Politechnika Warszawska
Wydział Matematyki i Nauk Informacyjnych

PRACA DYPLOMOWA MAGISTERSKA
NA KIERUNKU MATEMATYKA

STATYSTYCZNE METODY REGRESJI PORZĄDKOWEJ

Autor:
Marta Sommer

Promotor:
prof. nzw. dr hab.
Przemysław Grzegorzewski

Warszawa, czerwiec 2015

.....

podpis promotora

.....

podpis autora

Spis treści

Wstęp **5**

1. Opis teoretyczny dostępnych metod **7**
1.1. Postawienie problemu i podstawowe oznaczenia 7
1.2. Model proporcjonalnych szans 7
1.3. Wektory maszyn podpierających (SVM) 8
1.4. Sieci neuronowe 9
1.5. Metoda zaproponowana przez E. Franka i M. Halla 11
1.6. Procesy gaussowskie 13

2. Diagnostyka modelu **17**
2.1. Krzywa ROC w przypadku dwuklasowym 17
2.2. Krzywa ROC w przypadku regresji porządkowej 20
2.3. Współczynnik VUS 22

A. fdaaggfdadfsd **23**

Literatura **25**

Wstęp

Regresja porządkowa (ang. *ordinal regression*) jest jednym z działów uczenia maszynowego. Od problemu klasycznej regresji różni ją to, że zmienna odpowiedzi jest dyskretna, natomiast od problemu klasyfikacji to, że zmienna odpowiedzi ma pewien naturalny porządek. Regresja porządkowa zajmuje się zatem uczeniem i oceną jakości predyktora, który modeluje zmienną uporządkowaną i skończoną. Problem regresji porządkowej rozwija się dość szybko m.in. dlatego, że ma on bardzo wiele zastosowań, choćby w systemach rekomendacji, czy bardzo popularnych wyszukiwarkach internetowych. Prześledźmy to na konkretnym przykładzie. Wyobraźmy sobie sytuację, że chcielibyśmy określić, w jakim stopniu danemu człowiekowi spodoba się sprzedawany przez nas produkt. Mamy do dyspozycji zbiór treningowy składający się z wektora zmiennej objaśniającej $\mathbf{x} = (x_1, \ldots, x_d)$, gdzie x_i są różnymi cechami określającymi daną osobę (np. płeć, wiek, wykształcenie, ...). Cechy te – podobnie jak w przypadku zwykłej regresji – mogą być zarówno ciągłe, jak i dyskretne. Mamy również dostęp do zmiennej objaśnianej $\mathbf{y} = (y_1, \ldots, y_r)$, będącej wektorem zero-jedynkowym, wskazującym która klasa została przypisana danemu rekordowi. W naszym przykładzie, zmienną odpowiedzi mogłyby być na przykład: *zdecydowanie mi się nie podoba*, *nie podoba mi się*, *nie mam zdania*, *podoba mi się*, *zdecydowanie mi się podoba*. Widać wyraźnie, że są one uporządkowane.

Najprostszym podejściem do tego typu problemu byłoby zignorowanie kolejności zmiennej odpowiedzi i potraktowanie go, jak zwykłą klasyfikację. W takim przypadku tracimy jednak pewną informację, która prawdopodobnie mogłaby przyczynić się do poprawy naszego klasyfikatora. Idąc w drugą stronę, można potraktować nasz problem, jak zwykłą regresję, zamieniając zmienną odpowiedzi na pewną zmienną ciągłą i to ją modelując, a następnie z powrotem dyskretyzować. Pojawia się tu jednak problem, jak optymalnie zrobić taką transformację, uwzględniając chociażby fakt, że nasza odpowiedź niekoniecznie jest monotoniczna (tzn. np. różnica między *nie podoba mi się* a *nie mam zdania* wcale mnie musi być taka sama, jak między *podoba mi się* a *zdecydowanie mi się podoba*).

Możemy wyróżnić dwa główne nurty w regresji porządkowej:

- prognoza konkretnej obserwacji (nacisk kładziony jest tu na wyznaczenie konkretnego $\mathbf{y}$ dla konkretnego $\mathbf{x}$ np. czy potencjalnemu klientowi spodoba się dany produkt),
- uszeregowanie kilku obserwacji (celem nie jest poznanie estymacji konkretnej zmiennej odpowiedzi, ale takie uszeregowanie kilku rekordów, by te najbardziej preferowane znalazły się na samej górze, a te najmniej na samym dole np. w jakiej kolejności powinny wyświetlić się znalezione strony w wyszukiwarce).

W mojej pracy zajmować się będę przede wszystkim pierwszym punktem, lecz nakreślę też kilka podejść dotyczących drugiego.

Rozdział 1

Opis teoretyczny dostępnych metod

1.1. Postawienie problemu i podstawowe oznaczenia

Na wejściu dany mamy zbiór $\mathcal{D} = (\mathbf{x}^{(i)}, y^{(i)})_{i=1}^{n}$, składający się z n par $(\mathbf{x}, y)$, gdzie:

- $\mathbf{x}^{(i)}$ jest K–wymiarowym wektorem cech (częstym założeniem będzie, że $\mathbf{x}^{(i)} \in \mathbb{R}^K$),
- $y^{(i)}$ jest liczbą symbolizującą kategorię, do której przyporządkowana została i–ta obserwacja, tzn. $y^{(i)} \in \mathcal{Y}$, gdzie $\mathcal{Y} = \{1, \ldots, r\}$ jest zbiorem uporządkowanym według pewnego porządku „$\prec$”.

Naszym celem będzie stworzenie modelu, który pozwoli na wybranie najlepszej (nieznanej) kategorii $y_* \in \mathcal{Y}$ dla nowej obserwacji o zadanym wektorze cech $\mathbf{x}_*$.

W tym rozdziale opracujemy kilka rozwiązań, które pozwolą nam się z tym problemem uporać.

1.2. Model proporcjonalnych szans

Najbardziej rozpowszechnionym sposobem modelowania regresji porządkowej jest model proporcjonalnych szans. Jest to jedna z metod uogólnionych modeli liniowych, bardzo silnie opierająca się na regresji logistycznej. Interesują nas prawdopodobieństwa:

$$\Pi_j(\mathbf{x}) := \mathbb{P}(y_{=}j \mid \mathbf{x}), \quad \text{dla} \quad j = 1, \ldots, r.$$

Idea polega nie na bezpośrednim modelowaniu prawdopodobieństw $\Pi_j(\mathbf{x})$, lecz na wcześniejszym modelowaniu tzw. prawdopodobieństw skumulowanych:

$$\mathbb{P}(y \leq j \mid \mathbf{x}) = \Pi_1(\mathbf{x}) + \ldots + \Pi_j(\mathbf{x}), \quad \text{dla} \quad j = 1, \ldots, r-1.$$

Następnie rozważa się poniższy model logitowy:

$$\log \frac{\mathbb{P}(y \leq j \mid \mathbf{x})}{1 - \mathbb{P}(Y \leq j \mid \mathbf{x})} = \alpha_j + \beta^T \mathbf{x}, \quad \text{dla} \quad j = 1, \ldots, r-1,$$

gdzie $\alpha_j \in \mathbb{R}$ i $\beta \in \mathbb{R}^K$ są parametrami modelu. Należy zauważyć, że parametr β jest stały dla każdego $j = 1, \ldots, r-1$.

Współczynniki modelu – jak w przypadku regresji logistycznej – wyliczamy metodą Raphsona-Newtona, a skumulowane prawdopodobieństwa – po prostym przeliczeniu – dostaniemy ze wzoru:

$$\mathbb{P}(y \leq j \mid \mathbf{x}) = \frac{e^{\alpha_j + \beta^T \mathbf{x}}}{1 + e^{\alpha_j + \beta^T \mathbf{x}}}, \quad \text{dla} \quad j = 1, \ldots, r-1.$$

Szukane prawdopodobieństwa $\Pi_j(\mathbf{x})$ otrzymamy w poniższy sposób:

$$\begin{aligned} \Pi_1(\mathbf{x}) &= \mathbb{P}(Y \leq 1 \mid \mathbf{x}), \\ &\vdots \\ \Pi_i(\mathbf{x}) &= \mathbb{P}(Y \leq i \mid \mathbf{x}) - \mathbb{P}(Y \leq i-1 \mid \mathbf{x}), \\ &\vdots \\ \Pi_r(\mathbf{x}) &= 1 - \mathbb{P}(Y \leq r-1 \mid \mathbf{x}). \end{aligned}$$

Dla nowej obserwacji $\mathbf{x}_*$ wybieramy oczywiście klasę y_*, która maksymalizuje prawdopodobieństwa $\Pi_j(\mathbf{x}_*)$.

