



POLITECHNIKA WARSZAWSKA  
WYDZIAŁ MATEMATYKI I NAUK INFORMACYJNYCH



PRACA DYPLOMOWA MAGISTERSKA  
NA KIERUNKU MATEMATYKA

**STATYSTYCZNE METODY REGRESJI PORZĄDKOWEJ**

STATISTICAL METHODS FOR ORDINAL REGRESSION

AUTOR:  
MARTA SOMMER

PROMOTOR:  
PROF. NZW. DR HAB.  
PRZEMYSŁAW GRZEGORZEWSKI

WARSZAWA, GRUDZIEŃ 2015

.....  
podpis promotora

.....  
podpis autora

# Spis treści

<b>Streszczenie</b>	<b>5</b>
<b>Abstract</b>	<b>7</b>
<b>Wstęp</b>	<b>9</b>
<b>1. Opis teoretyczny dostępnych metod</b>	<b>11</b>
1.1. Postawienie problemu i podstawowe oznaczenia . . . . .	11
1.2. Model proporcjonalnych szans . . . . .	11
1.3. Wektory maszyn podpierających (SVM) . . . . .	12
1.4. Sieci neuronowe . . . . .	15
1.5. Metoda Franka i Halla . . . . .	17
1.6. Procesy gaussowskie . . . . .	18
<b>2. Diagnostyka modelu</b>	<b>25</b>
2.1. Wskaźniki diagnostyczne . . . . .	25
2.2. Procent poprawnej klasyfikacji (ACC) . . . . .	25
2.3. Średni błąd bezwzględny . . . . .	26
2.4. Krzywa ROC w przypadku dwuklasowym . . . . .	26
2.5. Krzywa ROC w przypadku regresji porządkowej . . . . .	29
2.6. Współczynnik VUS . . . . .	31
<b>3. Porównanie metod modelowania i współczynników diagnostycznych</b>	<b>33</b>
3.1. Opis eksperymentu . . . . .	33
3.2. Wstępna analiza wyników . . . . .	33
3.3. Współczynnik BSC . . . . .	37
3.4. Dalsze wnioski . . . . .	39
<b>Podsumowanie</b>	<b>41</b>
<b>Literatura</b>	<b>43</b>



# Streszczenie

W uczeniu maszynowym często natrafia się na problem klasyfikacji wieloklasowej (ang. *multinomial regression*), w której zmienna odpowiedzi jest w pewien naturalny sposób uporządkowana. Mamy wtedy do czynienia z tzw. regresją porządkową (ang. *ordinal regression*).

Głównym celem mojej pracy jest zebranie, omówienie i porównanie dostępnych w literaturze metod modelowania regresji porządkowej. Opisanych zostało w ten sposób pięć klasyfikatorów: model proporcjonalnych szans, model oparty o procesy gaussowskie, model Franka i Halla, sieci neuronowe oraz wektory maszyn podpierających (SVM).

Oprócz tego praca zawiera opis trzech różnych wskaźników diagnostycznych, które pomagają w ocenie wyżej wymienionych metod klasyfikacji, jak również ocenę tych metod na danych rzeczywistych wraz z porównaniem skuteczności samych współczynników.

## Słowa kluczowe:

regresja porządkowa, SVM, sieci neuronowe, model proporcjonalnych szans, procesy gaussowskie, krzywa ROC, współczynnik AUC, współczynnik VUS



# Abstract

A commonly encountered problem in the field of machine learning is multinomial regression, where the dependent variable is ordered in a natural, non-mathematical progression. This presents the issue of ordinal regression.

The main focus of my thesis is to put together, explain and compare methods of ordinal regression modelling currently available in professional literature. There are five classifiers described: the proportional odds model, Gaussian processes, the Frank-Hall method, neural networks and Support Vector Machines (SVM).

In addition, my thesis describes three different diagnostic coefficients which help to assess these methods of classification, as well as an assessment and comparison of these methods – and the coefficients themselves – using real data.

## Keywords

ordinal regression, SVM, neural networks, proportional odds model, Gaussian processes, ROC curve, AUC, VUS





# Wstęp

Uczenie maszynowe jest bardzo szybko rozwijającą się dziedziną nauki z pogranicza matematyki i informatyki, głównie ze względu na szeroką gamę jej zastosowań. Nawet prosta regresja i klasyfikacja pomagają w odkrywaniu pewnych zależności oraz pozwalają prognozować różne wielkości. Z uczeniem maszynowym bardzo często – choć nie zawsze świadomie – spotykamy się w życiu codziennym, np. w systemach rekomendacyjnych czy wyszukiwarkach internetowych. To właśnie do tych praktycznych zastosowań potrzebne było stworzenie tzw. regresji porządkowej (ang. *ordinal regression*). Zanim poznamy formalną definicję tego pojęcia i dalsze szczegóły na jego temat, prześledźmy poniższy przykład, by bardziej intuicyjnie zrozumieć, czym właściwie jest regresja porządkowa.

Wyobraźmy sobie następującą sytuację: chcielibyśmy przewidzieć, w jakim stopniu potencjalnemu klientowi spodoba się sprzedawany przez nas produkt. Mając taką wiedzę, moglibyśmy przewidywać, co opłaca się mu polecić bądź zareklamować. W celu uproszczenia analizy, skupimy się na następujących możliwych odpowiedziach klienta: *zdecydowanie mi się nie podoba*, *nie podoba mi się*, *nie mam zdania*, *podoba mi się*, *zdecydowanie mi się podoba*. Mamy również do dyspozycji pewne cechy danego klienta; może to być na przykład jego wiek, płeć czy wykształcenie. Nie zawsze jednak potrafimy uzyskać takie dane – szczególnie, gdy nie mamy z nim bezpośredniego kontaktu, na przykład w sytuacji, gdy prowadzimy sklep internetowy. Wtedy za wektor cech możemy przyjąć np. jego historię zakupów na naszej stronie. Jako dane do tworzenia modelu możemy użyć informacje na temat pewnej grupy klientów, których opinię o danym produkcie znamy (gdyż np. zapytaliśmy ich o to wprost lub za pomocą ankiety). Metoda działania powinna być więc następująca: najpierw dopasowujemy pewien model w oparciu o dane historyczne, a następnie, gdy mamy do czynienia z nowym klientem o znanych cechach, model pozwala przewidzieć reakcję tego klienta na proponowany przez nas produkt.

Analizę powyższego problemu można przeprowadzić na kilka różnych sposobów. Najbardziej naturalnym wydawałoby się zastosowanie klasyfikacji wieloklasowej (tzn. takiej, gdzie odpowiedź jest nominalna i ma więcej niż dwa poziomy; ang. *multinomial regression*). W tym procesie tracimy wtedy jednak istotną informację o naturalnym porządku otrzymanych odpowiedzi. Jeśli chcielibyśmy wziąć go pod uwagę, możemy potraktować nasz problem jak zwykłą regresję, zastępując zmienną odpowiedzi pewną zmienną ciągłą (np. *zdecydowanie mi się nie podoba* odpowiadałoby wartości liczbowej 1, a *zdecydowanie mi się podoba* wartości 5) i modelować właśnie tę zmienną ciągłą, a następnie z powrotem zamienić ją na uporządkowaną zmienną nominalną. Pojawia się tu jednak problem optymalnego wykonania takiej transformacji, z uwzględnieniem wielu czynników, m.in. tego, że możliwe odpowiedzi niekoniecznie są od siebie jednakowo odległe (np. różnica między *nie podoba mi się* a *nie mam zdania* wcale nie musi być taka sama, jak między *podoba mi się* a *zdecydowanie mi się podoba*). Podstawowe zagadnienia uczenia maszynowego nie znajdują tu zatem zastosowania i stąd wynikła konieczność rozwinięcia problemu regresji porządkowej.

Powyższy przykład zilustrował kwestię regresji porządkowej. W skrócie można określić ją jako problem klasyfikacji wieloklasowej, w którym zmienna odpowiedzi jest w pewnym naturalnym porządku. Formalne sformułowanie problemu przedstawimy na początku rozdziału pierwszego.

Celem mojej pracy jest omówienie teoretycznych zagadnień oraz praktycznych zastosowań regresji porządkowej. Praca ma zatem następującą strukturę: rozdział pierwszy poświęcony jest zebraniu, opisaniu i usystematyzowaniu dostępnych w literaturze metod tworzenia modeli; pokazane są w nim również wady i zalety poszczególnych z nich oraz dzielące je różnice. W rozdziale drugim w podobny sposób opisane są metody diagnostyki modelu. W ostatnim rozdziale, który – w przeciwieństwie do pierwszych dwóch, w dużej mierze teoretycznych – opiera się na danych rzeczywistych, zawiera porównanie metod modelowania regresji porządkowej oraz ocenę jakości współczynników diagnostycznych.

# Rozdział 1

## Opis teoretyczny dostępnych metod

### 1.1. Postawienie problemu i podstawowe oznaczenia

Mamy dany 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 symbolem kategorii, do której przyporządkowana została  $i$ -ta obserwacja. Najczęściej zakładamy, że  $y^{(i)} \in \mathcal{Y}$  oraz  $\mathcal{Y} = \{1, \dots, r\}$  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 uporać się z tym problemem.

