# Hồi quy tuyến tính (Linear Regression)

Quách Đình Hoàng

2022/09/16

# Nội dung

# Bài toán hồi quy (regression problem)

# Bài toán hồi quy (regression problem)

- Ta nghĩ rằng $y \in R$ và $x \in R^d$ được xấp xỉ bởi:

$$y \approx f(x)$$

  - $x$ được gọi là biến độc lập (independence variable) hay vector đặc trưng (feature vector)
  - $y$ được gọi là biến phụ thuộc (dependence variable) hoặc output hoặc response
  - Thông thường, $y$ là biến mà ta muốn dự đoán.
- Ta không biết mối quan hệ thật sự giữa $y$ và $x$, hàm $f(.)$ chỉ là một sự xấp xỉ.

# Explanatory vs. predictive modeling with regression

| | Explanatory Modeling (Statistical approach) | Predictive Modeling (machine learning approach) |
|---|---|---|
| **General goal** | Giải thích mối quan hệ giữa input $x$ và output $y$. | Dự đoán output $y$ từ input $x$. |
| **Modeling** | Tìm mô hình sinh ra dữ liệu (phân bố $p(x, y)$). | Tìm hàm $f$ (blackbox) để dự đoán $y$ từ $x$. |
| **Model validaion** | Dùng cả dataset để thực hiện "goodness-of-fit" test: $R^2$, residual analysis, p-values, ... | Chia dataset thành train/test set. Train mô hình trên train set và đánh giá mô hình trên test set |

Tham khảo thêm:

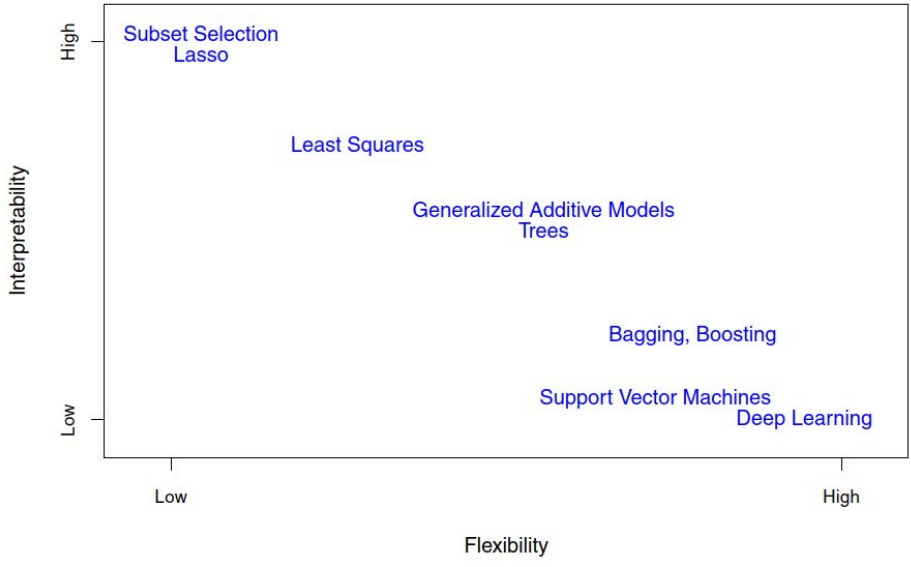

1. Leo Breiman, *Statistical Modeling: The Two Cultures, Statistical Science*, Vol. 16, No. 3, 199-231, 2001.
2. Galit Shmueli, *To Explain or to Predict?*, Statistical Science, Vol. 25, No. 3,

# Prediction vs. Inference

Ta quan tâm đến hai khía cạnh của một mô hình học máy:

- Prediction: khả năng dự đoán chính xác của mô hình
  - Ví dụ: Một công ty thực hiện một chiến dịch tiếp thị, mục tiêu là xác định các khách hàng có phản ứng tích cực với chiến dịch.
- Inference: khả năng mô tả (diễn giải) mối quan hệ giữa $X$ và $Y$ của mô hình
  - Ví dụ: Một công ty thực hiện một các chiến dịch tiếp thị (qua TV, radio, và qua newspaper), mục tiêu là trả lời các câu hỏi sau:
    - Loại quảng cáo nào có mối quan hệ với doanh thu?
    - Loại quảng cáo nào ảnh hưởng lớn nhất đến doanh thu?
    - Nếu tăng tiền (một lượng cho trước) cho một loại hình quảng cáo (như TV), doanh thu sẽ tăng bao nhiêu?
- Trong nhiều trường hợp, ta quan tâm cả prediction và inference.
- Tùy vào mục tiêu là prediction, inference, hay cả hai mà ta chọn mô hình phù hợp.

# Prediction Accuracy and Model Interpretability



# Regression algorithms

- Linear regression
- Linear model selection
  - Best subset selection
  - Forward/Backward stepwise selection
  - Ridge/Lasso/ElasticNet regression
- Linear regression extensions
  - Splines and smoothing splines
  - Local regression
  - Generalized additive models
- Non-linear regression
  - Polynomial regression
  - k-nn regression
  - Regression tree
    - Bagging for regression
    - Random forest for regression
- Support vector regression
- Neural network regression

Tham khảo thêm:


1. Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani, An introduction to statistical learning, Second edition, Springer, 2021.
2. AJ Smola and B Schölkopf, A tutorial on support vector regression, Statistics and computing, 14, 199-222, 2004.

# Hồi quy tuyến tính (linear regression)

# Hồi quy tuyến tính (linear regression)

- Mô hình $f$ là hàm tuyến tính theo $x$ là dạng đơn giản và phổ biến nhất
- Mô hình hồi quy tuyến tính có dạng:

$$\hat{y} = f(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_d x_d = \theta^T x$$

  - $\theta^T = (\theta_0, \theta_1, \cdots, \theta_d) \in R^{d+1}$ được gọi là tham số của mô hình
  - $x = (1, x_1, \cdots, x_d)^T \in R^{d+1}$ là đầu vào của mô hình.
  - Ta quy ước $x$ là vector dạng cột, $x^T$ là vector dạng dòng.
  - Khi cần nhấn mạnh, ta viết $f_\theta(x)$ để mô tả sự phụ thuộc của $f$ vào $\theta$.
  - $\theta_0$ là dự đoán của mô hình khi tất cả các đặc trưng bằng 0.