1.3. Wektory maszyn podpierających (SVM)

Wektory maszyn podpierających (ang. *Support Vector Machine*) to bardzo znana i powszechnie stosowana metoda klasyfikacji. W dużym uproszczeniu, polega ona na konstrukcji dwóch równoległych i maksymalnie oddalonych od siebie hiperpłaszczyzn w taki sposób, by minimalizować kwadrat odległości między nimi. By móc obsługiwać przypadki, w których brak liniowej separowalności, wprowadza się dodatkowo karę za nieidealne rozdzielenie klas. W przypadku dwuklasowym budowa modelu sprowadza się do rozwiązania następującego problemu optymalizacyjnego:

$$\min \left\{ \frac{1}{2}||\mathbf{w}||^2 + C \sum_{i=1}^{n} \xi_i \right\}$$

przy ograniczeniach:

$$\begin{cases} \mathbf{x}^{(i)^T} \mathbf{w} + b & \geq 1 - \xi_i \\ \mathbf{x}^{(i)^T} \mathbf{w} + b & \leq -1 + \xi_i, \end{cases}$$

gdzie $\mathbf{w} \in \mathbb{R}^K$, $b \in \mathbb{R}$, $C \in \mathbb{R}$ i $\xi_i \geq 0$ są parametrami modelu.

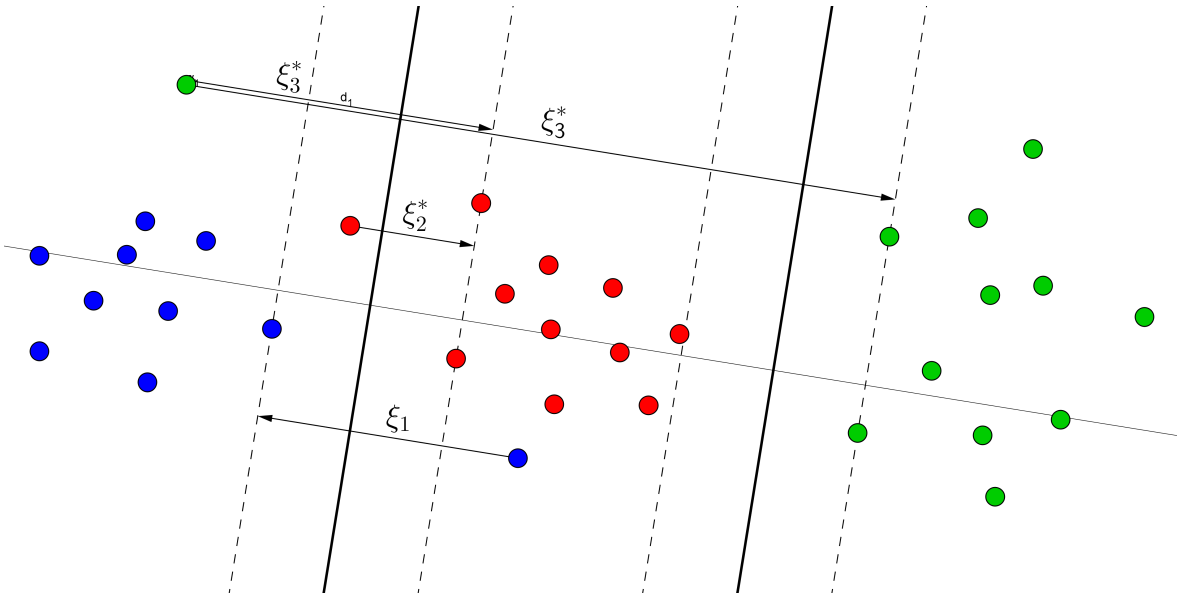
Tak to wyglądało w przypadku dwuklasowej klasyfikacji. Przyjrzyjmy się teraz, jak w łatwy sposób można zaadaptować powyższą metodę do rozważanej przez nas regresji porządkowej.

Tym razem optymalizować będziemy następujące wyrażenie:

$$\min_{\mathbf{w}, b_k} \left\{ \frac{1}{2}||\mathbf{w}||^2 + C \sum_{j=1}^{r-1} \left(\sum_{k=1}^{j} \sum_{i=1}^{n_k} \xi_{ki}^{j} + \sum_{k=j+1}^{r} \sum_{i=1}^{n_k} \xi_{ki}^{*j} \right) \right\}$$

przy ograniczeniach:

$$\begin{cases} \mathbf{w} x_k^{(i)} - b_j & \leq -1 + \xi_{ki}^{j}, \quad \text{dla} \quad k = 1, \ldots, j \quad \text{oraz} \quad i = 1, \ldots, n_k \\ \mathbf{w} x_k^{(i)} - b_j & \geq +1 - \xi_{ki}^{*j}, \quad \text{dla} \quad k = j+1, \ldots, r \quad \text{oraz} \quad i = 1, \ldots, n_k \end{cases}$$



Rysunek 1.1: Przykładowa klasyfikacja metodą SVM.

Przyjrzyjmy się, czym różni się nasz nowy problem od problemu optymalizacyjnego w standardowej klasyfikacji. Przede wszystkim – podobnie jak w modelu proporcjonalnych szans – mamy tu do czynienia z $(r-1)$–hiperpłaszczyznami, rzutowanymi na jeden, wspólny dla wszystkich obserwacji, kierunek $\mathbf{w}$. Przy wyznaczaniu kolejnych hiperpłaszczyzn, bierzemy pod uwagę wszystkie klasy. Kary naliczane są więc w następujący sposób. Dla progu b_j, wartości funkcji $\mathbf{w}x_k^{(i)}$ dla obserwacji z niższych klas, powinny być niższe niż dolna granica $b_j - 1$. Jeśli tak nie jest, wtedy jako błąd próbki ξ_{ki}^j dla progu b_j uznaje się $\xi_{ki}^j = \mathbf{w}x_k^{(i)} - (b_j - 1)$. Analogicznie, dla obserwacji z wyższych klas, otrzymujemy błędy ξ_{ki}^{*j}.

Budowa modelu i tym razem sprowadza się więc do problemu optymalizacyjnego.

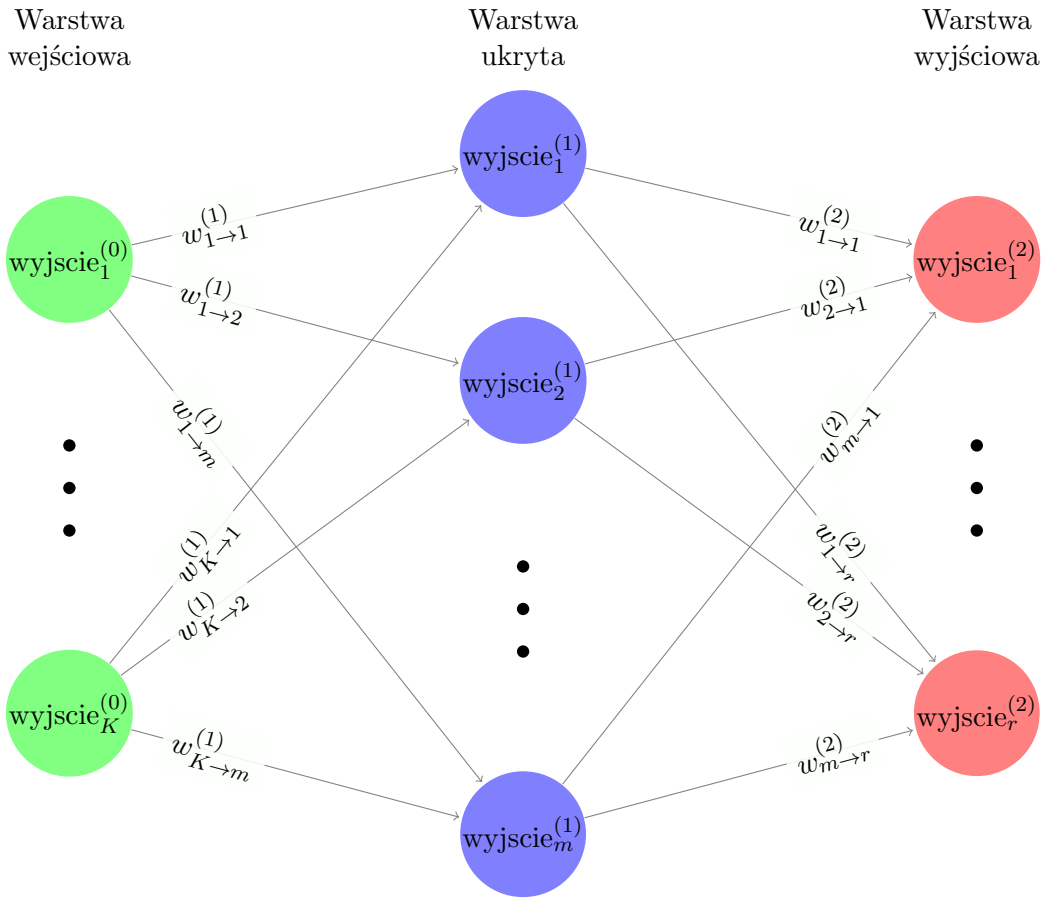
1.4. Sieci neuronowe

Sieci neuronowe to bardzo proste i szeroko stosowane narzędzie zarówno w problemach regresji, jak i klasyfikacji. Znalazło ono również swoje zastosowanie w regresji porządkowej (por. [2]).