### 1.2. Model proporcjonalnych szans

Najbardziej rozpowszechnionym sposobem modelowania regresji porządkowej jest model proporcjonalnych szans (ang. *proportional odds model*). Informacje na jego temat znaleźć można na przykład tu: [8]. Jest to jedna z metod uogólnionych modeli liniowych, ściśle związana z regresją logistyczną. Interesują nas następujące prawdopodobieństwa:

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

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

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

Następnie rozważa się poniższy model logitowy:

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

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

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

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

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 tę klasę  $y_*$ , która odpowiada maksymalnemu prawdopodobieństwu  $\Pi_j(\mathbf{x}_*)$ .

### 1.3. Wektory maszyn podpierających (SVM)

Wektory maszyn podpierających (ang. *Support Vector Machine*) to bardzo znana i powszechnie stosowana metoda klasyfikacji (patrz np. [12]). W dużym uproszczeniu polega ona na konstrukcji dwóch równoległych i maksymalnie oddalonych od siebie hiperpłaszczyzn rozdzielających klasy. By metoda działała dla przypadków, w których brak jest liniowej separowalności, wprowadza się dodatkowo karę za nieidealne rozdzielenie klas. W przypadku dwuklasowym (patrz rys. 1.1) budowa modelu sprowadza się do rozwiązania następującego problemu optymalizacyjnego:

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

przy następujących ograniczeniach:

$$\begin{cases} \mathbf{x}_{1i}^T \mathbf{w} + b \geq +1 - \xi_i, & \text{dla } i = 1 \dots n_1 \\ \mathbf{x}_{2i}^T \mathbf{w} + b \leq -1 + \xi_i, & \text{dla } i = 1 \dots n_2 \end{cases},$$

gdzie  $\mathbf{w} \in \mathbb{R}^K$ ,  $b \in \mathbb{R}$  i  $C \in \mathbb{R}$  są parametrami modelu,  $\xi_i \geq 0$  dla  $i = 1 \dots n$  są karą mierzoną dla każdej obserwacji przy ustalonej hiperpłaszczyźnie,  $\mathbf{x}_{1i}$  oznacza wektor cech obserwacji należących do klasy pierwszej, a  $\mathbf{x}_{2i}$  wektor cech obserwacji należących do klasy drugiej, zaś  $n_1$  i  $n_2$  to liczności tych klas.

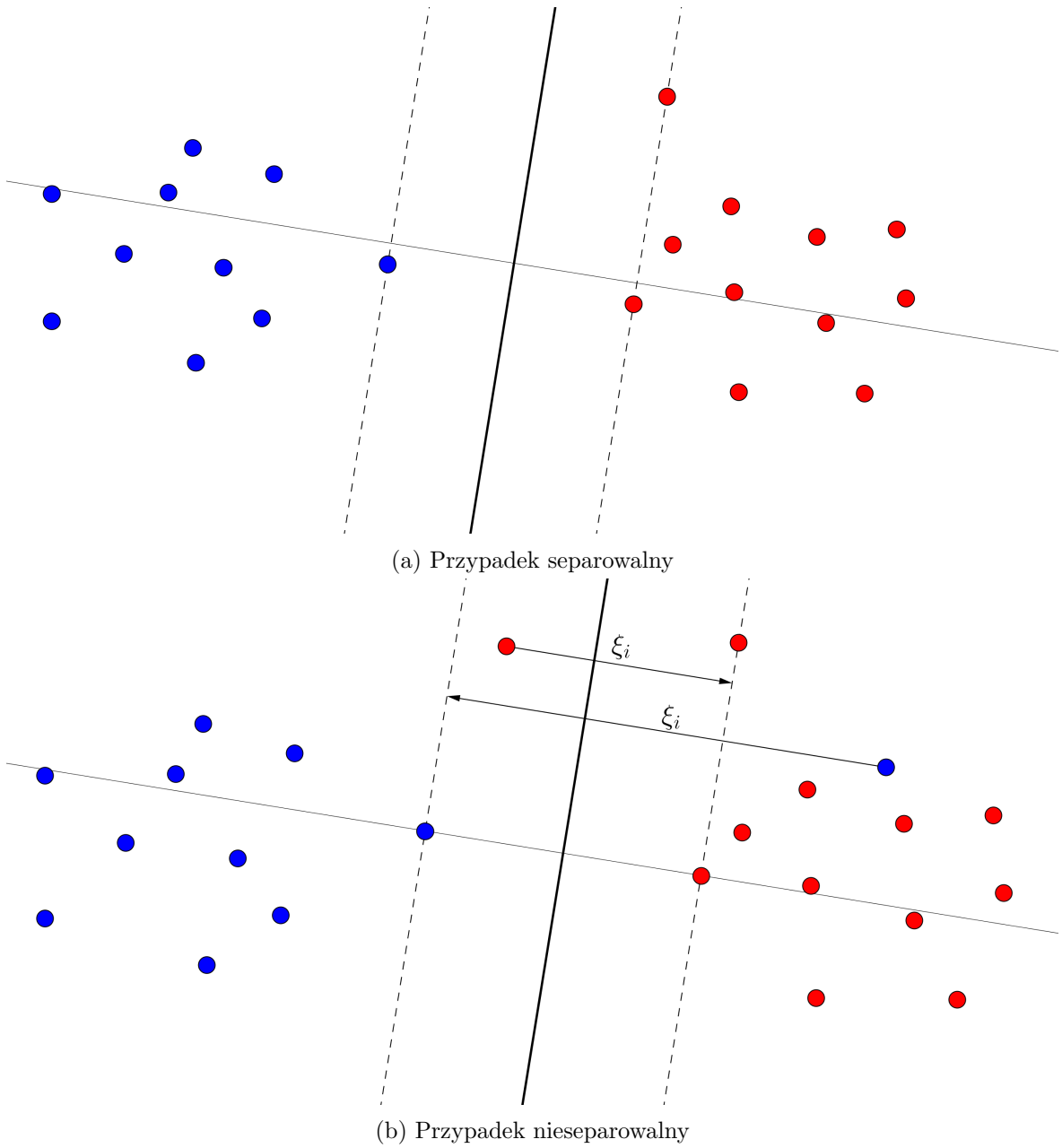
Powyżej krótko przypomniana została metoda SVM w przypadku dwuklasowym. Przyjrzyjmy się teraz, jak w łatwy sposób można dostosować powyższą metodę do rozważanej przez nas regresji porządkowej.

Tym razem będziemy rozwiązywać następujący problem optymalizacyjny (patrz [8] i [5]):

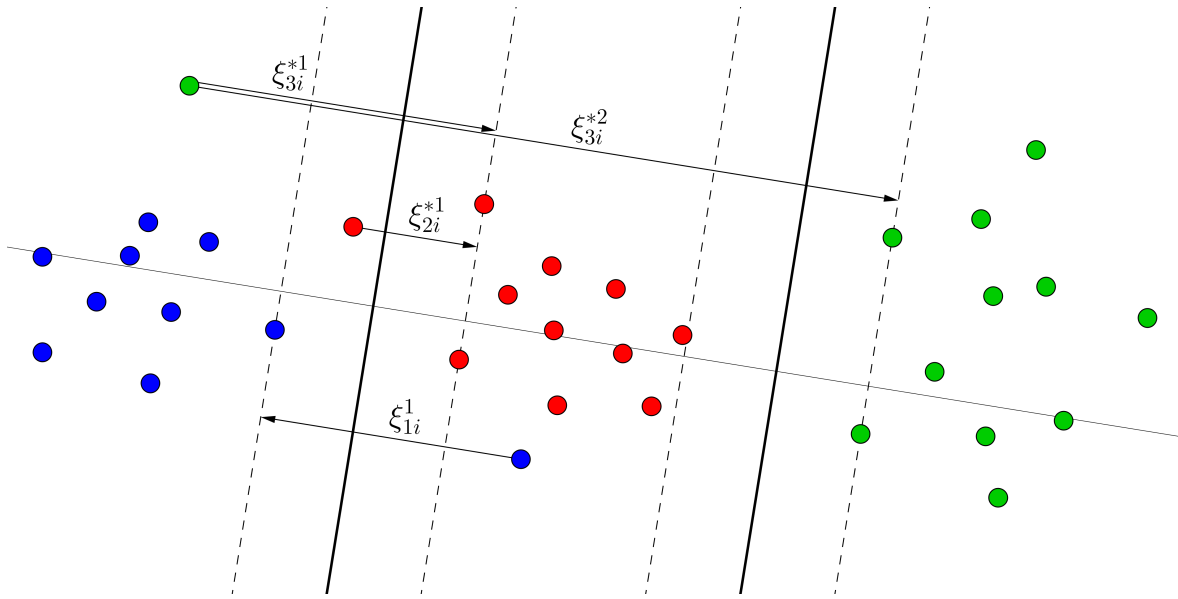
$$\min_{\mathbf{w}, b_1, \dots, b_{r-1}} \left\{ \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{j=1}^{r-1} \left( \sum_{k=1}^j \sum_{i=1}^{n_k} \xi_{ki}^j + \sum_{k=j+1}^r \sum_{i=1}^{n_k} \xi_{ki}^{*j} \right) \right\},$$

przy następujących ograniczeniach:

$$\begin{cases} \mathbf{x}_{ki}^T \mathbf{w} - b_j \leq -1 + \xi_{ki}^j, & \text{dla } k = 1, \dots, j \text{ oraz } i = 1, \dots, n_k \\ \mathbf{x}_{ki}^T \mathbf{w} - b_j \geq +1 - \xi_{ki}^{*j}, & \text{dla } k = j+1, \dots, r \text{ oraz } i = 1, \dots, n_k, \end{cases}$$