# Diễn giải các hệ số hồi quy

$$\widehat{y_i} = f(x_i) = \theta_0 + \theta_1 x_i + \cdots \theta_d x_d$$

- $\theta_i (i \neq 0)$ là mức độ $\widehat{y} = f(x)$ tăng khi $x_i$ lên một đơn vị
- $\theta_i = 0$ ngụ ý rằng $\widehat{y} = f(x)$ không phụ thuộc vào $x_i$
- $\theta$ nhỏ ngụ ý rằng mô hình không nhạy cảm (insensitive) với sự thay đổi của $x$

$$|f(x) - f(x')| = |\theta^T x - \theta^T x'| = |\theta^T (x - x')| \leq ||\theta|| ||x - x'||$$

# Hàm mất mát (loss function)

- Hàm mất mát $l : R \times R \to R$ xác định mức độ $\hat{y}$ xấp xỉ $y$
  - $l(\hat{y}, y) \geq 0, \forall \hat{y}, y$
  - $l(\hat{y}, y)$ nhỏ chứng tỏ $\hat{y}$ xấp xỉ $y$ tốt
- Hai hàm mất mát (loss function) phổ biến là:
  - Quadratic/square loss ($L_2$): $l(\hat{y}, y) = (\hat{y} - y)^2$
  - Absolute loss ($L_1$): $l(\hat{y}, y) = |\hat{y} - y|$

# Sai số thực nghiệm (empirical risk)

- Sai số thực nghiệm (empirical risk) là mất mát trung bình (average loss) trên cả tập dữ liệu $\{(x_i, y_i)\}_{i=1..N}$

$$L = \frac{1}{n}\sum_{i=1}^{n} l(\hat{y}_i, y_i) = \frac{1}{n}\sum_{i=1}^{n} l(f(x_i), y_i)$$

  - Nếu $L$ nhỏ, mô hình dự đoán tốt trên dữ liệu cho trước
  - Khi mô hình được tham số hóa bởi $\theta$, ta viết

$$L(\theta) = \frac{1}{n}\sum_{i=1}^{n} l(f_\theta(x_i), y_i)$$

    để thể hiện sự phụ thuộc của mô hình vào $\theta$

# Trung bình sai số bình phương (mean square error)

- Khi hàm mất mát là $L_2$: $l(\hat{y}, y) = (\hat{y} - y)^2$ thì sai số thực nghiệm là trung bình sai số bình phương (mean square error - MSE)

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

  - Tuy nghiên, ta thường dùng root-mean-square error, $RMSE = \sqrt{MSE}$, vì nó có cùng đơn vị với $y_i$

# Trung bình sai số tuyệt đối (mean absolute error)

- Khi hàm mất mát là $L_1$: $l(\hat{y}, y) = |\hat{y} - y|$ thì sai số thực nghiệm là trung bình sai số tuyệt đối (mean absolute error - MAE)

$$L = MAE = \frac{1}{n}\sum_{i=1}^{n} |f(x_i) - y_i|$$

  - MAE có dùng đơn vị với $y_i$

# Cực tiểu hóa sai số thực nghiệm (empirical risk minimization)

- Cực tiểu hóa sai số thực nghiệm (empirical risk minimization - ERM) là một phương pháp chung để chọn các tham số $\theta$ cho mô hình $f_\theta(x)$
- ERM chọn $\theta$ sao cho sai số thực nghiệm $L(\theta)$ đạt cực tiểu
- Thông thường, không có lời giải dạng giải tích cho bài toán tìm cực trị này. Do đó, ta thường phải dùng các phương pháp tối ưu số học (numerical optimization) để tìm $\theta$ sao cho $L(\theta)$ đạt cực tiểu.

# Least square linear regression

- Mô hình hồi quy tuyến tính

$$\hat{y} = f_\theta(x) = \theta^T x$$

  - $\theta \in R^{d+1}$ là tham số của mô hình
  - $x \in R^{d+1}$ là đầu vào của mô hình
- Ta dùng hàm mất mát $l(\hat{y}, y) = (\hat{y} - y)^2$
- Sai số thực nghiệm là mean square error - MSE

$$L(\theta) = \frac{1}{n} \sum_{i=1}^{n} (\theta^T x_i - y_i)^2$$

- Ta ước lượng $\theta$ dùng phương pháp cực tiểu hóa sai số thực nghiệm (empirical risk minimization - ERM)
  - Chọn $\theta$ sao cho $L(\theta)$ là nhỏ nhất

# Least square linear regression

- MSE được viết lại dạng ma trận

$$L(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\theta^T x_i - y_i)^2 = \frac{1}{n}||X\theta - y||^2$$

  - $\theta = (\theta_0, \theta_1, \cdots, \theta_d)^T \in R^{d+1}$
  - $X \in R^{n\times(d+1)}, y \in R^n$

$$X = \begin{bmatrix}(x_1)^T\\(x_2)^T\\\vdots\\(x_n)^T\end{bmatrix} = \begin{bmatrix}1 & x_{11} & x_{12} & \cdots & x_{1d}\\1 & x_{21} & x_{22} & \cdots & x_{2d}\\\vdots & & & & \\1 & x_{n1} & x_{n2} & \cdots & x_{nd}\end{bmatrix} \quad Y = \begin{bmatrix}y_1\\y_2\\\vdots\\y_n\end{bmatrix} \qquad (1)$$

- Ta cần chọn $\theta$ để $||X\theta - y||^2$ là nhỏ nhất.