Standardowo, na wejściu otrzymujemy zbiór uczący w postaci n par $(\mathbf{x}, y)$, gdzie $\mathbf{x} = (x_1, \ldots, x_K)^T$ jest wektorem cech, a y numerem klasy. Tym razem jednak, dodatkowo modyfikujemy zmienną odpowiedzi w taki sposób, by zamiast liczby rzeczywistej otrzymać zero-jedynkowy wektor odpowiedzi $\mathbf{y} = (y_1, \ldots, y_r)^T$ reprezentujący klasę, do której należy dana obserwacja, tzn. $y_i = \mathbb{I}\{y = i\}$.

W przeciwieństwie do zwykłej klasyfikacji, nasza sieć neuronowa będzie zakładać porządek zmiennej odpowiedzi. W jaki sposób? Mianowicie, jako wektor wyjściowy, zamiast wektora $\mathbf{y} = (0, 0, \ldots, 1, \ldots, 0)^T$, mającego jedynkę na i–tym miejscu, jeśli obserwacja należała do i–tej klasy, rozważać będziemy wektor $\mathbf{y} = (1, 1, \ldots, 1, \ldots, 0)^T$, mający jedynki na miejscach od pierwszego do i–tego.

Otrzymujemy w ten sposób sieć neuronową o K neuronach w warstwie wejściowej, z których każdy reprezentuje inną cechę z wektora $\mathbf{x}$, jednej (bądź więcej) warstwie ukrytej o m neuronach i warstwie wyjściowej zawierającej r neuronów, które reprezentują odpowiedź $\mathbf{y}$ w formie opisanej powyżej. Za funkcję przejścia przyjmiemy funkcję sigmoidalną $f(x) = \frac{1}{1+e^{-x}}$, dobrze reprezentującą przynależność do danej klasy jako prawdopodobieństwo. Może się zdarzyć, że wyjściowy wektor nie będzie ciągiem malejącym (co trochę przeczy intuicji), jednak nie jest to konieczne do robienia predykcji.



Rysunek 1.2: Przykładowa sieć neuronowa.

Uczenie sieci neuronowej będzie się odbywało algorytmem propagacji wstecznej z kwadratową funkcją straty (można też użyć jakiejś innej np. entropii). Algorytm wygląda następująco:

1. Wybieramy małe wagi początkowe oraz niewielki współczynnik $\eta > 0$.
2. Losujemy parę $(\mathbf{x}, \mathbf{y})$ ze zbioru uczącego.
3. Przebiegamy sieć w przód.
4. Przebiegamy sieć w tył (licząc błąd dla każdego neuronu).
5. Zmieniamy wagi.
6. Dopóki nie osiągniemy zadowalająco niskiego błędu, wracamy do punktu 2).

Ad. 3)

Dla każdego neuronu obliczamy wartość wejściową ze wzoru:

$$wejscie_j^{(i)} = \sum_{k:\ \exists\ w_{k\to j}^{(i)}} \left(w_{k\to j}^{(i)} \cdot wyjscie_k^{(i-1)} \right),$$

gdzie $wyjscie_i^{(0)} = x_i$. A następnie wyjściową:

$$wyjscie_j^{(i)} = f\left(wyjscie_j^{(i)} \right).$$

Ad. 4)

Dla warstwy wyjściowej błąd ma postać:

$$\delta_j = -2 \cdot wyjscie_j^2 \cdot (1 - wyjscie_j)^2 \cdot (y_j - wyjscie_j),$$

zaś dla warstw ukrytych:

$$\delta_j^{(i)} = wyjscie_j^{(i)} \cdot \left(1 - wyjscie_j^{(i)}\right) \cdot \sum_{k:\ \exists\ w_{j\to k}^{(i+1)}} \left(w_{j\to k}^{(i+1)} \cdot \delta_k^{(i-1)} \right).$$

Ad. 5)

Modyfikacja wag przebiega następująco:

$$w_{k\to j}^{(i)(new)} = w_{k\to j}^{(i)(old)} - \eta \cdot \delta_j^{(i)} \cdot wyjscie_k^{(i-1)}.$$

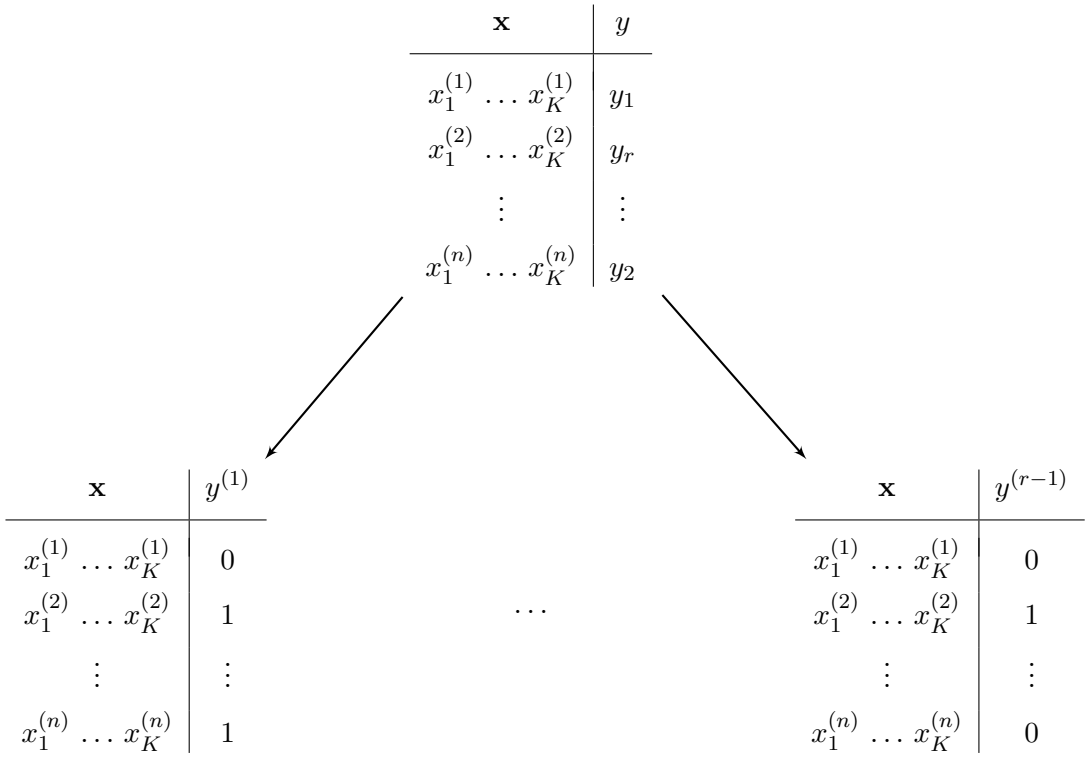
Predykcja opiera się już tylko na przejściu algorytmu w przód z nowymi obserwacjami wejściowymi $\mathbf{x}_*$ i ustaleniu progu (najczęściej równego $0,5$), klasyfikującego neuron wyjściowy jako jedynkę. Skanujemy wektor wyjściowy zaczynając od y_1 i kończymy, gdy pierwszy raz natkniemy się na 0. Przypisujemy obserwacji taką klasę, jaką długość miał znaleziony przez nas ciąg jedynek.

1.5. Metoda zaproponowana przez E. Franka i M. Halla

Podejście Franka i Halla (por. [1]) do zagadnienia regresji porządkowej jest nieco inne, niż przedstawione do tej pory metody. Opiera się bowiem nie na stworzeniu nowego modelu, ale na odpowiednim przedefiniowaniu zbioru danych, a następnie na sprowadzeniu zadania do problemu zwykłej klasyfikacji z dwoma klasami. Przekształcamy zatem r–klasowy model regresji porządkowej do $(r-1)$ dwuklasowych problemów klasyfikacji.

Algorytm do budowy modelu wygląda następująco:

1. Modyfikujemy zbiór uczący (otrzymując $r-1$ nowych zbiorów uczących).
2. Dla każdego nowo uzyskanego zbioru danych dopasowujemy zwykły model klasyfikacyjny (np. drzewo) taki, który zwraca prawdopodobieństwa przynależności do klas.
3. Robimy predykcję dla nowej obserwacji.