Rysunek 1.1: Przykładowe rozdzielanie klas metodą SVM. W przypadku nieseparowalnym stosujemy dodatkowo karę będącą odległością źle zaklasyfikowanej obserwacji od odpowiedniego marginesu



Rysunek 1.2: Przykładowa klasyfikacja metodą SVM w przypadku trzyklasowym

gdzie  $\mathbf{w} \in \mathbb{R}^K$ ,  $b_1 \in \mathbb{R}, \dots, b_{r-1} \in \mathbb{R}, C \in \mathbb{R}$  są parametrami modelu,  $\mathbf{x}_{ki}^T$  oznacza  $i$ -tą obserwację należącą do  $k$ -tej klasy,  $n_k$  to liczność  $k$ -tej klasy,  $j = 1 \dots r - 1$ , a  $\xi$  to kary, których konstrukcję wyjaśnimy poniżej.

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 w dość prosty sposób (patrz rys. 1.2). Dla ustalonego progu  $b_j$  obserwujemy wartości funkcji  $\mathbf{x}_{ki}^T \mathbf{w}$ . Dla obserwacji z niższych klas (tzn. klas  $1, \dots, j$ ), wartości te powinny być niższe niż dolna granica  $b_j - 1$ . Jeśli tak nie jest, wtedy za błąd obserwacji dla progu  $b_j$  uznaje się  $\xi_{ki}^j$ , czyli odległość tego punktu od rozpatrywanej dolnej granicy. Analogicznie dla obserwacji z wyższych klas wartości  $\mathbf{x}_{ki}^T \mathbf{w}$  powinny być wyższe niż górna granica  $b_j + 1$ . Jeśli tak nie jest, to otrzymujemy błędy  $\xi_{ki}^{*j}$ .

Budowa modelu i tym razem sprowadza się więc do problemu optymalizacyjnego. Wyznaczywszy – przy użyciu dowolnego algorytmu znajdującego minimum – szukane parametry, otrzymujemy równania  $r - 1$  hiperpłaszczyzn:

$$\begin{cases} \mathbf{x}^T \mathbf{w} - b_1 & = 0 \\ & \vdots \\ \mathbf{x}^T \mathbf{w} - b_{r-1} & = 0 \end{cases}.$$

Dla nowej obserwacji  $\mathbf{x}_*$  wystarczy wtedy policzyć  $\mathbf{x}_*^T \mathbf{w}$ , sprawdzić, między którymi dwoma hiperpłaszczyznami się znajduje, wreszcie przypisać jej odpowiednią klasę.

## 1.4. Sieci neuronowe

Sieci neuronowe to narzędzie bardzo proste i powszechnie stosowane zarówno w problemach regresji, jak i klasyfikacji. Znalazły one również zastosowanie w regresji porządkowej (patrz [2]).

W dotychczasowych metodach jako dane wejściowe otrzymywaliśmy zbiór uczący w postaci  $n$  par  $(\mathbf{x}, y)$ , gdzie  $\mathbf{x} = (x_1, \dots, x_K)^T$  był wektorem cech, a  $y$  numerem klasy. Tym razem jednak dodatkowo zmodyfikujemy zmienną odpowiedzi w taki sposób, by zamiast liczby rzeczywistej otrzymać zerojedynekowy wektor odpowiedzi  $\mathbf{y} = (y_1, \dots, y_r)^T$  reprezentujący klasę, do której należy dana obserwacja, tzn.  $y_i = \mathbb{I}\{y = i\}$ .

W przeciwieństwie do sieci neuronowych wykorzystywanych przy zwykłej klasyfikacji, nasza sieć neuronowa zakładać będzie porządek zmiennej odpowiedzi. Jako wektor wyjściowy zamiast wektora  $\mathbf{y} = (\underbrace{0, \dots, 0}_{i-1}, 1, \dots, 0)^T$ , mającego wartość  $i$ -tej współrzędnej równą 1, je-

śli obserwacja należała do  $i$ -tej klasy, rozważać będziemy wektor  $\mathbf{y} = (\underbrace{1, \dots, 1}_i, 0, \dots, 0)^T$ , mający wartość 1 współrzędnych o numerach od 1 do  $i$ .

Otrzymamy 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 przyjmujemy funkcję sigmoidalną  $f(x) = \frac{1}{1+e^{-x}}$ , dobrze reprezentującą przynależność do danej klasy jako prawdopodobieństwo.

Uczenie sieci neuronowej będzie odbywało się za pomocą algorytmu propagacji wstecznej (patrz algorytm 1) z kwadratową funkcją straty (można też wykorzystać inną, np. entropię).

---

### Algorytm 1 Algorytm propagacji wstecznej

---

1. Wybieramy małe wagi początkowe oraz pewnie 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).
- 

Aby dokładnie zrozumieć zasadę działania algorytmu 1 w przypadku regresji porządkowej, poniżej przeanalizujemy najważniejsze jego kroki po kolei. Dodatkowo pomocne może okazać się jednocześnie śledzenie rysunku 1.3.

#### Ad. 3)

Dla każdego neuronu obliczamy wartość wejściową ze wzoru:

$$wejście_j^{(i)} = \sum_{k: \exists w_{k \rightarrow j}^{(i)}} \left( w_{k \rightarrow j}^{(i)} \cdot wyjście_k^{(i-1)} \right),$$

gdzie  $wyjście_i^{(j)}$  to wartość  $i$ -tego neuronu w  $j$ -tej warstwie, a  $wyjście_i^{(0)} = x_i$ , zaś  $w_{k \rightarrow j}^{(i)}$  oznacza wagę między  $k$ -tym neuronem  $(i-1)$ -warstwy i  $j$ -tym neuronem  $i$ -tej warstwy. Następnie wyznaczamy wartość wyjściową:

$$wyjście_j^{(i)} = f\left(wyjście_j^{(i)}\right),$$

gdzie  $f(\cdot)$  to wybrana przez nas funkcja sigmoidalna – w naszym przypadku  $f(x) = \frac{1}{1+e^{-x}}$ .

Ad. 4)

Dla warstwy wyjściowej błąd ma postać:

$$\delta_j = wyjście_j^{(i)} \cdot \left(1 - wyjście_j^{(i)}\right) \cdot (wyjście_j^{(i)} - y_j),$$

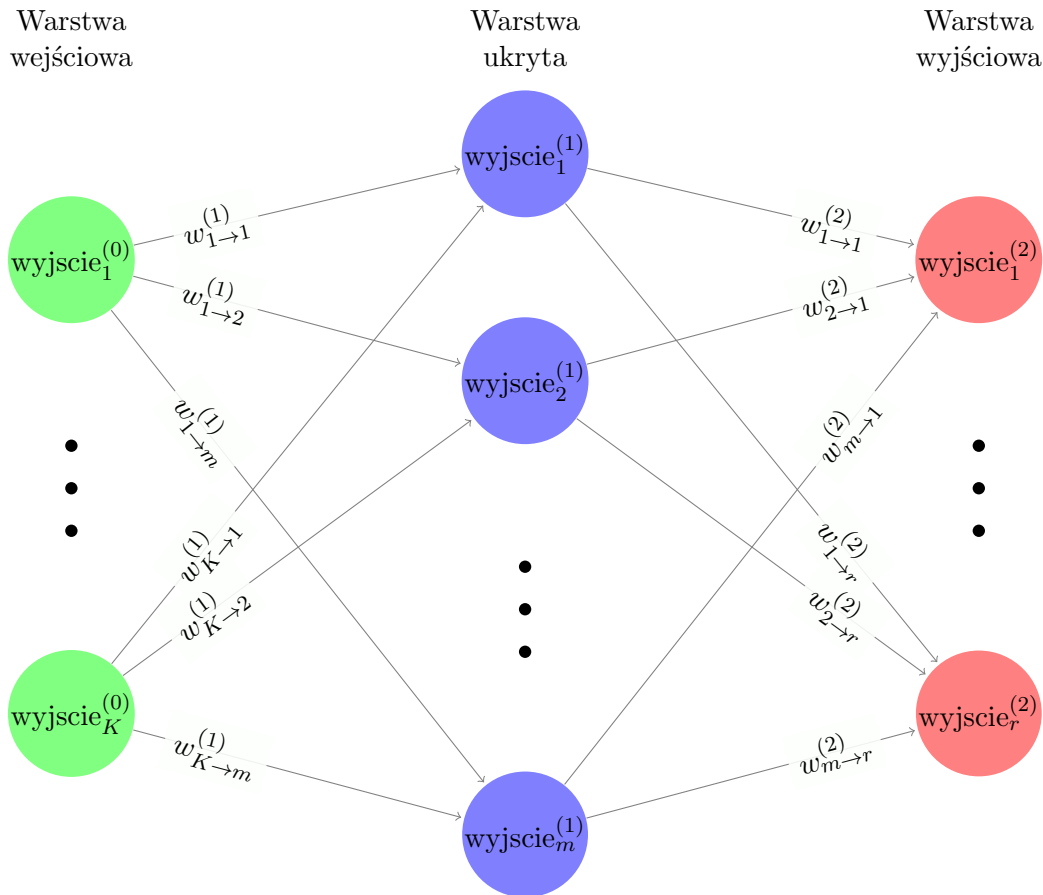
zaś dla warstw ukrytych:

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

Ad. 5)

Modyfikacja wag przebiega następująco:

$$w_{k \rightarrow j}^{(i)} := w_{k \rightarrow j}^{(i)} - \eta \cdot \delta_j^{(i)} \cdot wyjście_k^{(i-1)}.$$



Rysunek 1.3: Przykładowa sieć neuronowa



Predykcja polega już tylko na przejściu algorytmu w przód z nowymi obserwacjami wejściowymi  $\mathbf{x}_*$  i ustaleniu progu (najczęściej równego 0,5, gdyż wartość neuronu wyjściowego reprezentuje pewne prawdopodobieństwo), klasyfikującego neuron wyjściowy jako jedynek. Następnie 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. Może się zdarzyć, że wyjściowy wektor nie będzie ciągiem malejącym, tzn. zamiast łatwo interpretowalnego wektora  $(1, \dots, 1, 0, \dots, 0)$  otrzymamy na przykład wektor  $(1, 1, 0, 1, 1, 1, 0, \dots, 0)$ , co trochę przeczy intuicji, bo sugeruje, że obserwacja należy do klasy czwartej, piątej i szóstej, ale do trzeciej już nie. W takim wypadku – tak jak zostało to opisane powyżej – zaklasyfikowalibyśmy tę obserwację do klasy drugiej, przymykając niejako oko na to, co dzieje się później.

## 1.5. Metoda Franka i Halla

Podejście zaproponowane przez E. Franka i M. Halla (patrz [11]) do zagadnienia regresji porządkowej jest nieco inne, niż w metodach przedstawionych do tej pory metody. Polega bowiem nie na stworzeniu nowego modelu, ale na odpowiednim przedefiniowaniu zbioru danych, a następnie na sprowadzeniu zadania do problemu zwykłej klasyfikacji z dwiema klasami. Dokładniej, przekształcamy  $r$ -klasowy model regresji porządkowej do  $(r - 1)$  dwuklasowych problemów klasyfikacji (patrz algorytm 2).

---

**Algorytm 2** Uproszczony algorytm budowy modelu zaproponowanego przez Franka i Halla

---

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

Przyjrzyjmy się teraz kolejnym krokom algorytmu 2 dokładnie.

### Ad. 1)

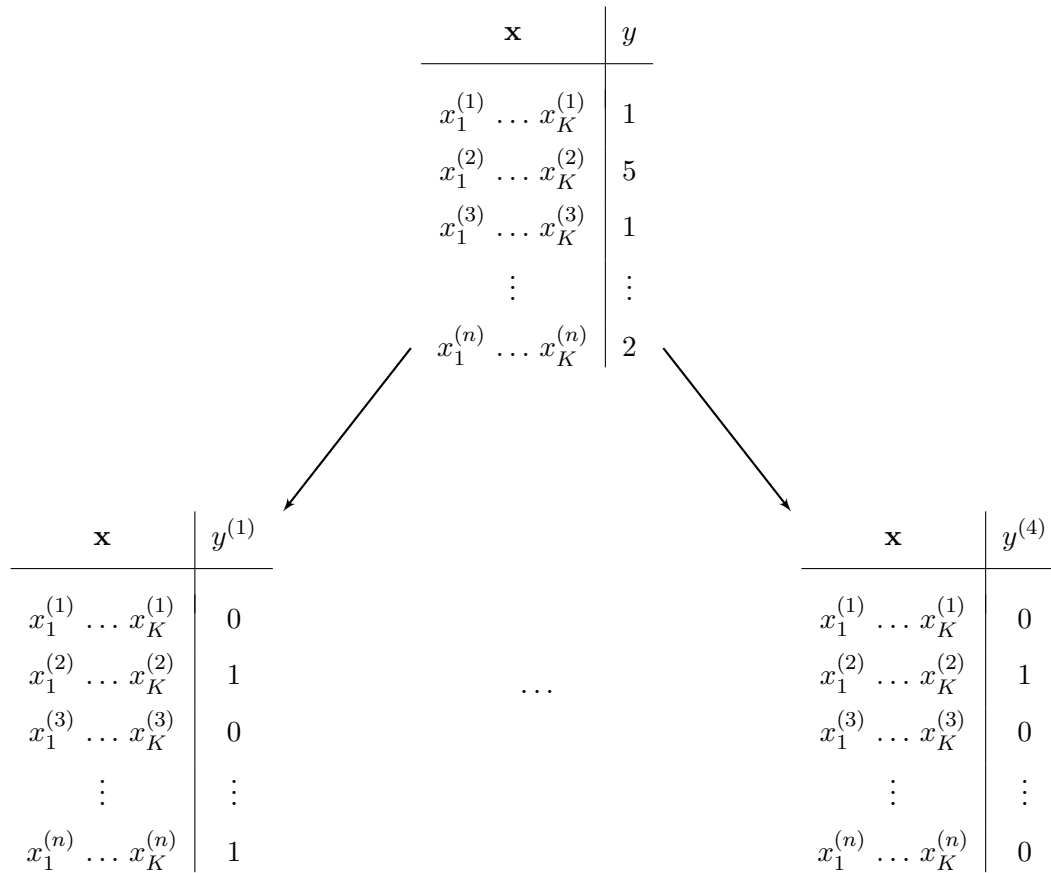
Chcemy otrzymać  $r - 1$  nowych zbiorów o zerojedynekowej zmiennej odpowiedzi. W jaki sposób to zrobić? Macierz atrybutów pozostaje bez zmian, a zmienia się jedynie wektor zmiennej odpowiedzi (patrz rys. 1.4) według zasady:

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

gdzie  $y_i^{(j)}$  to  $i$ -ta odpowiedź w  $j$ -tym, utworzonym przez nas, zbiorze oraz  $i = 1, \dots, n$ , zaś  $j = 1, \dots, r - 1$ .

### Ad. 3)

Dla nowego wektora atrybutów  $\mathbf{x}$  robimy predykcję na  $r - 1$  modelach uzyskanych w punkcie drugim. Zwracamy jednak nie predykcję klasy, ale prawdopodobieństwo przynależności



Rysunek 1.4: Modyfikacja przykładowego zbioru uczącego w metodzie Franka i Halla

do klasy oznaczonej przez nas jako pierwszej. Uzyskujemy w ten sposób  $r - 1$  następujących prawdopodobieństw:  $\mathbb{P}(y > 1), \dots, \mathbb{P}(y > r - 1)$ .

Nas natomiast interesują prawdopodobieństwa:  $\mathbb{P}(y = 1), \dots, \mathbb{P}(y = r - 1)$ , które łatwo otrzymamy, korzystając z następującego wzoru łańcuchowego:

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

Ostatecznie nowej obserwacji przypisujemy klasę będącą wynikiem  $\arg\max_i \{\mathbb{P}(y = i)\}$ .

## 1.6. Procesy gaussowskie

Kolejną metodą modelowania problemu regresji porządkowej jest użycie procesu gaussowskiego. Jest to metoda popularna szczególnie przy zwykłej regresji, znalazła ona jednak również

zastosowanie w klasyfikacji zarówno jedno, jak i wieloklasowej. Chu i Ghahramani w pracy [4] 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. Mianowicie, zamiast dawać od razu odpowiedź, do której klasy przypisujemy daną obserwację, próbujemy ją najpierw scharakteryzować jako liczbę rzeczywistą, by móc ją niejako umieścić na prostej. Dzięki temu uzyskamy pewne uporządkowanie między naszymi obserwacjami, by następnie wybrać progi, które będą już klasyfikować obserwacje. W celu wyrobienia sobie intuicji przeanalizujemy całe rozumowanie „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  $\Sigma$ . Dla celów tej pracy przyjmujemy, że elementy macierzy kowariancji definiowane są w następujący sposób:

$$\Sigma_{ij} = \Sigma(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \exp \left\{ -\frac{\kappa}{2} \sum_{\xi=1}^K (x_{\xi}^{(i)} - x_{\xi}^{(j)})^2 \right\},$$

gdzie  $\kappa > 0$ , a  $x_{\xi}^{(i)}$  oznacza  $\xi$ -ty element wektora  $\mathbf{x}^{(i)}$ . Zatem  $F|\mathbf{X} \sim \mathcal{N}(0, \Sigma)$ , czyli:

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

gdzie  $\mathbf{f} = [f(\mathbf{x}^{(1)}), \dots, f(\mathbf{x}^{(n)})]^T$  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  $y_*$ .