# Least square linear regression - analytical solution

$$L(\theta) = ||X\theta - y||^2 = \sum_{i=1}^{n}(\sum_{j=0}^{d} x_{ij}\theta_j - y_i)^2$$

- Nghiệm tối ưu $\hat{\theta}$ thỏa

$$\frac{\delta L}{\delta \theta_j}(\hat{\theta}) = \nabla L(\hat{\theta})_j = 0, j \in \{1, \cdots, d\}$$

- Lấy đạo hàm riêng theo $\theta_j$: $\nabla L(\theta)_j = (2X^T(X\theta - Y))_j$
- Viết dạng ma trận: $\nabla L(\hat{\theta}) = 2X^T(X\hat{\theta} - Y) = 0$
- $\hat{\theta}$ cần thỏa phương trình: $(X^T X)\hat{\theta} = X^T Y$
- Do đó: $\hat{\theta} = (X^T X)^{-1} X^T Y$ (nếu $X^T X$ khả nghịch)

# Least square linear regression

- Bài toán: Ta cần chọn $\theta$ để $||X\theta - y||^2$ là nhỏ nhất.
- Nếu các cột (hoặc dòng) của $X$ là độc lập tuyến tính (linearly independent) thì $X^T X$ là khả nghịch và bài toán có lời giải duy nhất

$$\theta^* = (X^T X)^{-1} X^T Y = X^\dagger Y$$

- Nếu $X$ các cột và dòng của $X$ đều phụ thuộc tuyến tính (linearly dependent) thì $X^T X$ không khả nghịch, ma trận nghịch đảo giả (pseudo-inverse) của $X^T X$ với công thức $X^T (X X^T)^{-1}$ có thể được sử dụng

# Least square linear regression - gradient descent

$t = 0$

Khởi tạo $\theta_j^{(t)}$ ngẫu nhiên

repeat

- $partial[j] = 0$ for all $0 \leq j \leq d$
- foreach data point $i = 1, 2, ..., n$
  - foreach parameter $j = 0, 1, ..., d$
    - $partial[j]$ += $\left(-x_i(y_i - x_i^T \theta_j^{(t)})\right)$
- $\theta^{(t+1)} = \theta^{(t)} - \eta \cdot partial[j]$
- $t \leftarrow t + 1$

until $||\theta^{(t)} - \theta^{(t-1)}|| \leq \delta$

# Least square linear regression - stochastic gradient descent

$t = 0$

Khởi tạo $\theta^{(1)}$ ngẫu nhiên

repeat

- foreach $i = 1, 2, \ldots, n$ (thứ tự ngẫu nhiên)
  - $partial[i] = -x_i(y_i - x_i^T \theta_i^{(t)})$
  - $\theta^{(t+1)} = \theta^{(t)} - \eta \cdot partial[i]$
- $t \leftarrow t + 1$

until $||\theta^{(t)} - \theta^{(t-1)}|| \leq \delta$

# Least square linear regression - mini-batch gradient descent

$t = 0$

Khởi tạo $\theta_j^{(t)}$ ngẫu nhiên

repeat

- Chia dataset thành $k$ mini_batch với kích cỡ $l$ ngẫu nhiên $(k * l = n)$
- foreach random $mini\_batch$
  - $partial[j] = 0$ for all $0 \leq j \leq d$
  - foreach parameter $j = 0, 1, \ldots, d$
    - $partial[j] \mathrel{+}= \sum_{(x_i, y_i) \in mini\_batch} \left(-x_i(y_i - x_i^T \theta_j^{(t)})\right)$
  - $\theta^{(t+1)} = \theta^{(t)} - \eta \cdot partial[j]$
  - $t \leftarrow t + 1$

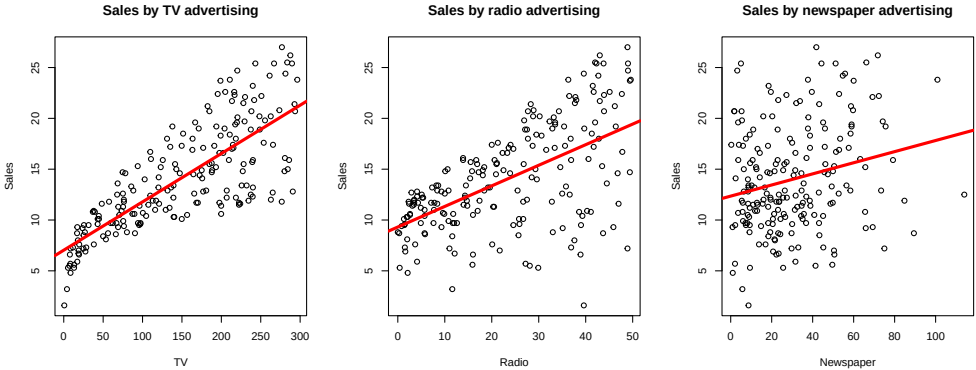
until $||\theta^{(t)} - \theta^{(t-1)}|| \leq \delta$

# Ví dụ: Advertising data

- Đầu vào $u$ là các biến $TV, radio, newspaper$ mô tả số tiền quảng các cho các loại hình này.
- Đầu ra $v$ là biến $sales$ mô tả doanh thu của công ty.
- Ta muốn trả lời một số câu hỏi sau:
  - Liệu có mối quan hệ giữa tiền chi cho quảng cáo với doanh số?
  - Mối quan hệ đó có mạnh không?
  - Mối quan hệ đó có tuyến tính không?
  - Loại quảng cáo nào đóng góp nhiều hơn vào doanh thu?
  - Ta có thể dự đoán được doanh số tương lai dựa vào số tiền chi cho quảng cáo không?