Rysunek 1.3: Modyfikacja przykładowego zbioru uczącego.

Ad. 1)

Chcemy otrzymać $r-1$ nowych zbiorów o zero-jedynkowej zmiennej odpowiedzi. W jaki sposób to zrobić? Macierz atrybutów pozostaje bez zmian, zmienia się jedynie wektor zmiennej odpowiedzi według zasady:

$$\begin{aligned} y_i^{(1)} &= \mathbb{I}\{y_i > 1\} \\ &\vdots \\ y_i^{(r-1)} &= \mathbb{I}\{y_i > r-1\} \end{aligned}$$

Ad. 3)

Dla nowego wektora atrybutów $\mathbf{x}$ robimy predykcję na $r-1$ modelach uzyskanych w punkcie 2). Zwracamy jednak nie predykcję klasy, ale prawdopodobieństwo przynależności do klasy pierwszej. Uzyskujemy w ten sposób $r-1$ następujących prawdopodobieństw:

$$\begin{aligned} \mathbb{P}(y &> 1) \\ &\vdots \\ \mathbb{P}(y &> r-1). \end{aligned}$$

Nas natomiast interesują prawdopodobieństwa:

$$\begin{aligned} \mathbb{P}(y &= 1) \\ &\vdots \\ \mathbb{P}(y &= r-1). \end{aligned}$$

Łatwo otrzymamy korzystając z następującego wzoru łańcuchowego:

$$\begin{aligned} \mathbb{P}(y=1) &= 1 - \mathbb{P}(y>1) \\ &\vdots \\ \mathbb{P}(y=i) &= \mathbb{P}(y>i-1) - \mathbb{P}(y>i) \qquad \text{dla} \quad i=2,\ldots,r-1 \\ &\vdots \\ \mathbb{P}(y=r) &= \mathbb{P}(y>r-1). \end{aligned}$$

Ostatecznie, nowej obserwacji przypisujemy klasę, której prawdopodobieństwo $\mathbb{P}(y=i)$ było największe.

1.6. Procesy gaussowskie

Kolejną metodą modelowania problemu regresji porządkowej jest użycie procesu gaussowskiego. Jest to metoda popularna szczególne przy zwykłej regresji, znalazła ona jednak również zastosowanie w klasyfikacji zarówno jedno, jak i wieloetykietowej. Chu i Ghahramani w pracy [5] pokazują, jak rozszerzyć ją na regresję porządkową.

Pomysł modelowania regresji porządkowej polega na wprowadzeniu tzw. zmiennej ukrytej, będącej niejako krokiem pośrednim w modelowaniu zmiennej odpowiedzi. W celu wyrobienia sobie intuicji, przeanalizujmy całe rozumowanie nieco od tyłu, zaczynając od predykcji. Podstawowym założeniem *a priori* tej metody jest to, że zmienna ukryta f jest procesem gaussowskim tzn., że jej rozkłady skończenie wymiarowe są normalne. Pełną charakteryzację takiego procesu tworzą dwie informacje – średnia (standardowo przyjmuje się 0) oraz macierz kowariancji $\mathbf{\Sigma}$. Dla celów tej pracy przyjmiemy, że elementy macierzy kowariancji definiowane są w następujący sposób:

$$\mathbf{\Sigma}_{ij} = \mathbf{\Sigma}(x_i, x_j) = \exp\left\{ -\frac{\kappa}{2} \sum_{\xi=1}^{K} (x_i^\xi - x_j^\xi)^2 \right\},$$

gdzie $\kappa > 0$, a x_i^ξ oznacza ξ-ty element wektora $\mathbf{x}_i$. Zatem $\mathbf{f}|\mathbf{X} \sim \mathcal{N}(0, \mathbf{\Sigma})$, czyli:

$$\mathbb{P}(\mathbf{f}|\mathbf{X}) = \frac{1}{(2\Pi)^{\frac{n}{2}} |\mathbf{\Sigma}|^{\frac{1}{2}}} \exp\left\{ -\frac{1}{2} \mathbf{f}^T \mathbf{\Sigma}^{-1} \mathbf{f} \right\}, \tag{1.1}$$

gdzie $\mathbf{f} = [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n)]^T$ to wektor zawierający realizację zmiennej ukrytej, odpowiadający kolejnym obserwacjom ze zbioru uczącego.

Wyobraźmy sobie teraz, że dopasowaliśmy model i znamy wszystkie niezbędne parametry. W uproszczony sposób predykcja wygląda następująco:

1. na wejściu otrzymujemy nową obserwację o danym wektorze cech $\mathbf{x}_*$,
2. w pewien sposób wyliczamy dla niej liczbę rzeczywistą $f(\mathbf{x}_*)$,
3. za pomocą przekształcenia prostej rzeczywistej na r podzbiorów, wyznaczamy najlepszy y_*.

A teraz prześledźmy wszystko krok po kroku. Interesuje nas wyznaczenie y_*, dla którego prawdopodobieństwo $\mathbb{P}(y_*|\mathbf{X}, \mathbf{y}, x_*)$ jest największe. Przy pomocy zmiennej ukrytej rozpiszmy je w następujący sposób:

$$\mathbb{P}(y_*|\mathbf{X}, \mathbf{y}, \mathbf{x}_*) = \int \mathbb{P}(y_*|f(\mathbf{x}_*))\mathbb{P}(f(\mathbf{x}_*)|\mathbf{X}, \mathbf{y})df(\mathbf{x}_*). \tag{1.2}$$

Analogicznie:

$$\mathbb{P}(f(\mathbf{x}_*)|\mathbf{X}, \mathbf{y}) = \int \mathbb{P}(f(\mathbf{x}_*)|\mathbf{f})\mathbb{P}(\mathbf{f}|\mathbf{X}, \mathbf{y})d\mathbf{f}. \tag{1.3}$$

Szukamy więc powyższych prawdopodobieństw, by następnie całość scałkować.

Korzystając z informacji, że zmienna f jest procesem gaussowskim, czyli:

$$\begin{bmatrix} \mathbf{f} \\ f(\mathbf{x}_*) \end{bmatrix} \sim \mathcal{N}\left[\begin{pmatrix} \mathbf{0} \\ 0 \end{pmatrix}, \begin{pmatrix} \mathbf{\Sigma} & \mathbf{\Sigma}_* \\ \mathbf{\Sigma}_*^T & \Sigma_{**} \end{pmatrix}\right],$$

gdzie $\mathbf{\Sigma}_* = [\Sigma(\mathbf{x}_1, \mathbf{x}_*), \ldots, \Sigma(\mathbf{x}_n, \mathbf{x}_*)]^T$, a $\Sigma_{**} = \Sigma(x_*, x_*)$, otrzymujemy, że:

$$f(x_*)|\mathbf{f} \sim \mathcal{N}(\mathbf{f}^T\mathbf{\Sigma}^{-1}\mathbf{\Sigma}_*, \Sigma_{**} - \mathbf{\Sigma}\mathbf{\Sigma}^{-1}\mathbf{\Sigma}_*). \tag{1.4}$$

Zajmijmy się teraz prawdopodobieństwem $\mathbb{P}(\mathbf{f}|\mathbf{X}, \mathbf{y})$. Korzystając z podejścia bayesowskiego, rozpiszmy je jako:

$$\mathbb{P}(\mathbf{f}|\mathbf{X}, \mathbf{y}) = \frac{\mathbb{P}(\mathbf{y}|\mathbf{f})\mathbb{P}(\mathbf{f}|\mathbf{X})}{\mathbb{P}(\mathbf{y}|\mathbf{X})}. \tag{1.5}$$

Znamy już $\mathbb{P}(\mathbf{f}|\mathbf{X})$ – jest to prawdopodobieństwo *a priori* (1.1). $\mathbb{P}(\mathbf{y}|\mathbf{X})$, jako stała niezależna od $\mathbf{f}$, nie jest nam potrzebne do wyznaczenia $\hat{\mathbf{f}}$. Zostawmy je więc na razie i wróćmy do niego później, kiedy będziemy estymować parametry modelu. Zostaje nam więc do znalezienia $\mathbb{P}(\mathbf{y}|\mathbf{f})$ tzw. wiarogodność. Ponieważ wszystkie obserwacje są niezależne, otrzymujemy:

$$\mathbb{P}(\mathbf{y}|\mathbf{f}) = \prod_{i=1}^{n} \mathbb{P}(y_i|f(\mathbf{x}_i)). \tag{1.6}$$

