に大きければいいわけではない？
  - 基本的には大きい方が汎化誤差推定の精度はよくなります．ただし，時間がかかります．一番精度が高いのは，データ数NについてN-fold作るleave one out CV (LOOCV)という方法ですが，Nが大きい場合はかなり時間がかかります
- リッジ回帰のλはどう決める？
  - k-fold CVで決めます．
- 二乗誤差でモデルの良し悪しを判断した場合，次数が多ければ理想のモデルに近づくのか，実際にすべてのパターンでそう言い切れるのか？
  - 難しい問題です．まず，k-fold CVなどでテスト誤差を低く抑えるようなパラメータ選択やモデル選択を行うことが必須です．そのうえで，テスト誤差が小さければ，真のモデルf(x)に本当に近いのか？　人工的に生成したデータでない限り、f(x)は結局だれにもわからないので、これが真のモデルに近いかどうかは、神のみぞ知ることです。ただし、一般的には、（ある条件において）学習に利用できる訓練サンプル数が多ければ多いほど、その汎化誤差を小さくできることが知られています

ROADMAP


最適化手法
教師あり学習
解析的な
凸最適化
特徴量
教師なし学習
最急降下法
(勾配法)
回帰(線形)
識別
主成分分析
クラスタリング
確率的最適化
k-means
過学習/
モデル選択
決定的識別
モデル
確率的識別
モデル
回帰
(非線形)
特徴選択
SVM
ロジスティック
回帰
浅い構造
L2正則化
(Ridge回帰)
L1正則化
(LASSO)
ソフトマックス
回帰
逆誤差伝播法
深い構造
表現学習
畳込みNN
再帰的NN
LSTM
敵対的生成
ネットワーク