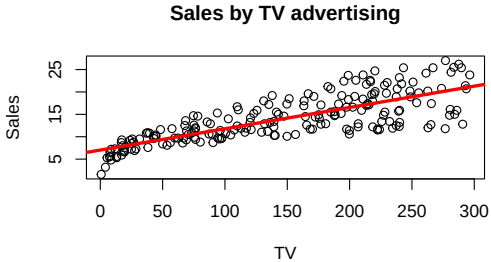
# Mô hình hồi quy tuyến tính cho advertising data

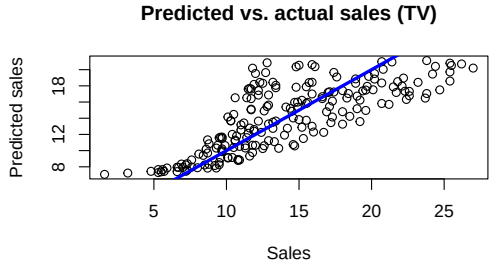
- Hồi quy tuyến tính cho mỗi biến trên advertising data




- Biến dự đoán tốt nhất là $TV$, với $MSE$ là 10.51
  - $radio$ có $MSE$ là 18.09
  - $newspaper$ có $MSE$ là 25.67

# Dự đoán doanh số với TV advertising




- Hình bên trái là mô hình hồi quy tuyến tính với biến $TV$

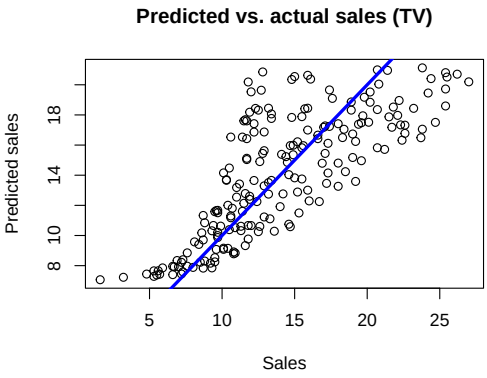
$$sales \approx 7.03 + 0.05 \times TV$$

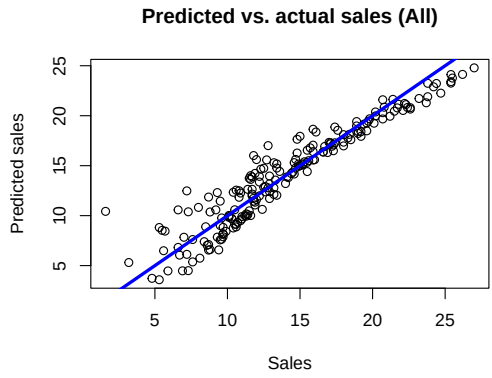
- Hình bên phải là doanh số dự đoán và doanh số thật sự
  - Lý tưởng là mọi điểm đều nằm trên đường màu xanh

# Dự đoán doanh số với TV advertising

```
##
## Call:
## lm(formula = sales ~ TV, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.3860 -1.9545 -0.1913  2.0671  7.2124
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 7.032594   0.457843   15.36   <2e-16 ***
## TV          0.047537   0.002691   17.67   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.259 on 198 degrees of freedom
## Multiple R-squared:  0.6119, Adjusted R-squared:  0.6099
```

# Dự đoán doanh số với tất cả các biến




- Hình bên trái là mô hình chỉ sử dụng biến $TV$, $MSE$ là 10.51
- Hình bên phải mô hình chỉ sử dụng tất cả các biến, $MSE$ là 2.78

$$sales \approx 2.94 + 0.05 \times TV + 0.19 \times radio - 0.001 \times newpaper$$