Gdybyśmy zakładali idealną sytuację, wtedy $\mathbb{P}_{ideal}(y_i|f(\mathbf{x}_i)) = \mathbb{I}_{\{f(\mathbf{x}_i)\in(b_{y_i-1}, b_{y_i}]\}}$, gdzie $b_0 = -\infty, b_r = +\infty$, a $b_i \in \mathbb{R}$ dla $i = 1, \ldots, r-1$ to parametry modelu. Wygodniej, można b_i sparametryzować jako: $b_1 \in \mathbb{R}$ oraz $b_i = \sum_{t=2}^{j} \Delta_t + b_1$, gdzie $\Delta_t > 0$ oraz $j = 2, \ldots, r-1$. Bardzo rzadko mamy jednak do czynienia z sytuacją idealną, dlatego będziemy budować model, zakładając dodatkowy szum δ o rozkładzie $\mathcal{N}(0, \sigma^2)$. Wtedy prawdopodobieństwo zmienia się następująco:

$$\begin{aligned} \mathbb{P}(y_i|f(\mathbf{x}_i)) &= \int \mathbb{P}_{ideal}(y_i, \delta_i|f(\mathbf{x}_i))d\delta_i = \int \mathbb{P}_{ideal}(y_i|f(\mathbf{x}_i), \delta_i)\mathbb{P}(\delta_i)d\delta_i = \\ &= \int \mathbb{P}(\delta_i)\mathbb{I}_{\{f(\mathbf{x}_i)+\delta_i\in(b_{y_i-1}, b_{y_i}]\}}d\delta_i = \\ &= \Phi\left(\frac{b_{y_i} - f(x_i)}{\sigma}\right) - \Phi\left(\frac{b_{y_i-1} - f(x_i)}{\sigma}\right), \end{aligned} \tag{1.7}$$

gdzie $\Phi(\cdot)$ to dystrybuanta standardowego rozkładu normalnego.

Przejdźmy teraz do szukania najlepszej estymacji $\hat{\mathbf{f}}$. Zdefiniujmy $S(\mathbf{f}) := -\ln \mathbb{P}(\mathbf{f}|\mathbf{X}, \mathbf{y})$, wtedy mamy, że:

$$\hat{\mathbf{f}} := \underset{\mathbf{f}}{\operatorname{argmax}}\{\mathbb{P}(\mathbf{f}|\mathbf{X}, \mathbf{y})\} = \underset{\mathbf{f}}{\operatorname{argmax}}\{\ln \mathbb{P}(\mathbf{f}|\mathbf{X}, \mathbf{y})\} = \underset{\mathbf{f}}{\operatorname{argmin}}\{-\ln \mathbb{P}(\mathbf{f}|\mathbf{X}, \mathbf{y})\} = \underset{\mathbf{f}}{\operatorname{argmin}}\{S(\mathbf{f})\}.$$

Korzystając z równań (1.1), (1.5) i (1.6) można łatwo zobaczyć, że:

$$S(\mathbf{f}) \propto \sum_{i=1}^{n} l(y_i, f(\mathbf{x}_i)) + \frac{1}{2}\mathbf{f}^T \boldsymbol{\Sigma}^{-1}\mathbf{f},$$

gdzie $l(y_i, f(\mathbf{x}_i)) := -\ln \mathbb{P}(y_i|f(\mathbf{x}_i))$. Nie da się znaleźć minimum tej funkcji analitycznie, ponieważ $l(y_i, f(\mathbf{x}_i))$ nie jest gaussowska. Natomiast, żeby uzyskać najlepsze przybliżenie $\hat{\mathbf{f}}$, wystarczy zastosować do funkcji $S(\mathbf{f})$ dowolny algorytm optymalizacyjny (np. algorytm Newtona-Raphsona).

Przypomnijmy, że naszym celem jest w tej chwili wyznaczenie $\mathbb{P}(\mathbf{f}|\mathbf{X}, \mathbf{y})$. Ponieważ później będziemy chcieli obliczyć całkę (1.3), nie wystarczy nam jedynie estymator $\hat{\mathbf{f}}$, wygodnie byłoby dla nas, gdyby to prawdopodobieństwo okazało się gaussowskie. Da się to osiągnąć dzięki przybliżeniu Laplace'a.

Na początku zauważmy, że:

$$\frac{\partial^2 S(\mathbf{f})}{\partial \mathbf{f} \partial \mathbf{f}^T} = \boldsymbol{\Sigma}^{-1} + \boldsymbol{\Lambda},$$

gdzie

$$\boldsymbol{\Lambda} = \begin{bmatrix} \frac{\partial^2 l(y_1, f(\mathbf{x}_1))}{\partial^2 f(\mathbf{x}_1)} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \frac{\partial^2 l(y_n, f(\mathbf{x}_n))}{\partial^2 f(\mathbf{x}_n)} \end{bmatrix}.$$

Rozwijając funkcję $S(\mathbf{f})$ w szereg Taylora w punkcie $\hat{\mathbf{f}}$ i pamiętając, że $S'(\hat{\mathbf{f}}) = 0$, otrzymamy następujące przybliżenie:

$$S(\mathbf{f}) = S(\hat{\mathbf{f}}) + \frac{1}{2}(\mathbf{f} - \hat{\mathbf{f}})^T(\boldsymbol{\Sigma}^{-1} + \hat{\boldsymbol{\Lambda}})(\mathbf{f} - \hat{\mathbf{f}}),$$

gdzie $\hat{\boldsymbol{\Lambda}}$ jest macierzą $\boldsymbol{\Lambda}$ wyznaczoną dla $\hat{\mathbf{f}}$. Z powyższego równania bezpośrednio wynika, że:

$$\mathbf{f}|\mathbf{X}, \mathbf{y} \sim \mathcal{N}\left(\hat{\mathbf{f}}, (\boldsymbol{\Sigma}^{-1} + \hat{\boldsymbol{\Lambda}})^{-1}\right). \tag{1.8}$$

Tak więc zostało nam już tylko zoptymalizowanie $\mathbb{P}(\mathbf{y}|\mathbf{X})$ tak, by wyznaczyć najlepszy wektor parametrów $\boldsymbol{\Theta} = [\kappa, \sigma, b_1, \Delta_2, \dots, \Delta_{r-1}]^T$, który przyda nam się przy predykcji. Znów, odwołując się do przybliżenia Laplace'a i do faktu, że:

$$\mathbb{P}(\mathbf{y}|\mathbf{X}) = \int \mathbb{P}(\mathbf{y}|\mathbf{f}, \mathbf{X})\mathbb{P}(\mathbf{f}|\mathbf{X})d\mathbf{f},$$

otrzymujemy:

$$\mathbb{P}(\mathbf{y}|\mathbf{X}) \simeq e^{-S(\hat{\mathbf{f}})}\left|\mathbf{I} + \boldsymbol{\Sigma}\hat{\boldsymbol{\Lambda}}\right|^{-\frac{1}{2}},$$

gdzie $\mathbf{I}$ jest macierzą jednostkową $n \times n$. Bez problemu możemy teraz znaleźć maksimum prawdopodobieństwa $\mathbb{P}(\mathbf{y}|\mathbf{X})$ optymalizacyjnie lub nawet analitycznie (por.[5]).

Wróćmy teraz do szukanych przez nas całek (1.2) i (1.3). Korzystając z równań (1.4) i (1.8) dostaniemy, że w przybliżeniu:

$$f(\mathbf{x}_*)|\mathbf{X}, \mathbf{y} \sim \mathcal{N}(\mu_*, \sigma_*^2),$$

gdzie $\mu_* = \mathbf{\Sigma}^T \mathbf{\Sigma^{-1}} \hat{\mathbf{f}}$ oraz $\sigma_*^2 = \Sigma_{**} - \mathbf{\Sigma}^T (\mathbf{\Sigma} + \hat{\mathbf{\Lambda}}^{\mathbf{-1}})^{-1} \mathbf{\Sigma}$. Natomiast, korzystając jeszcze z równania (1.7), otrzymujemy rozkład predykcyjny następującej postaci:

$$\mathbb{P}(y_*|\mathbf{X}, \mathbf{y}, \mathbf{x}_*) = \Phi\left(\frac{b_{y_*} - \mu_*}{\sqrt{\sigma^2 + \sigma_*^2}}\right) - \Phi\left(\frac{b_{y_*-1} - \mu_*}{\sqrt{\sigma^2 + \sigma_*^2}}\right).$$

Dla nowej obserwacji wystarczy teraz jedynie wyznaczyć $\underset{i}{\operatorname{argmax}} \mathbb{P}(y_* = i|\mathbf{X}, \mathbf{y}, \mathbf{x}_*)$.