Wyrobiliśmy już sobie pewną intuicję, spróbujmy więc teraz prześledzić wszystko krok po kroku. Interesuje nas wyznaczenie  $y_*$ , dla którego prawdopodobieństwo  $\mathbb{P}(y_*|\mathbf{X}, \mathbf{y}, x_*)$  jest największe. Za pomocą zmiennej ukrytej rozpiszmy je w następujący sposób:

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

Analogicznie:

$$\mathbb{P}(f(\mathbf{x}_*)|\mathbf{X}, \mathbf{y}) = \int_{\mathbb{R}^n} \mathbb{P}(f(\mathbf{x}_*)|\mathbf{f}) \mathbb{P}(\mathbf{f}|\mathbf{X}, \mathbf{y}) d\mathbf{f}. \quad (1.3)$$

By móc policzyć powyższe całki, poszukamy kolejno prawdopodobieństw, których iloczynny je tworzą.

Korzystając z informacji, że zmienna  $F$  jest procesem gaussowskim, czyli:

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

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

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

Zajmijmy się teraz prawdopodobieństwem  $\mathbb{P}(\mathbf{f}|\mathbf{X}, \mathbf{y})$ . Korzystając z podejścia bayesowskiego oraz faktu, że  $\mathbf{f}$  jest zależne od  $\mathbf{X}$ , możemy rozpisać je jako:

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

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. wiarygodność. Ponieważ wszystkie obserwacje są niezależne, otrzymujemy:

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

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  $i = 1, \dots, r-1$  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  $j = 2, \dots, r-1$ . Bardzo rzadko mamy jednak do czynienia z sytuacją idealną, dlatego będziemy budować model, zakładając dodatkowy szum  $\delta$  o rozkładzie  $\mathcal{N}(0, \sigma^2)$ . Wtedy prawdopodobieństwo zmienia się następująco:

$$\begin{aligned} \mathbb{P}(y^{(i)}|f(\mathbf{x}^{(i)})) &= \int_{\mathbb{R}} \mathbb{P}_{ideal}(y^{(i)}, \delta_i|f(\mathbf{x}^{(i)})) d\delta_i = \int_{\mathbb{R}} \mathbb{P}_{ideal}(y^{(i)}|f(\mathbf{x}^{(i)}), \delta_i) \mathbb{P}(\delta_i) d\delta_i = \\ &= \int_{\mathbb{R}} \mathbb{P}(\delta_i) \mathbb{I}_{\{f(\mathbf{x}^{(i)}) + \delta_i \in (b_{y^{(i)}}-1, b_{y^{(i)}}]\}} d\delta_i = \\ &= \Phi\left(\frac{b_{y^{(i)}} - f(\mathbf{x}^{(i)})}{\sigma}\right) - \Phi\left(\frac{b_{y^{(i)}}-1 - f(\mathbf{x}^{(i)})}{\sigma}\right), \end{aligned} \quad (1.7)$$

gdzie  $\Phi(\cdot)$  to dystrybucja 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:

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

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

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

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. 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 rozkładu  $\hat{\mathbf{f}}$  – wygodne byłoby dla nas, gdyby ten rozkład okazał się gaussowski. Da się to osiągnąć dzięki przybliżeniu Laplace'a.

Na początku zauważmy, że:

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

gdzie

$$\mathbf{\Lambda} = \begin{bmatrix} \frac{\partial^2 l(y^{(1)}, f(\mathbf{x}^{(1)}))}{\partial^2 f(\mathbf{x}^{(1)})} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{\partial^2 l(y^{(n)}, f(\mathbf{x}^{(n)}))}{\partial^2 f(\mathbf{x}^{(n)})} \end{bmatrix}.$$

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

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

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

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

Tak więc zostało nam już tylko zoptymalizowanie  $\mathbb{P}(\mathbf{y}|\mathbf{X})$  tak, by wyznaczyć najlepszy wektor parametrów  $\mathbf{\Theta} = [\kappa, \sigma, b_1, \Delta_2, \dots, \Delta_{r-1}]^T$ , 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

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

gdzie  $\mathbf{I}$  jest macierzą jednostkową  $n \times n$ . Bez problemu możemy teraz znaleźć maksimum prawdopodobieństwa  $\mathbb{P}(\mathbf{y}|\mathbf{X})$  iteracyjnie lub nawet analitycznie (patrz [4]).

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

$$\begin{aligned} \mathbb{P}(y_*|\mathbf{X}, \mathbf{y}, \mathbf{x}_*) &= \int_{\mathbb{R}} \mathbb{P}(y_*|f(\mathbf{x}_*))\mathbb{P}(f(\mathbf{x}_*)|\mathbf{X}, \mathbf{y})df(\mathbf{x}_*) = \\ &= \int_{\mathbb{R}} \left[ \Phi\left(\frac{b_{y_*} - f(x_*)}{\sigma}\right) - \Phi\left(\frac{b_{y_*-1} - f(x_*)}{\sigma}\right) \right] \cdot \frac{1}{\sqrt{2\pi}\sigma_*} e^{-\frac{(x-\mu_*)^2}{2\sigma_*^2}} dx = \\ &= \int_{\mathbb{R}} \Phi\left(\frac{b_{y_*} - f(x_*)}{\sigma}\right) \cdot \frac{1}{\sqrt{2\pi}\sigma_*} e^{-\frac{(x-\mu_*)^2}{2\sigma_*^2}} dx - \\ &- \int_{\mathbb{R}} \Phi\left(\frac{b_{y_*-1} - f(x_*)}{\sigma}\right) \cdot \frac{1}{\sqrt{2\pi}\sigma_*} e^{-\frac{(x-\mu_*)^2}{2\sigma_*^2}} dx. \end{aligned} \quad (1.9)$$

Musimy policzyć zatem dwie bardzo podobne całki i odjąć je od siebie. Żeby uprościć obliczenia, policzymy najpierw te całki w ogólności, żeby następnie podstawić tam konkretne symbole. Naszym problemem jest zatem policzenie całki:

$$I = \int_{\mathbb{R}} \Phi\left(\frac{x-m}{\nu}\right) \cdot \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx.$$

Zacznijmy od zwykłego rozpisania podstawowych symboli.

$$I = \int_{\mathbb{R}} \int_{-\infty}^{\frac{x-m}{\nu}} \frac{1}{\sqrt{2\pi}} e^{-\frac{y^2}{2}} dy \cdot \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx = \frac{1}{2\pi\sigma} \int_{\mathbb{R}} \int_{-\infty}^{\frac{x-m}{\nu}} e^{-\frac{y^2}{2} - \frac{(x-\mu)^2}{2\sigma^2}} dy dx$$

Następnie zrobmy po kolei trzy podstawienia  $u = \nu y + m$ ,  $z = (u-m) - (x-\mu)$  oraz  $w = x - \mu$  i zamieńmy kolejność całkowania.

$$\begin{aligned} I &= \frac{1}{2\pi\sigma\nu} \int_{\mathbb{R}} \int_{-\infty}^x e^{-\frac{(u-m)^2}{2\nu^2} - \frac{(x-\mu)^2}{2\sigma^2}} du dx = \frac{1}{2\pi\sigma\nu} \int_{\mathbb{R}} \int_{-\infty}^{\mu-m} e^{-\frac{(z+(x-\mu))^2}{2\nu^2} - \frac{(x-\mu)^2}{2\sigma^2}} dz dx = \\ &= \frac{1}{2\pi\sigma\nu} \int_{-\infty}^{\mu-m} \int_{\mathbb{R}} e^{-\frac{(z+w)^2}{2\nu^2} - \frac{w^2}{2\sigma^2}} dw dz \end{aligned}$$

Zajmijmy się na razie tylko środkową całką. Po sprowadzeniu do wspólnego mianownika i prostych przekształceniach, otrzymamy:

$$\begin{aligned} A &= \int_{\mathbb{R}} e^{-\frac{(z+w)^2}{2\nu^2} - \frac{w^2}{2\sigma^2}} dw = \int_{\mathbb{R}} e^{-\frac{\left(w\sqrt{\sigma^2+\nu^2} + z\frac{\sigma^2}{\sqrt{\sigma^2+\nu^2}}\right)^2}{2\nu^2\mu^2} - \frac{z^2}{2(\sigma^2+\nu^2)}} dw = \\ &= e^{-\frac{z^2}{2(\sigma^2+\nu^2)}} \int_{\mathbb{R}} e^{-\frac{\left(w\sqrt{\sigma^2+\nu^2} + z\frac{\sigma^2}{\sqrt{\sigma^2+\nu^2}}\right)^2}{2\nu^2\mu^2}} dw. \end{aligned}$$

Robiąc podstawienie  $u = \frac{w\sqrt{\sigma^2+\nu^2} + z\frac{\sigma^2}{\sqrt{\sigma^2+\nu^2}}}{\nu\sigma}$  oraz korzystając z faktu, że gęstość rozkładu prawdopodobieństwa całkuje się do jedynki, otrzymamy:

$$A = e^{-\frac{z^2}{2}} \cdot \sqrt{2\pi} \cdot \underbrace{\frac{1}{\sqrt{2\pi}} \int_{\mathbb{R}} e^{-\frac{u^2}{2}} du}_{=1} \cdot \frac{\nu\sigma\sqrt{\sigma^2+\nu^2}}{\sigma^2+\nu^2} = \sqrt{2\pi} \frac{\nu\sigma}{\sqrt{\sigma^2+\nu^2}} e^{-\frac{z^2}{2(\sigma^2+\nu^2)}}.$$