# Dự đoán doanh số với tất cả các biến

```
##
## Call:
## lm(formula = sales ~ TV + radio + newspaper, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.8277 -0.8908  0.2418  1.1893  2.8292
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.938889   0.311908   9.422   <2e-16 ***
## TV           0.045765   0.001395  32.809   <2e-16 ***
## radio        0.188530   0.008611  21.893   <2e-16 ***
## newspaper   -0.001037   0.005871  -0.177     0.86
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

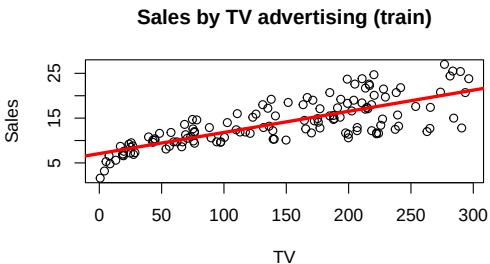
# Sự đánh giá (validation)

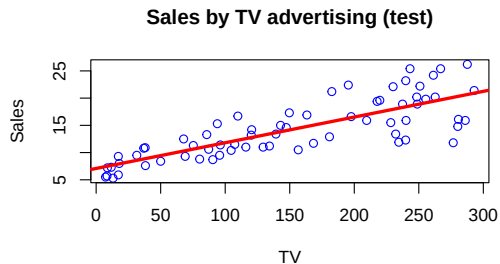
# Sự tổng quát (generalization)

- Sự tổng quát (generalization) là khả năng một mô hình dự đoán tốt trên dữ liệu mới
  - Dự đoán tốt trên dữ liệu huấn luyện (training data) không phải là mục tiêu cuối cùng
- Ta xây dựng mô hình dựa trên dữ liệu huấn luyện (training data) hay in-sample data
- Ta mong muốn mô hình cũng dự đoán tốt trên dữ liệu mới (test data) hay out-of-sample data
- Nếu nó không dự đoán tốt trên dữ liệu mới, ta nói mô hình không tổng quát (thất bại trong việc tổng quát hóa)

# Ví dụ trên advertising data

- Ta huấn luyện (train) mô hình dùng $2/3$ tập dữ liệu để dự đoán trên $1/3$ còn lại.




- MSE trên tập train là 10.65, MSE trên tập test là 10.64
- Ta có thể kết luận mô hình mang tính tổng quát
  - Sự khác biệt của MSE trên tập train và tập test không lớn lắm

# Out-of-sample validation

- Ta dùng validation/test set để kiểm tra khả năng dự đoán của mô hình trên dữ liệu mới (chưa thấy) như thế nào.
  - Đây được gọi nguyên lý out-of-sample validation
- Hai phương pháp đánh giá phổ biến dựa theo nguyên lý out-of-sample validation:
  - Holdout method
  - Cross-validation

# Holdout validation

Holdout validation là dạng đơn giản nhất của out-of-sample validation.

- Ý tưởng: sử dụng một phần của tập dữ liệu làm dữ liệu chưa thấy và giả định dữ liệu tương lai sẽ tương như vậy.
  - Chia tập dữ liệu ta có thành hai tập, train và test một cách ngẫu nhiên
  - Sử dụng tập train để xây dựng mô hình
  - Sử dụng tập test để đánh giá mô hình
- Đây là một cách để ta mô phỏng khả năng dự đoán của mô hình trên dữ liệu chưa thấy.
  - Thông thường, ta chỉ có một tập dữ liệu, dữ liệu mới (chưa thấy) thường khó thu thập.

# Holdout validation

- Sai số khi dự đoán trên tập test (test error) là điều ta quan tâm
  - Sai số khi dự đoán trên tập train (train error) không quan trọng
- Ta chia dữ liệu thành 2 tập train và test một cách ngẫu nhiên (randomly)
  - Tỷ lệ train/test thường là 80/20 hay 90/10
  - Khi dữ liệu nhiều ta có chia theo tỷ lệ 50/50 hay 60/40
- Test error thường lớn hơn train error một tí
- Nếu test error lớn hơn nhiều so với train error, ta nói mô hình bị quá khớp (overfit).
- Random sampling là một biến thể của holdout
  - Lặp lại holdout $k$ lần
  - Dùng average test error qua $k$ lần để đánh giá mô hình

# Holdout validation

- Kết quả train/test error có các trường hợp sau

| test/train | small train error | large train error |
|---|---|---|
| small test error | generalizes (performs well) | lucky (or fraud) |
| large test error | fails to generalize (overfit) | generalizes (underfit) |

# Ví dụ trên advertising data

| features | train error | test error |
|---|---|---|
| TV | 10.74 | 10.06 |
| radio | 17.01 | 20.29 |
| newspaper | 24.54 | 28.04 |
| TV + radio | 2.68 | 2.98 |
| TV + newspaper | 2.68 | 9.17 |