Rozdział 2

Diagnostyka modelu

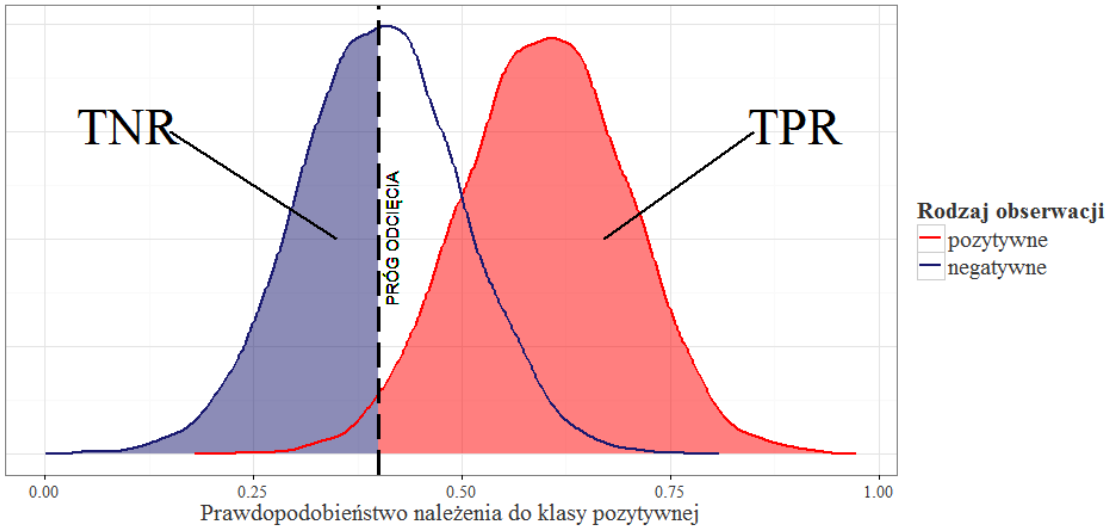
W poprzednim rozdziale poznaliśmy kilka metod rozwiązania problemu regresji porządkowej. Teraz chcielibyśmy dowiedzieć się, która z nich jest najlepsza. Oczywiście nie da się stwierdzić tego w ogólności, gdyż skuteczność metod zależy od konkretnych danych. W przypadku dwuklasowym najczęściej stosowaną metodą oceny jest krzywa ROC (ang. *Receiver Operating Characteristic*) i pole pod tą krzywą, czyli AUC (ang. *Area Under the Curve*). Okazuje się, że można uogólnić ją na nasz przypadek. Przyjrzyjmy się temu dokładniej.

2.1. Krzywa ROC w przypadku dwuklasowym

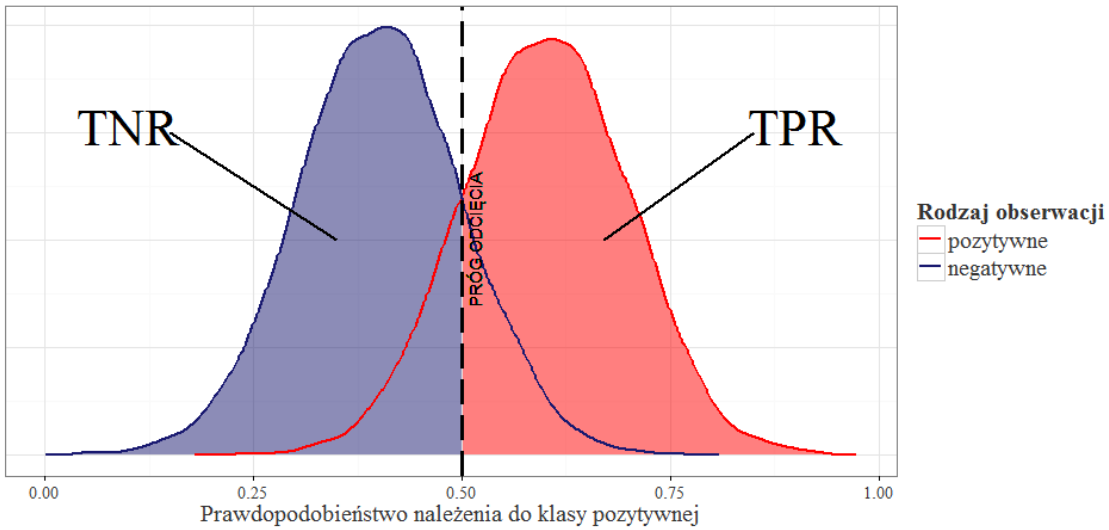
Żeby łatwiej zrozumieć konstrukcję krzywej ROC w przypadku regresji porządkowej, przypomnijmy sobie najpierw, jak to się działo w najprostszej, dwuklasowej sytuacji. Załóżmy, że mamy już dopasowany model, a nasza zmienna odpowiedzi jest binarna z odpowiedzią pozytywną lub negatywną. Na zbiorze testowym możemy wtedy otrzymać tabelę jakości dopasowania (patrz Rys.2.1).

		Prawdziwa klasa +	Prawdziwa klasa −
Wyestymowana przez nas klasa	+	TP	FP
	−	FN	TN

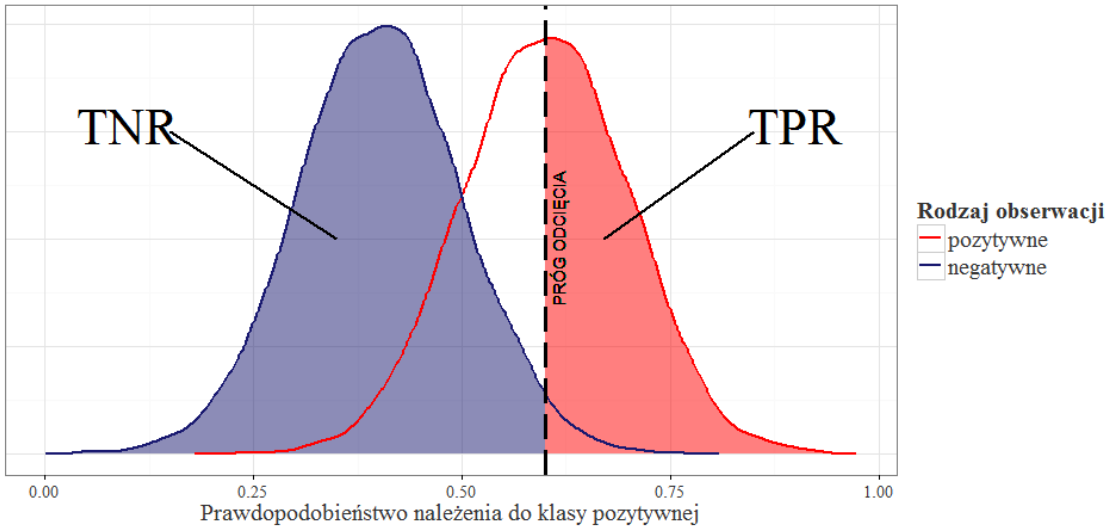
Rysunek 2.1: Tabela jakości dopasowania. TP (ang. *True Positives*) to liczba rekordów z klasy pozytywnej, które zostały zakwalifikowane przez nas jako klasa pozytywna. Analogicznie, TN (ang. *True Negatives*) to liczba rekordów z klasy negatywnej, które zostaly zakwalifikowane przez nas jako klasa negatywna. FP (ang. *False Positives*) oznacza rekordy z klasy negatywnej, zakwalifikowane jako klasa pozytywna i wreszcie, FN (ang. *False Negatives*) to rekordy z klasy pozytywnej, które błędnie zakwalifikowane zostały jako klasa negatywna.



(a) Duże TPR, ale małe TNR.



(b) Zrównoważone TPR i TNR.



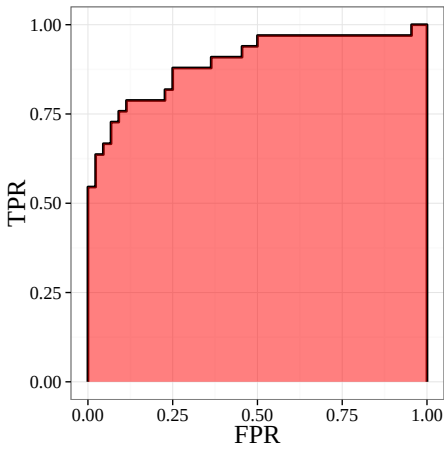
(c) Małe TPR, ale duże TNR.

Rysunek 2.2: Sposób konstruowania krzywej ROC w przypadku dwuklasowym.