Wróćmy teraz do szukanej całki.

$$I = \frac{1}{2\pi\sigma\nu} \int_{-\infty}^{\mu-m} A \, dz = \frac{1}{2\pi\sigma\nu} \sqrt{2\pi} \frac{\nu\sigma}{\sqrt{\sigma^2+\nu^2}} \int_{-\infty}^{\mu-m} e^{-\frac{z^2}{2(\sigma^2+\nu^2)}} dz.$$

Robiąc podstawienie  $x = \frac{z}{\sqrt{\sigma^2+\nu^2}}$ , otrzymamy wreszcie:

$$I = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\frac{\mu-m}{\sqrt{\sigma^2+\nu^2}}} e^{-\frac{x^2}{2}} dx = \Phi\left(\frac{\mu-m}{\sqrt{\sigma^2+\nu^2}}\right). \quad (1.10)$$

Wróćmy teraz do równania (1.9). Korzystając z (1.10), dostaniemy:

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

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





## Rozdział 2

# Diagnostyka modelu

### 2.1. Wskaźniki diagnostyczne

W poprzednim rozdziale poznaliśmy kilka metod radzenia sobie z problemem regresji porządkowej. Teraz chcielibyśmy dowiedzieć się, która z nich jest najlepsza. Oczywiście nie da się stwierdzić tego w jednoznaczny sposób, gdyż skuteczność metod zależy od konkretnych danych. Mamy jednak do dyspozycji kilka wskaźników, które pomogą nam w ocenie jakości modelu. Z podobnymi współczynnikami spotkaliśmy się już przy okazji zwykłej klasyfikacji czy regresji – te stosowane w regresji porządkowej są pewną ich modyfikacją. Możemy zatem używać:

- procentu poprawnej klasyfikacji, znanego też jako „dokładność” (ang. *accuracy*),
- średniego błędu bezwzględnego (ewentualnie kwadratowego),
- krzywej ROC i współczynnika VUS.

### 2.2. Procent poprawnej klasyfikacji (ACC)

Procent poprawnej klasyfikacji (ACC od ang. *accuracy*) mówi nam, ile obserwacji ze zbioru testowego zostało sklasyfikowanych poprawnie. Jeśli założymy, że  $y$  jest  $n$ -wymiarowym wektorem prawdziwych klas, zaś  $\hat{y}$  wektorem klas przez nas wyestymowanych, wówczas procentem poprawnej klasyfikacji nazywać będziemy:

$$ACC = \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{y_i = \hat{y}_i\}. \quad (2.1)$$

Współczynnik ten waha się od 0 (w przypadku, gdy żadna klasa nie została przyporządkowana poprawnie) do 1 (w przypadku właściwego sklasyfikowania wszystkich obserwacji).

Jego zaletami są niewątpliwie prostota i intuicyjność. Niestety, ma on także wiele wad. Przede wszystkim, działa źle w przypadku, gdy klasy są niezbilansowane. Przyjrzyjmy się temu problemowi na przykładzie.

Założmy, że rozpatrujemy przypadek trzyklasowy, w którym rozkład zmiennej odpowiedzi jest taki, że prawdopodobieństwo wystąpienia klasy pierwszej jest równe 0,9, drugiej 0,05,

a trzeciej też 0,05. Załóżmy również, że nasz klasyfikator jest bardzo prosty i – niezależnie od wektora cech – zawsze przyporządkowuje konkretnej obserwacji klasę pierwszą. Łatwo policzyć, że współczynnik ACC wyniesie wtedy aż 90%, mimo że klasyfikator jest bardzo słaby. Warto byłoby wziąć więc jakoś informację o niebilansowaniu klas pod uwagę.

Kolejną jego wadą jest to, że traktuje każdy błąd zerojedynekowo. Przykładowo, gdy rozważamy dziesięcioklasową regresję porządkową, ACC tak samo traktować będzie błąd spowodowany przypisaniu obserwacji o rzeczywistej klasie 10, jedynek jak i dziewiątki. Jako że klasy są z założenia uporządkowane, intuicja jest taka, że przyporządkowanie w powyższym przykładzie jedynki, powinno być niejako bardziej karane.

## 2.3. Średni błąd bezwzględny

Kolejnym wskaźnikiem jest średni błąd bezwzględny (MAE od ang. *Mean Absolute Error*). Korzystając z tych samych oznaczeń, co w przypadku ACC, możemy go zdefiniować jako:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|. \quad (2.2)$$

Jego niewątpliwą zaletą jest to, że bierze pod uwagę porządek klas (czyli – odwołując się do przykładu związanego ze współczynnikiem ACC – inaczej traktuje błąd przyporządkowujący dziesiątkę dziewiątkę, niż dziesiątkę jedynkę).

Wciąż nie jest jednak odporny na niebilansowane klasy. Co więcej, współczynnik ACC może być przedstawiany za pomocą procentu, a MAE niestety już nie – nie waha się bowiem w przedziale 0 – 1. Tracimy przez to jasność przekazu, gdyż ciężko stwierdzić, czy np.  $MAE = 1,19$  to dużo, czy mało.

## 2.4. Krzywa ROC w przypadku dwuklasowym

W przypadku dwuklasowym najczęściej stosowaną metodą oceny modelu jest porównywanie krzywych ROC (ang. *Receiver Operating Characteristic*) i pól pod tą krzywą, czyli AUC (ang. *Area Under the Curve*). Okazuje się, że można tę metodę uogólnić na nasz przypadek. Przyjrzyjmy się temu dokładniej.

Żeby łatwiej zrozumieć konstrukcję krzywej ROC w przypadku regresji porządkowej, przypomnijmy sobie najpierw, jak otrzymać ją w najprostszym – dwuklasowym – przypadku. 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).

Do stworzenia krzywej ROC potrzebne nam będą dwa wskaźniki – **czułość** (TPR, ang. *True Positive Rate*) i **specyficzność** (TNR, ang. *True Negative Rate*). Czuość 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.

		Prawdziwa klasa	
		+	–
Wystymowana 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 zostały 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

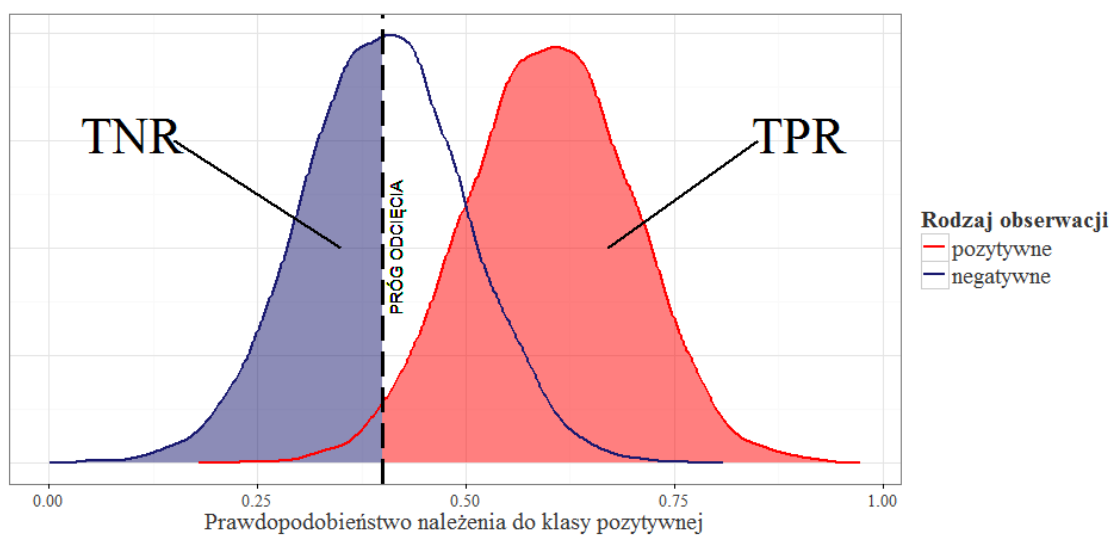
Korzystając z tabeli jakości dopasowania (patrz rys. 2.1), otrzymujemy:

$$TPR = \frac{TP}{TP + FN},$$

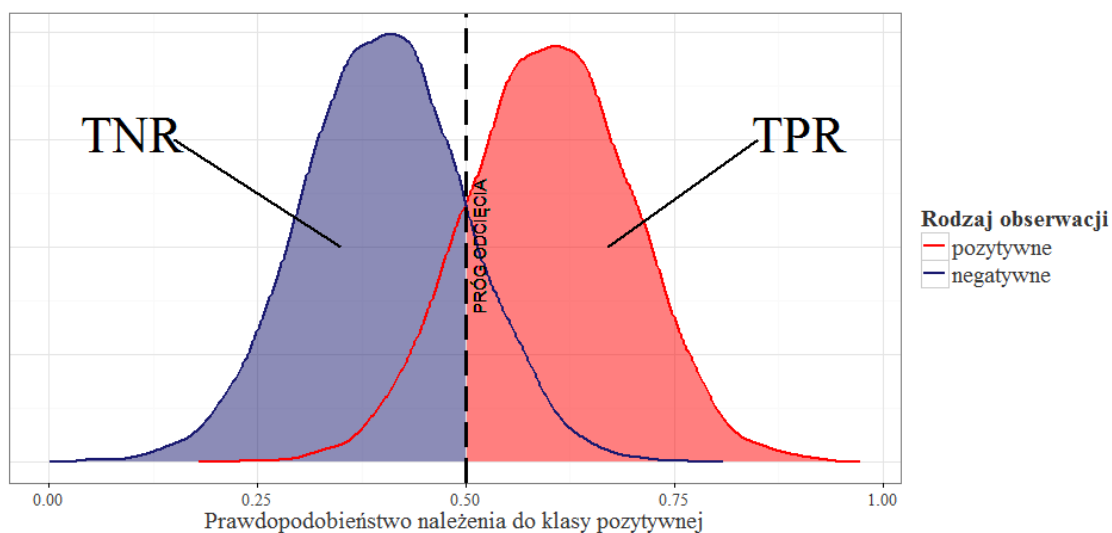
$$TNR = \frac{TN}{FP + TN}.$$

Krzywa ROC to wykres punktów (1–TNR, TPR), wyliczonych dla różnych progów odcięcia. Czym jest jednak próg odcięcia? Zanim odpowiemy sobie na to pytanie, zauważmy, że – w większości przypadków – model generuje nam nie tylko klasę, którą powinniśmy przypisać danej obserwacji, ale przede wszystkim prawdopodobieństwo tego, że ta klasa jest klasą prawdziwą. W przypadku dwuklasowym, przyjąć możemy, że dostajemy prawdopodobieństwo, że obserwacja należy do klasy pozytywnej. Standardowo przyjmuje się, że aby przypisać klasę pozytywną danej obserwacji, to owe prawdopodobieństwo powinno być większe niż 0,5. Niekoniecznie jednak musi tak być. Czasem wystarczy nam 40% pewności, żeby coś zaklasyfikować jako pozytywne. Dużo tu zależy od historii, która stoi za naszymi danymi. Na przykład, gdy zajmujemy się klasyfikowaniem pacjentów na chorych i zdrowych, wolimy częściej zakwalifikować kogoś jako chorego, gdy rzeczywiście jest zdrowy, niż odwrotnie, dlatego to prawdopodobieństwo często zmniejszamy. W przypadku analizowania na przykład kampanii reklamowych, może się zdarzyć, że podniesienie progu (czyli zakwalifikowanie mniejszej liczby klientów jako tych, do których wysłać reklamę) może znacznie zmniejszyć koszty naszej kampanii reklamowej i w rezultacie zwiększyć zysk. I właśnie tę liczbę, będącą rodzajem pewności wystarczającym, żeby zaklasyfikować obserwację jako pozytywną, nazywać będziemy progiem odcięcia.

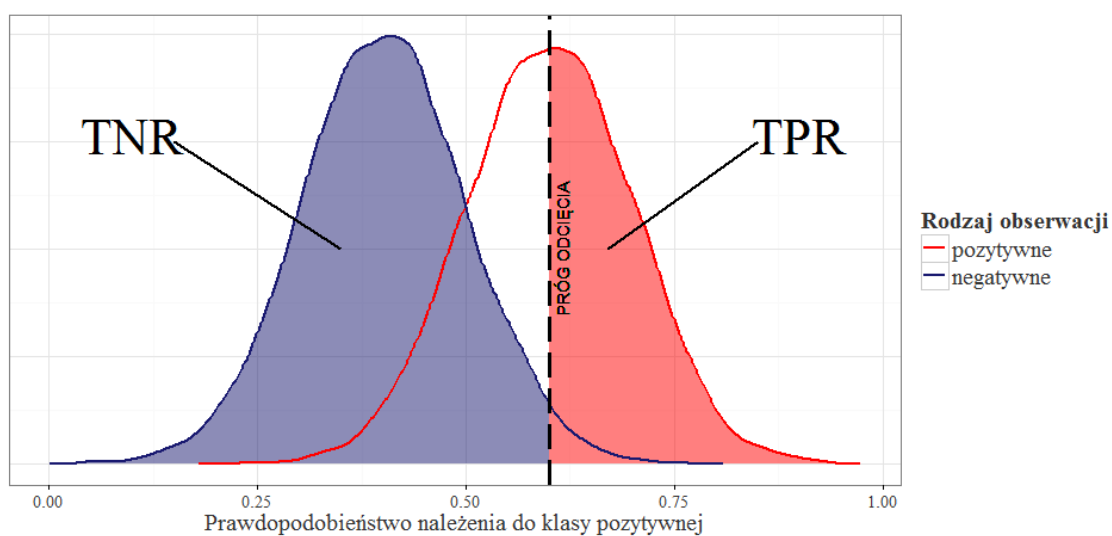
Przyjrzyjmy się rysunkowi 2.2. Mamy tu wykresy gęstości obserwacji pozytywnych i negatywnych, w zależności od przyjętego progu odcięcia. Większość obserwacji pozytywnych osiąga około 60–procentowe prawdopodobieństwo przynależności do klasy pozytywnej, a negatywnych 40–procentowe. Z wykresów wyraźnie widać, że poruszanie progiem odcięcia w prawo zwiększy nam czułość, ale zmniejszy specyficzność, natomiast poruszanie w lewo odwrotnie. Krzywa ROC łączy niejako te informacje, przedstawiając je na jednym wykresie. Patrząc na nią, możemy łatwo 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 jej dopełnienia