| radio + newspaper | 2.68 | 20.77 |
| all | 2.68 | 3.07 |

- Kết quả cho thấy rằng mô hình chỉ với hai thuộc tính $TV$ và $radio$ cho kết quả dự đoán tốt tương tự với khi dùng tất cả các thuộc tính ($all$).

# Quá khớp (overfitting)

- Ta vừa thử nghiệm nhiều mô hình
- Ta có thể chọn mô hình khớp (fit) với dữ liệu huấn luyện (traning data) nhất.
- Nhưng việc này có thể dẫn đến mô hình dữ đoán không tốt trên dữ liệu kiểm thử (test data)
- Điều này được gọi là quá khớp (overfitting)

# Ví dụ - polynomial fit

- Giả sử dữ liệu thô là $u \in R$
- Ta sử dụng các đặc trưng là đa thức theo $u$ (polynomial features)

$$x = \phi(u) = (1, u, u^2, \cdots, u^k)^T$$

và mô hình hồi quy tuyến tính $f(x) = \theta^T x$

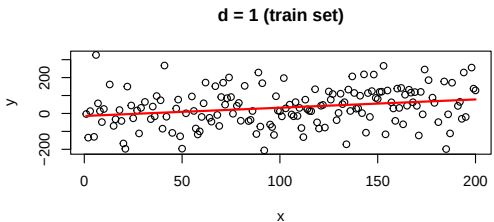
- $f(x)$ là đa thức bậc $d$ theo $u$

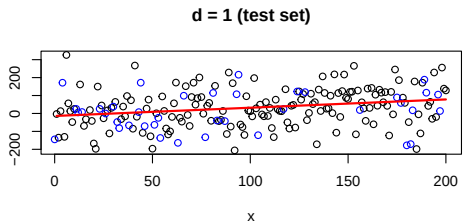
$$\hat{y} = f(x) = \theta^T x = \theta_0 + \theta_1 u + \theta_2 u^2 + \cdots \theta_d u^d$$

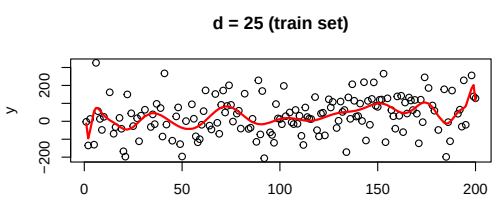
- Ta chọn $\theta$ dùng ERM với lost function là $l(\hat{y}, y) = (\hat{y} - y)^2$

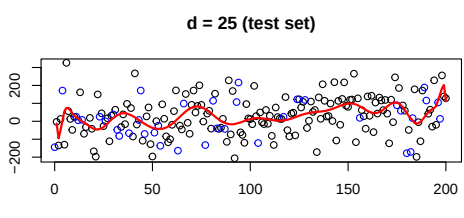
# Ví dụ - polynomial fit

- 200 điểm dữ liệu được chia làm 2 tập train/test với tỷ lệ 80/20



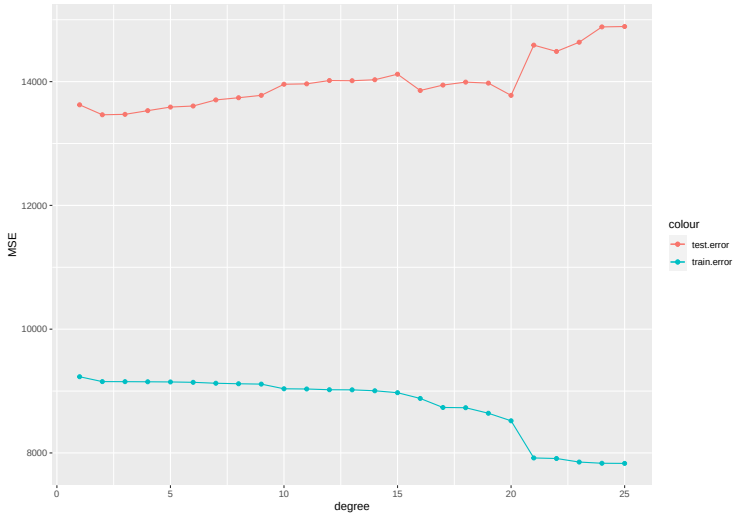



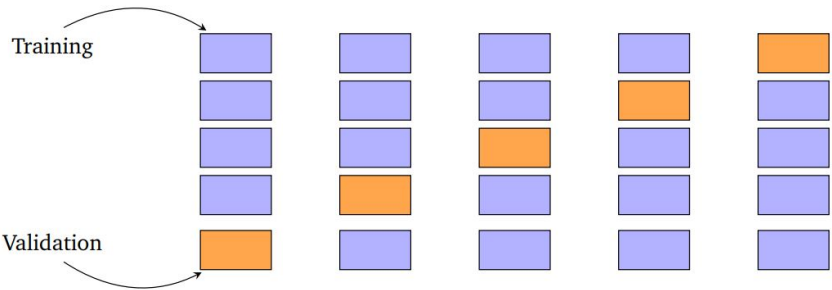
# Ví dụ - polynomial fit

- 200 điểm dữ liệu được chia làm 2 tập train/test với tỷ lệ 80/20



# Cross validation

- Đây là một phương pháp phổ biến để đánh giá mô hình
  - Chia tập dữ liệu thành $k$ phần (fold) xấp xỉ nhau
  - Với mỗi phần $i$, xây dựng mô hình trên các phần khác ngoại trừ phần $i$
  - Đánh giá mô hình trên phần $i$
  - Dùng everage test error trên các tất cả các phần $i$ để đánh giá mô hình.



# Cross validation

Có 2 trường hợp đặt biệt của cross validation

- Leave-one-out cross validation
  - Chia dữ liệu thành $n$ phần (fold), với $n$ là số đối tượng trong tập dữ liệu
  - Phương pháp này hữu ích đối với các tập dữ liệu nhỏ
- Stratified cross-validation
  - Các phần (fold) được phân tầng (stratified) để phân bố lớp trong mỗi phần xấp xỉ với phân bố lớp của tập dữ liệu
  - Phương pháp này hữu ích đối với các tập dữ liệu có phân bố lớp không đều nhau

# Lựa chọn mô hình tuyến tính (linear model selection)

# Best subset selection

- Gọi $M_0$ là null model, không chứa biến nào.
  - Mô hình này dùng sample mean để dự đoán cho mỗi data point
- for $k = 1, 2, \ldots, d$
  - Fit tất cả $\begin{pmatrix} d \\ k \end{pmatrix}$ mô hình chứa $k$ biến
  - Chọn mô hình tốt nhất (có $L(\theta)$ nhỏ nhất) trong $\begin{pmatrix} d \\ k \end{pmatrix}$ mô hình, và gọi nó là $M_k$.
- Chọn mô hình tốt nhất (có validation error nhỏ nhất) từ $M_0, \ldots M_d$ dùng cross validation.

# Best Subset Selection (cont)

- Best subset selection khó thể áp dụng khi số chiều $d$ lớn.
- Best subset selection có thể dẫn đến overfiting và high variance khi $d$ lớn.
  - Không gian tìm kiếm quá lớn $\rightarrow$ nhiều khả năng tìm được mô hình có performance tốt trên traning set nhưng tệ trên test set.
- Do vậy, các phương pháp stepwise selection nhằm giới hạn không gian tìm kiếm, được dùng thay cho best subset selection.

# Forward stepwise selection

- Gọi $M_0$ là null model, không chứa biến nào.
  - Mô hình này dùng sample mean để dự đoán cho mỗi data point
- for $k = 0, 2, \dots, d-1$
  - Xét tất cả $d-k$ mô hình được tạo ra bằng cách thêm một biến vào $M_k$
  - Chọn mô hình tốt nhất (có $L(\theta)$ nhỏ nhất) trong số $d-k$ mô hình, gọi nó là $M_{k+1}$.