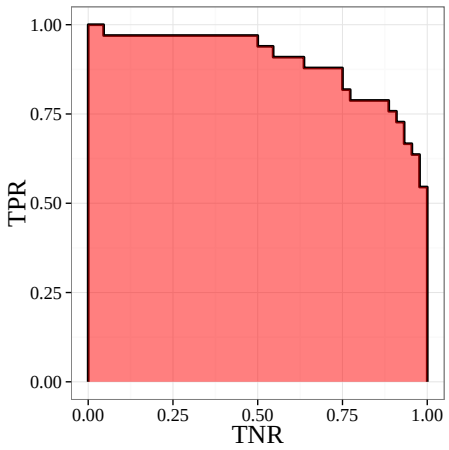
Do stworzenia krzywej ROC, potrzebne nam będą dwa wskaźniki – czułość (TPR, ang. *True Positive Rate*) i specyficzność (TNR, ang. *True Negative Rate*). Czułość definiować będziemy jako prawdopodobieństwo, że pozytywny rekord zostanie poprawnie zakwalifikowany jako pozytywny, a specyficzność jako prawdopodobieństwo, że negatywny rekord zostanie poprawnie zakwalifikowany jako negatywny. Inaczej mówiąc, czułość i specyficzność to procent poprawnie sklasyfikowanych rekordów, odpowiednio w grupie pozytywnej i negatywnej. Korzystając z tabeli jakości dopasowania (patrz Rys. 2.1), otrzymujemy:

$$\begin{aligned} TPR &= \frac{TP}{TP+FN}, \\ TNR &= \frac{TN}{FP+TN}. \end{aligned}$$

Ale jak stworzyć krzywą, mając tylko dwa wskaźniki? Otóż krzywa ROC to wykres punktów (1−TNR, TPR), wyliczonych dla różnych progów odcięcia. Czym jest zatem próg odcięcia? W większości przypadków, model generuje nam nie tylko klasę, do której powinniśmy zaklasyfikować daną obserwację, ale przede wszystkim prawdopodobieństwo, z jakim możemy coś zakwalifikować do klasy pozytywnej. Standardowo przyjmuje się, że to prawdopodobieństwo wynosi $0,5$, ale niekoniecznie musi tak być. Czasem wystarczy nam 40% pewności, żeby coś zaklasyfikować jako pozytywne. Dużo tu zależy od historii, która stoi za naszymi danymi. I właśnie ten procent pewności to liczba, którą będziemy nazywać progiem odcięcia.



(a) Standardowy sposób rysowania krzywej ROC.



(b) Krzywa ROC z TNR (zamiast FPR) na osi OX.

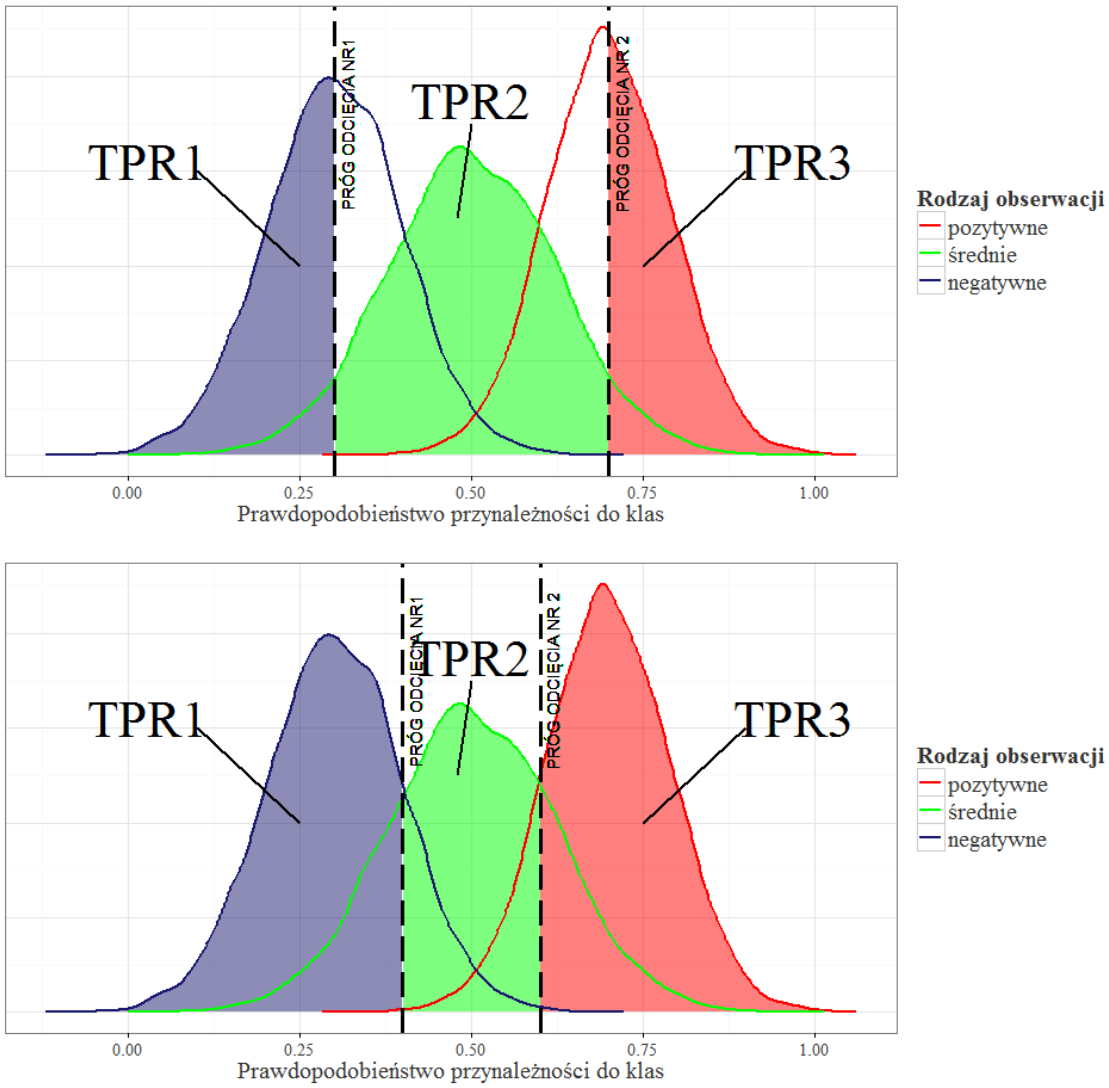
Rysunek 2.3: Przykładowe krzywe ROC.

Przyjrzyjmy się rysunkowi 2.2. Mamy tu wykresy gęstości obserwacji pozytywnych i negatywnych, w zależności od przyjętego progu odcięcia. Większość obserwacji pozytywnych osiąga około 60−procentowe prawdopodobieństwo przynależności do klasy pozytywnej, a negatywnych 40−procentowe prawdopodobieństwo przynależności do klasy negatywnej. Z wykresów wyraźnie widać, że poruszanie tym progiem odcięcia w prawo zwiększy nam czułość, ale zmniejszy specyficzność, natomiast poruszanie w lewo odwrotnie. Patrząc na krzywą ROC, możemy zobaczyć ich zależność od siebie na jednym wykresie i wybrać taki próg, jaki nam najbardziej odpowiada (najczęściej taki, który jest dobrym kompromisem między czułością a specyficznością). Standardowo, krzywą ROC rysuje się nie w zależności od specyficzności, tylko od 1−specyficzności, nazywanej FPR (ang. *False Positive Rate*). My jednak, by łatwiej

było nam uogólnić ją na więcej wymiarów, zastosujemy tę mniej popularną reprezentację, czyli na osi OX będziemy przedstawiać specyficzność (patrz Rys.2.3).

Idealna krzywa to taka, która ma duże TPR i małe FPR, tworzy zatem kwadrat jednostkowy. Zła krzywa, czyli taka, która powstaje, gdy model daje losowe wyniki, to taka, która jest przekątną tego kwadratu. Ponieważ, patrząc na dwie często wielokrotnie przecinające się krzywe ROC, odpowiadające różnym modelom, ciężko jest stwierdzić, która krzywa jest lepsza, wprowadzono współczynnik AUC, czyli pole pod tą krzywą, który pozwala łatwiej to ocenić. Idealny model ma współczynnik AUC równy 1, a model losowy charakteryzuje się AUC równym $0,5$. Na rysunku 2.3 łatwo widać, że w naszym przypadku (czyli z inaczej zdefiniowaną osią OX) współczynnik AUC definiuje się identycznie.

