11MAMY – Cvičení 9

Klasifikace

Jan Přikryl

ČVUT FD

čtvrtek 7. dubna 2022

verze: 2022-04-06 19:51

Příklad 1

Data z akciového trhu

Prozkoumáme nejprve data z akciových trhů, konkrétně denní vývoj indexu S&P v letech 2001–2005.

Načteme a zobrazíme základní charakteristiky

```
smarket = ...;
summary(...)
```

Proměnná `Direction` je kategorická (buď `Up` nebo `Down`) a je potřeba to Matlabu sdělit:

```
smarket.Direction = ...;
```

Příklad 1

Pokračování

Pokud bychom chtěli zkoumat korelace mezi jednotlivými numerickými proměnnými, nabízí se funkce `corrcoef()` ...

```
corrcoef(smarket)
```

...jenže jako vstup je potřeba matice:

```
smarket_mat = ...; % Ne všechny sloupce!
smarket_cc = corrcoef(...);
```

Vysvětlete, proč indexujeme `smarket{:,2:end-1}`. Jaké jsou jiné možnosti?

Příklad 1

Pokračování

Pokud bychom chtěli zkoumat korelace mezi jednotlivými numerickými proměnnými, nabízí se funkce `corrcoef()` ...

```
corrcoef(smarket)
```

...jenže jako vstup je potřeba matice:

```
smarket_mat = ...; % Ne všechny sloupce!
smarket_cc = corrcoef(...);
```

Vysvětlete, proč indexujeme `smarket{:,2:end-1}`. Jaké jsou jiné možnosti?

Najdete v korelační matici `smarket_cc` hodnoty naznačující, že nějaké veličiny jsou korelované? Pokud ano, kterým proměnným odpovídají?

Příklad 1

Pokračování

Pokud bychom chtěli zkoumat korelace mezi jednotlivými numerickými proměnnými, nabízí se funkce `corrcoef()` ...

```
corrcoef(smarket)
```

...jenže jako vstup je potřeba matice:

```
smarket_mat = ...; % Ne všechny sloupce!
smarket_cc = corrcoef(...);
```

Vysvětlete, proč indexujeme `smarket{:,2:end-1}`. Jaké jsou jiné možnosti?

Vykreslíme

```
plot(smarket.Volume)
```

Na základě grafu vysvětlete, proč je mezi `Year` a `Volume` pozitivní korelace.

Příklad 2

Logistická regrese

Zkusme použít jednoduchý klasifikátor pro předpověď směrování trhu. Budeme chtít předpovědět, zda trh roste nebo padá.

Natrénujeme generalizovaný lineární model závislosti `Direction` na `Lag1` až `Lag5`. Logistickou závislost specifikujeme volbou `'Distribution'`, `'binomial'`:

```
mdl = fitglm(...,
       '...',
       'Distribution', 'binomial')
```

Q: Který regresní koeficient má nejmenší p-hodnotu? Naznačuje tato hodnota silnou vazbu na výstup modelu?

Příklad 2

Pokračování

Podívejme se na predikci modelu na původních pozorováních:

```
probs = predict(mdl)
```

Jak dobře model predikuje vývoj trhu zjistíme porovnáním s trénovacími hodnotami v `Direction`. Musíme ale `probs` převést na kategorickou proměnnou s hodnotami `Up` a `Down`:

```
predictions = repmat(categorical({'Down'}), mdl.NumObservations, 1);
predictions(probs>0.5) = 'Up'; % Boolovská indexace, 'Up' -> cat()
```

Pokračujeme maticí záměn:

```
confusionmat(...)
err = ... % Třeba (TP+TN)/N
mean(...) % Nebo průměrná chyba
```

Příklad 2

Je náš model lepší, než náhodné rozhodování? Jaká je jeho trénovací chyba?