- Chọn mô hình tốt nhất (có validation error nhỏ nhất) từ $M_0, \dots M_d$ dùng cross validation.

# Forward stepwise selection (cont)

- Forward stepwise selection có không gian tìm kiếm nhỏ hơn nhiều so với best subset selection}
  - Số trường hợp phải xét là $1 + d(d+1)/2 = O(d^2)$ so với $O(2^d)$
- Forward stepwise selection là phương pháp greedy nên không đảm bảo tìm được mô hình tốt nhất như best subset selection

# Backward stepwise selection

- Gọi $M_d$ là full model, chứa tất cả các biến.
- for $k = d, d-1, \dots, 1$
  - Xét tất cả $k$ mô hình được tạo ra bằng cách bỏ bớt đi một biến từ $M_k$
  - Chọn mô hình tốt nhất (có $L(\theta)$ nhỏ nhất) trong số $k$ mô hình, gọi nó là $M_{k-1}$.
- Chọn mô hình tốt nhất (có validation error nhỏ nhất) từ $M_0, \dots M_d$ dùng cross validation.

# Backward stepwise selection (cont)

- Giống forward stepwise selection, backward stepwise selection cũng có không gian tìm kiếm nhỏ hơn nhiều so với best subset selection.
  - Số trường hợp phải xét là $1 + d(d+1)/2 = O(d^2)$ so với $O(2^d)$
- Backward stepwise selection cũng là phương pháp greedy nên không đảm bảo tìm được mô hình tốt nhất như best subset selection
- Backward stepwise selection cần $n > d$ (số điểm dữ liệu $n$ lớn hơn số biến $d$) để có thể fit full model
  - Nếu $n < d$, ta có thể dùng forward stepwise selection.

# Sự điều chuẩn (Regularization)

# Độ nhạy (sensitivity) và sự điều chuẩn (regularization)

- Giả sử ta có mô hình $\hat{y} = f_\theta(x) = \theta^T x$
- Nếu $\theta_i$ lớn, thì $\hat{y}_i$ sẽ rất nhạy cảm (sensitive) với $x_i$
  - Thay đổi nhỏ trong $x_i$ dẫn đến thay đổi lớn trong $\hat{y}_i$
- Quá nhạy cảm (large sensitivity) có thể dẫn đến quá khớp (overfit) và không tổng quát (poor generalization)
- Ta đo độ lớn của $\theta$ dùng một hàm điều chuẩn (regularizer function) $r : \mathbb{R}^d \to \mathbb{R}$
- $r(\theta)$ là một độ đo cho độ lớn của $\theta$
  - Square regularizer ($l_2$)
$$r(\theta) = ||\theta||_2 = \theta_1^2 + \cdots + \theta_d^2$$
  - Absolute regularizer ($l_1$)
$$r(\theta) = ||\theta||_1 = |\theta_1| + \cdots + |\theta_d|$$

# Regularized empirical risk minimization

- Mô hình nên khớp (fit) với dữ liệu cho trước tốt, tức sai số thực nghiệm (empirical risk) $L(\theta)$ nhỏ

$$L(\theta) = \frac{1}{n}\sum_{i=1}^{n} l(\theta^T x_i, y_i)$$

- Mô hình nên không quá nhạy cảm (sensitive), tức $r(\theta)$ nhỏ
- Để cân bằng hai mục tiêu này, ta dùng sai số thực nghiệm đã điều chuẩn (regularized empirical risk)

$$L(\theta) + \lambda r(\theta)$$

  trong đó: $\lambda > 0$ là regularization parameter (hay hyper-parameter)

- Regularized empirical risk minimization (RERM): chọn $\theta$ để regularized empirical risk đạt cực tiểu

# Regularized empirical risk minimization (cont)

- Với $\lambda = 0$, RERM trở thành ERM
- Với $\lambda = \infty$, $\theta = 0$
- RERM tạo ra một họ các mô hình ứng với các $\lambda$ khác nhau
- Ta sẽ chọn vài (chục) giá trị $\theta$, thường là theo khoảng cách logarit trên một miền giá trị lớn
- Ta dùng cross validation để chọn mô hình tốt nhất
- Thường ta sẽ chọn giá trị $\lambda$ lớn nhất cho test error gần giá trị cực tiểu (để nó ít nhạy cảm và tổng quát tốt)

# Ridge regression ($L_2$ regularization)

- Chọn $\lambda$ để cực tiểu hóa

$$L(\theta) + \lambda||\theta||_2^2$$

$$\begin{aligned} L(\theta) &= \sum_{i=1}^{n}(y_i - \theta^T x_i)^2 \\ &= (y - X\theta)^T(y - X\theta) \end{aligned}$$