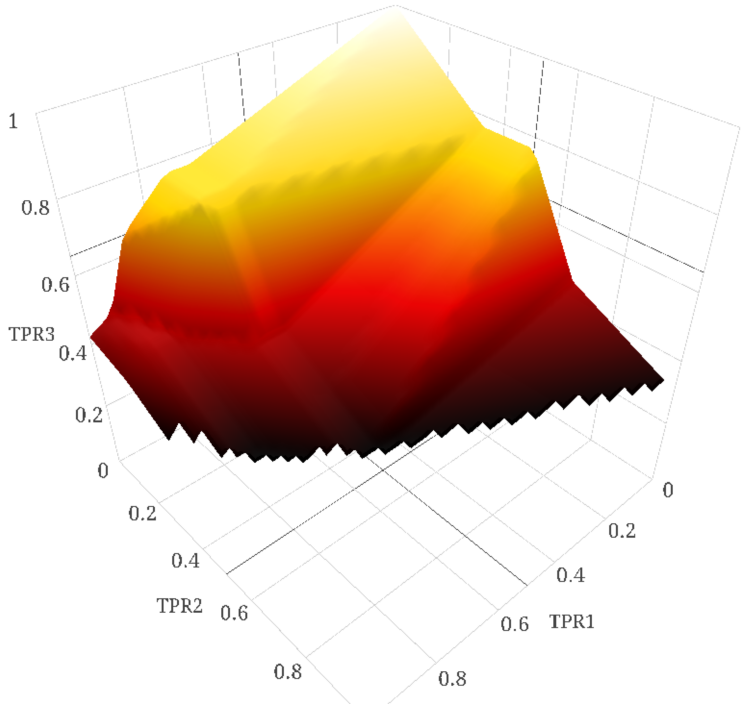
2.2. Krzywa ROC w przypadku regresji porządkowej



Rysunek 2.4: Sposób konstruowania krzywej ROC w przypadku trzyklasowym.

W przypadku regresji porządkowej nie będzie już tak łatwo. Przede wszystkim nie mamy tu podziału na klasę pozytywną i negatywną, jak więc stworzyć współczynnik FPR? Można próbować robić to parami tzn. traktować jedną z klas jako pozytywną, a pozostałe połączyć w jedną i traktować jako negatywną. Robiąc w ten sposób z każdą klasą, otrzymamy r (bo tyle jest możliwych poziomów zmiennej odpowiedzi) krzywych ROC, a tym samym r współczynników AUC. Jako ostateczne AUC przyjmuje się wtedy średnią z nich. Nie jest to jednak dobry wskaźnik. Może się bowiem zdarzyć tak, że współczynnik między środkowymi klasami wyjdzie duży, natomiast ten między klasami skrajnymi słaby, tworząc tym samym nienajgorszą średnią. Nie jest to dobre, gdyż często zależy nam na dobrym odróżnieniu właśnie klas skrajnych. Wyobraźmy sobie, że chcemy sprawdzić, czy komuś spodobałaby się sprzedawana przez nas książka. Możliwe odpowiedzi to: bardzo mi się podoba, podoba mi się, nie mam zdania, nie podoba mi się, bardzo mi się nie podoba. Jasne jest, że wolelibyśmy oddzielić klientów, którym bardzo spodobałaby się książka od tych, którym bardzo by się nie spodobała, a nie na przykład tych, którym by się nie spodobała od tych, którym by się bardzo nie spodobała. Żeby udało nam się poradzić sobie z takim problemem, trzeba spojrzeć na niego globalnie.

Opisując krzywą ROC w przypadku dwuklasowym powiedzieliśmy sobie, że będziemy rozważać nie zależność TPR od FPR, ale TPR od TNR. Dlaczego? Właśnie po to, żebyśmy teraz mogli ją łatwiej uogólnić. Zarówno TPR, jak i TNR jest to procent poprawnie sklasyfikowanych odpowiednio pozytywnych bądź negatywnych obserwacji. Nic nie staje zatem na przeszkodzie, by stworzyć r takich współczynników $(\mathrm{TPR}1, \ldots, \mathrm{TPR}r)$, każdy odpowiadający procentowi poprawnie sklasyfikowanych obserwacji z $i-$tej klasy. Przyjmując różne progi odcięcia (patrz Rys. 2.4), których tym razem będzie $r-1$, możemy narysować krzywą ROC, a raczej pewną hiperpowierzchnię. Oczywiście jest to możliwe tylko w przypadku trzyklasowym (patrz Rys. 2.5), ale rysunek taki i tak jest raczej mało czytelny.



Rysunek 2.5: Krzywa ROC w przypadku trzyklasowym.

2.3. Współczynnik VUS

Po co zatem tworzyć wielowymiarową krzywą ROC, skoro i tak trudno cokolwiek z niej odczytać? Otóż głównie po to, by otrzymać współczynnik AUC, który, będąc konkretną liczbą, jest znacznie prostszy w interpretacji. W przypadku więcej niż dwuwymiarowym będziemy go nazywać VUS (ang. *Volume Under the Surface*).

Jako, że liczenie objętości pod hiperpłaszczyzną jest numerycznie raczej trudnym zadaniem, w celu wyliczenia współczynnika VUS, skorzystamy z jego nieco innej interpretacji niż tylko pole pod krzywą ROC. Wróćmy znów do przypadku dwuklasowego i przyjrzyjmy się wykresowi 2.3b. Na osi OY mamy współczynnik TNR, czyli prawdopodobieństwo, że wyestymujemy klasę negatywną pod warunkiem, że klasa rzeczywiście jest negatywna. Równoważnie, można to zapisać jako prawdopodobieństwo, że prawdopodobieństwo odpowiadające negatywnej obserwacji jest mniejsze niż pewien próg odcięcia. Analogicznie TPR to prawdopodobieństwo, że prawdopodobieństwo odpowiadające pozytywnej obserwacji jest większe niż próg odcięcia. Łącząc oba wyniki otrzymamy, że współczynnik AUC to nic innego tylko prawdopodobieństwo, że losowo wybrana pozytywna obserwacja będzie mieć wyższe prawdopodobieństwo niż losowo wybrana negatywna obserwacja. Innymi słowy, będą one dobrze uporządkowane. Łatwo to już uogólnić na więcej, w naturalny sposób uporządkowanych, wymiarów. Interesować nas będzie pewna estymacja tego prawdopodobieństwa. Łatwo można zauważyć, że będzie nią tzw. statystyka U Manna–Whitney'a–Wilcoxona (por. [9], [10]), czyli wyrażenie:

$$VUS = \frac{1}{n_1 n_2 \cdot \ldots \cdot n_r} \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \cdots \sum_{i_r=1}^{n_r} \mathbb{I}_{\{f(\mathbf{x}_{i_1}^1) < \ldots < f(\mathbf{x}_{i_r}^r)\}},$$

gdzie $\mathbf{x}_i^j$ oznacza $i-$ty wektor cech $\mathbf{x}$ o prawdziwej klasie j, n_i to liczba obserwacji zaklasyfikowanych przez nas jako klasa i, a f to pewna funkcja, która zwraca liczbę rzeczywistą, mającą estymować uporządkowanie obserwacji.

Krzywa ROC i współczynnik VUS jest więc dość prostym, bardzo łatwo interpretowalnym i pomocnym narzędziem do oceny jakości modelu i podejmowania decyzji, który model jest najlepszy. Największą jego wadą wydaje się konieczność znania prawdopodobieństw przynależności do klas (lub po prostu funkcji, która pozwoli nasze obserwacje uporządkować), a nie każda metoda takie prawdopodobieństwa zwraca (np. nie robią tego sieci neuronowe). Trzeba wtedy odwołać się do prostszych metod (takich jak procent poprawności dopasowania lub czułość). Większość modeli oferuje jednak taką możliwość, więc niewątpliwie warto z tego narzędzia diagnostycznego korzystać.

Dodatek A

fdaaggfdadfsd

Literatura

[1] Frank E., Hall M., A simple approach to ordinal classification, *Proceedings of the European Conference on Machine Learning*, Freibourg, Niemcy, 2001, str. 146–156.

[2] Cheng J., Wang Z., Pollastri G., A neural network approach to ordinal regression, *Neutral Networks*, Hong Kong, 2008.

[3] Dobson A. J., An Introduction to Generalized Linear Models, 2nd Edition, 2001

[4] Chu W., Sathiya Keerthi S., *Support Vector Ordinal Regression*

[5] Chu W., Ghahramani Z., *Gaussian Processes for Ordinal Regression*

[6] Ebden M., *Gaussian Processes for Classification: A Quick Introduction*, August 2008.

[7] Ebden M., *Gaussian Processes for Regression: A Quick Introduction*, August 2008.

[8] Rasmussen C., Williams C., *Gaussian Processes for Machine Learning*, 2006.

[9] Waegman W., De Baets B., A survey on ROC-based ordinal regression, w: Fürnkranz J., Hüllermeier E. (Eds.), *Preference Learning*, Springer, 2010, str. 127-154.

[10] Nakas C.T., Yiannoutsos C.T., Ordered Multiple Class Receiver Operating Characteristic (ROC) Analysis, *Encyclopedia of Biopharmaceutical Statistics*, Taylor and Francis, 2006.

Marta Sommer
Nr albumu 237503

Warszawa, 14 lipca 2015

Oświadczenie

Oświadczam, że pracę magisterską pod tytułem „Statystyczne metody regresji porządkowej", której promotorem jest prof. nzw. dr hab. Przemysław Grzegorzewski wykonałam samodzielnie, co poświadczam własnoręcznym podpisem.

.....
Marta Sommer