Lepší odhad chyby, kterou model bude v reálu vykazovat, lze získat rozdělením na trénovací a testovací sadu. Zkusme identifikovat model na datech z let 2001–2004 a ověřit jeho předpovědi na datech z roku 2005.

```
train = ...; % Indexy dat z let < 2005, logický vektor true/false
```

Q: Co znamená `~train`?

```
smarket_train = smarket(...); % Trénovací data podle indexu
smarket_test = smarket(...); % NE trénovací data
```

Jak velká je trénovací a testovací množina?

```
...(smarket_train)
...(smarket_test)
```

Příklad 2

Pokračování

Identifikujeme model a porovnáme jej na datech z roku 2005:

```
mdlt = fitglm(...,
        'Direction␣~␣Lag1+Lag2+Lag3+Lag4+Lag5+Volume',
        'Distribution', 'binomial')
prob_test = ...(...); % Predikuj testovací výstupy
% Převod na Up/Down s prahem 0.5
predictions = repmat(categorical({'Down'}),...,1);
predictions(...) = 'Up'; % Převede se na kategorii
% Matice záměn a procento správných předpovědí
confusionmat(..., ...)
...(...) % Stačí spočítat průměrnou shodu
```

Q: Jaká je chyba testovací sady?

Příklad 2

Pokračování

Identifikujeme jednodušší model pouze se členy Lag1 a Lag2, které v originální logistické regresi měly nejsilnější vztah k výstupu:

```
mdt2 = fitglm(...,
              '...',
              'Distribution', 'binomial')
prob_te2 = ...(...);
predictions = repmat(categorical({'Down'}), ..., 1);
predictions(...) = 'Up';
confusionmat(..., ...)
mean(...)
```

Q: Jaký je odhad testovací chyby nyní? Jaká je pravděpodobnost předpovědi růstu trhu? Poklesu trhu?

Příklad 2

Pokračování

Na závěr si ukážeme, jak spočítat predikce u nových hodnot Lag1 a Lag2 daných následující tabulkou:

Lag1	Lag2
1,2	1,1
1,5	−0,8

```
% Vytvoříme novou Matlabí tabulku
pt = table([1.2;1.5], [1.1;-0.8], 'VariableNames', {'Lag1', 'Lag2'});
% Vyhodnotíme model na datech uložených v ‘pt‘
...(...)'
```

Místo tabulky můžete v tomto případě použít i pt reprezentované maticí. Jak to uděláte?

Příklad 3

Diskriminační analýza

Nyní zkusíme to samé pomocí lineární diskriminační analýzy. V Matlabu je na to obecná metoda `fitcdiscr()`, implementující i vyšší polynomiální reprezentace hranice. Vstupem metody byla standardně zvlášť **matice** prediktorů a zvlášť **vektor** odpovědi modelu:

```
x = [ smarket_train.Lag1, smarket_train.Lag2 ];
y = smarket_train.Direction;
cmdl = fitcdiscr(x,y)
```

Vidíme, že `cmdl` neobsahuje údaje o názvech proměnných, doplníme:

```
cmdl = fitcdiscr(x, y, 'PredictorNames', {'Lag1','Lag2'},
                       'ResponseName', 'Direction')
```

V nových verzích Matlabu lze volat i s tabulkou — zkuste `fitcdiscr(smarket_train,'Direction␣~␣Lag1+Lag2')`.

Příklad 3

Pokračování

Zkusíme si vykreslit hranici a hodnoty v jednotlivých třídách. Podívejte se nejprve, k čemu slouží funkce `gscatter()` a `ezplot()`.

```
% Vykreslíme data a jejich třídu Up/Down
gscatter(..., ..., ...);
hold on
% Definice funkce pro ezplot()
K = cmdl.Coeffs(1,2).Const;
L = cmdl.Coeffs(1,2).Linear;
f = @(x1,x2) ...; % Lineární hranice
% Vykreslime hranici
h2 = ezplot(f, [-6,6,-6,6]);
```

Příklad 3

Pokračování

Matice záměn a celková testovací chyba modelu je totožná s logit modelem:

```
xtest = [smarket_test.Lag1, smarket_test.Lag2];
predictions = cmdl.predict(xtest);
confusionmat(predictions, smarket_test.Direction)
mean(predictions == smarket_test.Direction)
```

Samostatná práce

Kvadratická diskriminační analýza a KNN

Samostatně vyzkoušejte:

(a) kvadratickou diskriminační analýzu,

(b) klasifikaci pomocí metody k nejbližších sousedů (KNN).