$$cost(\theta) = L(\theta) + \lambda||\theta||_2^2 = (y - X\theta)^T(y - X\theta) + \lambda\theta^T\theta$$

# Gradient of ridge regression cost

$$
\begin{aligned}
\nabla[cost(\theta)] &= \nabla[L(\theta) + \lambda||\theta||_2^2] \\
&= \nabla[(y - X\theta)^T(y - X\theta) + \lambda\theta^T\theta] \\
&= \nabla[(y - X\theta)^T(y - X\theta)] + \lambda\nabla[\theta^T\theta] \\
&= -2X^T(y - X\theta) + 2\lambda\theta \\
&= -2X^T(y - X\theta) + 2\lambda I_n\theta
\end{aligned}
$$

# Analytical solution for ridge regression

- Nghiệm $\hat{\theta}$ thỏa:

$$\nabla[cost(\hat{\theta})] = -2X^T(y - X\hat{\theta}) + 2\lambda I_n\hat{\theta} = 0$$

$$-X^T y + X^T X\hat{\theta} + \lambda I_n\hat{\theta} = 0$$
$$X^T X\hat{\theta} + \lambda I_n\hat{\theta} = X^T y$$

$$\hat{\theta} = (X^T X + \lambda I_n)^{-1} X^T y$$

# Analytical solution for ridge regression (cont)

$$\hat{\theta} = (X^T X + \lambda I_n)^{-1} X^T y$$

- Nếu $\lambda = 0$, $\hat{\theta}^{ridge} = (X^T X)^{-1} X^T y = \hat{\theta}^{LS}$
  - $(X^T X)$ là ma trận bậc $d \times d$
  - Độ phức tạp của việc tính $(X^T X)^{-1}$ là $O(d^3)$
- Nếu $\lambda = \infty$, $\hat{\theta}^{ridge} = 0$
  - Với $\lambda > 0$, $(X^T X + \lambda I_n)$ luôn khả nghịch

# Stochastic gradient descent for ridge regression

$t = 0$

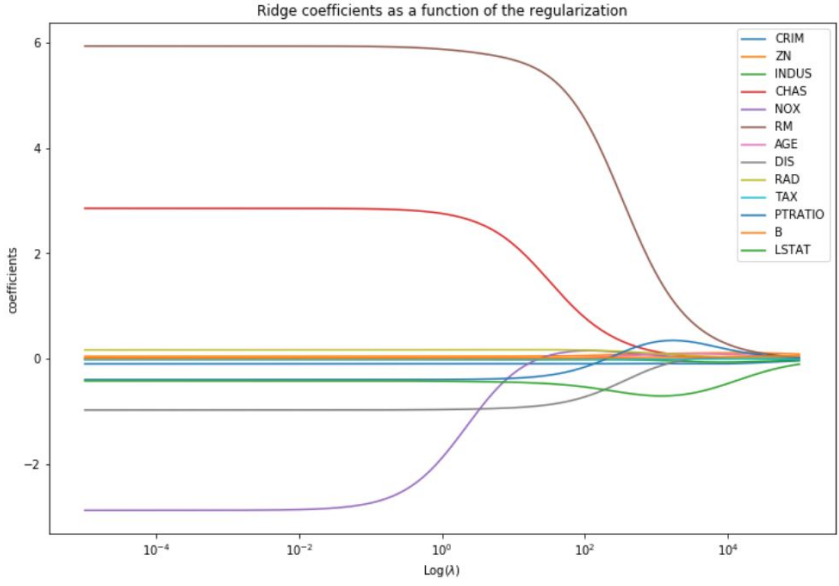
Khởi tạo $\theta^{(t)}$ ngẫu nhiên

repeat

- foreach $i = 1, 2, \ldots, n$ (thứ tự ngẫu nhiên)
  - $partial[i] = -x_i(y_i - x_i^T\theta_i^{(t)}) + \frac{\lambda}{n}\theta_i^{(t)}$
  - $\theta^{(t+1)} = \theta^{(t)} - \eta \cdot partial[i]$
- $t \leftarrow t + 1$

until $||\theta^{(t)} - \theta^{(t-1)}|| \leq \delta$

# Ridge regression on housing data



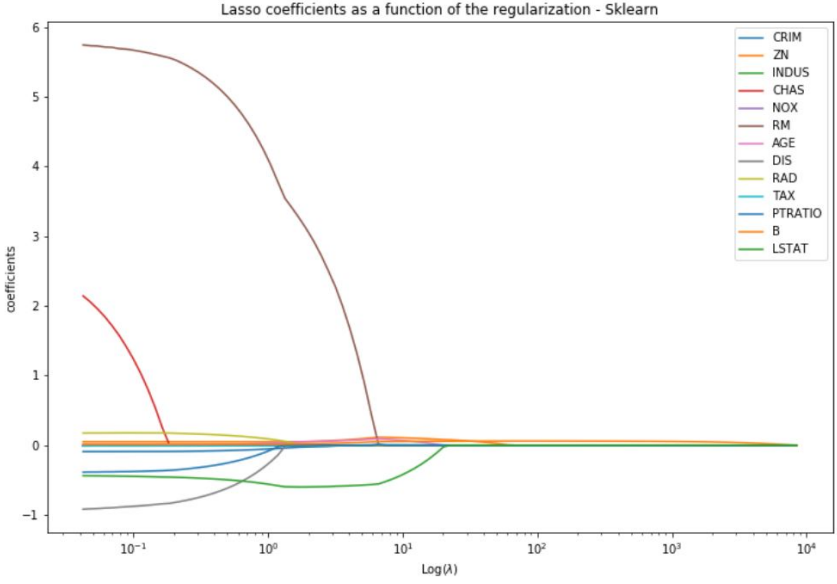
# Lasso regression ($L_1$ regularization)

- Chọn $\lambda$ để cực tiểu hóa

$$L(\theta) + \lambda||\theta||_1$$

- Lasso regression không có analytical solution
- Một số thuật toán cho lasso regression:
  - Least angle regression (LARS): "Bradley Efron, Trevor Hastie, Iain Johnstone, and Robert Tibshirani, Least angle regression, Annals of Statistics, Volume 32, Number 2 (2004), 407-499."
  - Coordinate descent: "Jerome Friedman, Trevor Hastie, and Robert Tibshirani, Sparse inverse covariance estimation with the graphical lasso, Biostatistics, Volume 9, Issue 3 (2008), 432-441"
  - ...
- $L_1$ regularization thường dẫn đến nghiệm thưa (sparsity solution), nên nó được xem như một phương pháp lựa chọn đặc trưng (feature selection).

# Lasso regression on housing data



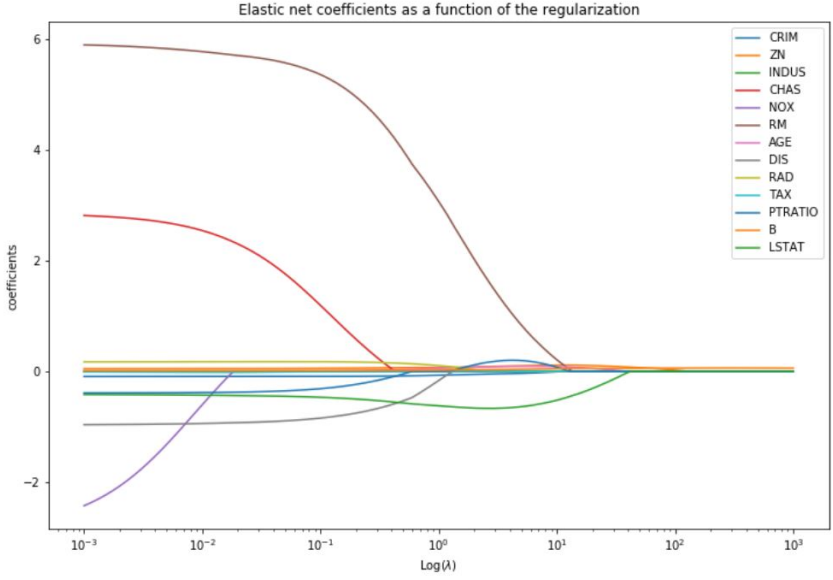
# Elastic net regression (kết hợp $L_1$ và $L_2$ regularization)

- Chọn $\lambda, \alpha$ để cực tiểu hóa

$$L(\theta) + \lambda(\frac{1-\alpha}{2}||\theta||_2^2 + \alpha||\theta||_1)$$

- Elastic net regression cũng không có analytical solution
- Thuật toán cho elastic net regression:
  - LARS-EN: "Hui Zou and Trevor Hastie, Regularization and variable selection via the elastic net, Journal of the royal statistical society: series B (statistical methodology), Volume 67, Issue 2 (2005), 301-320."

# ElasticNet regression on housing data



# Summary

- Hồi quy tuyến tính (linear regression)
- Đánh giá (Validation)
  - Holdout
  - Cross validation
- Linear model selection
  - Best subset selection
  - Forward/Backward stepwise selection
- Sự chỉnh hóa (Regularization)
  - Ridge regression
  - Lasso regression
  - ElasticNet regression