

Modelowanie Matematyczne - Lab 1

Adam Przemysław Chojecki

13 maja 2024

Zadanie 1

1. Wygeneruj $n = 100$ liczb losowych $x \sim U[-10, 10]$.
2. Przekształć te liczby według wzoru:

$$y = b_1x + b_2x^2 + b_3 \sin(x) + b_4 \cos(x) + b_5|x| + \epsilon$$

gdzie ϵ to błąd losowy z rozkładu normalnego o $sd = 1$, a wektor współczynników jest dany jako:

$$b = (5.1, -0.4, 1.3, 10.1, 0.0, -3.8)$$

3. Narysuj wykres wylosowanych y w zależności od x .
4. Użyj funkcji `lm()` w R, aby dopasować model liniowy:

$$y \sim x + I(x^2) + \sin(x) + \cos(x) + abs(x)$$

Zwróć uwagę, że z powodów technicznych zmienna $I(x^2)$ musi być obłożona funkcją $I()$, jeśli ma być użyta w funkcji `lm()`.

5. Narysuj wykres przedstawiający rzeczywiste wartości y oraz dopasowane wartości modelu liniowego.
6. Porównaj współczynniki uzyskane z modelu liniowego z rzeczywistymi współczynnikami.
 - Zwróć uwagę, że w oryginalnych danych y nie ma części związanej z $\cos(x)$ (jest ona mnożona przez 0). Czemu więc model daje jej niezerową wartość? Jak myślisz, co by się stało, gdyby dać modelowi więcej funkcji do dostosowania się do nich? Np. $y \sim x + I(x^2) + I(x^3) + \dots + I(x^{50})$? Czy model byłby lepszy, czy gorszy?
 - Eksperymentuj ze zmienną liczbą danych n oraz dodanego szumu sd : Czy zwiększając liczbę danych uczących, jakość modelu się znacząco poprawia? Do jakiego momentu? Czy zwiększanie wartości sd w stopniu znaczącym pogarszało model?

- Intuicyjnie odpowiedz na pytanie: Czy dla dużej wartości $sd = 100$ istnieje tak duża wartość $n = ?$, że model jest dobrze dopasowany (z dokładnością do 1 miejsca po przecinku)? Swoją intuicję sprawdź eksperymentalnie. (Uważaj, aby nie kazać R rysować dużej liczby punktów)
- Intuicyjnie odpowiedz na pytanie: Czy dla dużej wartości $sd = 100$ istnieje tak duża wartość $n = ?$, że model jest dobrze dopasowany (z dokładnością do 1 miejsca po przecinku)? Swoją intuicję sprawdź eksperymentalnie. (Uważaj, aby nie kazać R rysować dużej liczby punktów)
- Dopasuj model bez części x^2 . Jak się on pogarsza? Zinterpretuj pierwszy obrazek wyświetlany po wywołaniu `plot()` na modelu.

Podpowiedzi:

1. Użyj funkcji `rnorm()`.
2. Użyj zapisu wektorowego.
3. Użyj `plot(..., pch = 16)` dla ładniejszych punktów.
4. Popatrz na przykłady we wbudowanej do R dokumentacji: `?lm`.
5. Użyj funkcji `predict()` na modelu z punktu 4. Później użyj `points(..., pch = 16, col = "red")`, aby dodać nowe punkty do obrazka.
6. Użyj funkcji `coef()` na modelu z punktu 4.

Zadanie 2

Wykorzystując zbiór danych "mtcars" wbudowany w R, wykonaj następujące czynności:

1. Wczytaj zbiór danych "mtcars" używając funkcji `data(mtcars)`.
2. Zidentyfikuj dostępne zmienne w zbiorze danych oraz ich typy.
3. Chcemy przewidzieć zmienną `qsec`. Dowiedz się, co ona oznacza oraz zastanów się, które z pozostałych zmiennych mogą mieć sens jako zmienne dla liniowego modelu predykcyjnego.
4. Narysuj wykresy punktowe (scatter plots) przedstawiające związek między zmienną `qsec` a każdą z potencjalnie znaczących zmiennych niezależnych.
5. Użyj funkcji `lm()`, aby dopasować model liniowy, gdzie `qsec` będzie zmienną zależną, a wybrane przez Ciebie zmienne będą zmiennymi niezależnymi.
6. Narysuj wykres przedstawiający rzeczywiste wartości `qsec` oraz dopasowane wartości modelu liniowego. Czy w Twoim modelu jakaś obserwacja nie pasuje?

7. Sprawdź statystyki modelu za pomocą funkcji `summary()` oraz zinterpretuj uzyskane współczynniki.

Podpowiedzi:

1. Użyj funkcji `head()` do wyświetlenia kilku pierwszych wierszy danych.
2. Zastosuj funkcję `str()`, bądź `head()`.
3. W R jest też dokumentacja dla zbiorów danych: `?mtcars`.
4. Użyj funkcji `plot()`, bądź `pairs()`.
5. Popatrz na przykłady w `?lm()`.
6. Użyj funkcji `predict()`, a potem `plot()`.
7. `summary()` wywołane na modelu liniowym zawiera bardzo dużo przydatnych informacji, którymi będziemy się zajmować w przyszłych tygodniach.