(a) Duże TPR, ale małe TNR

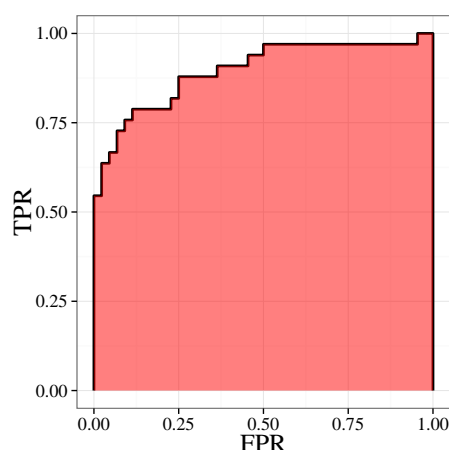


(b) Zrównoważone TPR i TNR

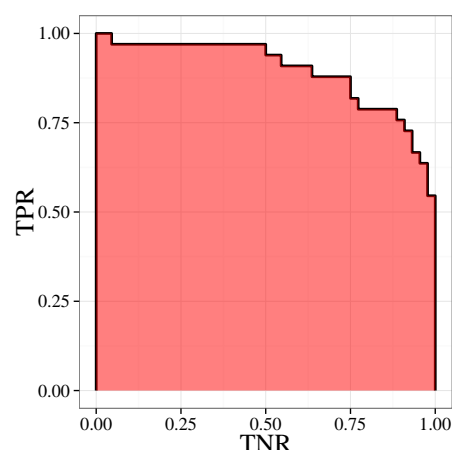


(c) Małe TPR, ale duże TNR

Rysunek 2.2: Sposób konstruowania krzywej ROC w przypadku dwuklasowym



(a) Standardowy sposób rysowania krzywej ROC



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

Rysunek 2.3: Przykładowe krzywe ROC

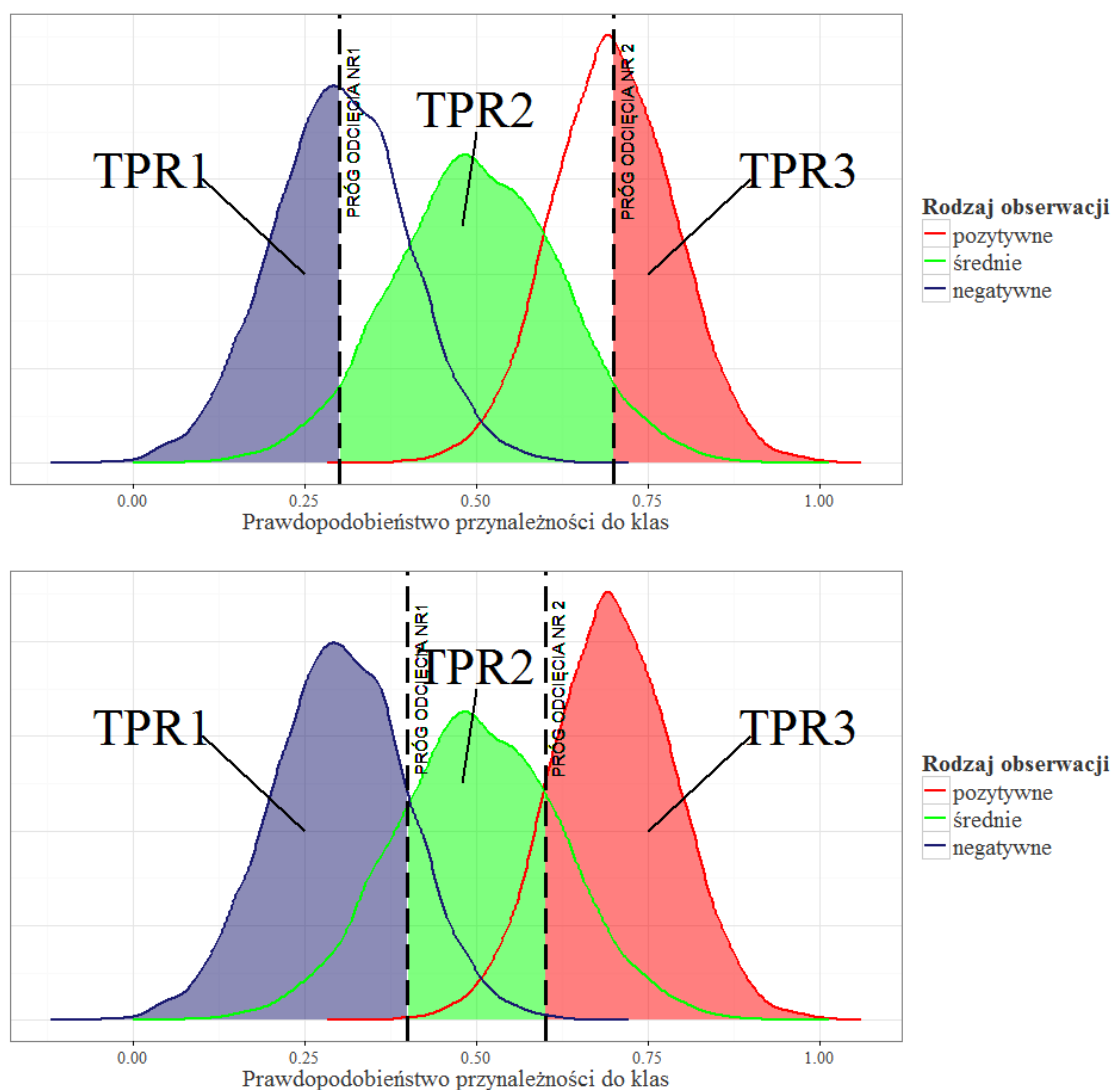
(tzn. 1–specyficzności), nazywanego FPR (ang. *False Positive Rate*). My jednak – by łatwiej było nam uogólnić krzywą ROC na więcej wymiarów – zastosujemy tę mniej popularną reprezentację, czyli na osi OX będziemy przedstawiać specyficzność. Jak widać na rysunku 2.3 nie zmienia to interpretacji krzywej.

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 (pole kwadratu jednostkowego), a model losowy charakteryzuje się AUC równym 0,5. Na rysunku 2.3 widać wyraźnie, że w naszym przypadku (czyli z inaczej zdefiniowaną osią OX) współczynnik AUC definiuje się identycznie.

## 2.5. Krzywa ROC w przypadku regresji porządkowej

W przypadku regresji porządkowej dochodzi problem wielowymiarowości. 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. Często zależy nam jednak na dobrym odróżnieniu właśnie tych skrajnych klas.

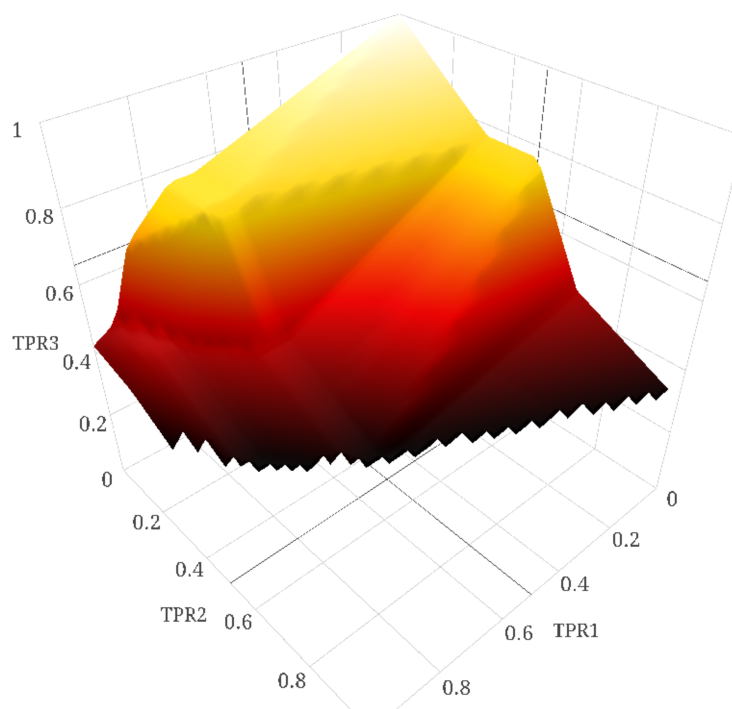


Rysunek 2.4: Sposób konstruowania krzywej ROC w przypadku trzyklasowym

Wyobraźmy sobie na przykład, 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 nieco bardziej 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 ( $TPR_1, \dots, TPR_r$ ), 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 wielowymiarowy

odpowiednik krzywej ROC, czyli pewną hiperpowierzchnię. Oczywiście jest to możliwe tylko w przypadku trzyklasowym (patrz rys. 2.5), a i tak rysunek taki jest mało czytelny.



Rysunek 2.5: Hiperpowierzchnia ROC w przypadku trzyklasowym

## 2.6. Współczynnik VUS

Po co zatem tworzyć wielowymiarową krzywą ROC, skoro i tak trudno cokolwiek z niej odczytać (w przypadku trzyklasowym) albo w ogóle narysowanie jej jest niemożliwe (w przypadku więcej niż trzyklasowym)? 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 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 rysunkowi 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 zwracany przez model procent pewności, że obserwacja należy do klasy pozytywnej, jest mniejszy niż pewien próg odcięcia. Analogicznie TPR to prawdopodobieństwo, że zwracany przez model procent pewności, że obserwacja należy do klasy pozytywnej, jest większy niż pewien próg odcięcia. Łącząc oba wyniki otrzymamy, że współczynnik AUC to nic innego tylko prawdopodobieństwo, że dwie losowo wybrane obserwacje (jedna pozytywna, a druga negatywna) będą poprawnie uporządkowane. Taką interpretację łatwo już uogólnić na więcej wymiarów. VUS będzie zatem prawdopodobieństwem, że losowo wybrane  $r$  obserwacji (każda z innej klasy) będzie dobrze posortowane.

Interesować nas będzie oczywiście pewna estymacja tego prawdopodobieństwa – będzie nią tzw. statystyka  $U$  Manna–Whitney’a–Wilcoxon (patrz [13], [16]), czyli wyrażenie:

$$VUS = \frac{1}{n_1 n_2 \cdot \dots \cdot n_r} \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \dots \sum_{i_r=1}^{n_r} \mathbb{I}_{\{f(\mathbf{x}_{i_1}^1) < \dots < f(\mathbf{x}_{i_r}^r)\}}, \quad (2.3)$$

gdzie  $\mathbf{x}_i^j$  oznacza  $i$ -ty wektor cech pewnej obserwacji o prawdziwej klasie  $j$ ,  $n_i$  to liczba obserwacji o rzeczywistej klasie  $i$ , a  $f(\cdot)$  to pewna funkcja, która zwraca liczbę rzeczywistą, mającą estymować uporządkowanie obserwacji (w większości przypadków wygląda to tak, że dla konkretnej obserwacji model zwraca  $r$  prawdopodobieństw przynależności do każdej z klas; wtedy jako  $f$  przyjmuje się maksimum z tych prawdopodobieństw).

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 pozwala wyliczyć (np. nie robią tego poznane przez nas sieci neuronowe). Trzeba wtedy odwołać się do prostszych metod (takich jak ACC, czy MAE). Większość modeli oferuje jednak taką możliwość, więc niewątpliwie warto z tego narzędzia diagnostycznego korzystać.



## Rozdział 3

# Porównanie metod modelowania i współczynników diagnostycznych

### 3.1. Opis eksperymentu

W poprzednich rozdziałach poznaliśmy różne metody radzenia sobie z problemem regresji porządkowej. Znamy już również wskaźniki, które pozwalają takie metody oceniać. Wszystko jednak zostało przedstawione w sposób bardzo teoretyczny. W tym rozdziale spróbujemy zobaczyć, jak poznane przez nas techniki modelowania i diagnostyki zachowują się na danych rzeczywistych. W tym celu przeprowadzimy następujący eksperyment.

Do dyspozycji mamy dziewięć zbiorów (*abalone*, *auto*, *diabetes*, *housing*, *machine*, *pyrim*, *stock*, *triazines* oraz *wdbc*). Są to standardowe i ogólnodostępne zbiory danych, wykorzystywane przy testowaniu regresji porządkowej, które znaleźć można na tych stornach internetowych: [3] i [15]. Każdy z nich charakteryzuje się inną liczbą obserwacji, inną strukturą macierzy obserwacji, inną liczbą klas zmiennej odpowiedzi oraz innym ich rozkładem. Większość z tych cech zaobserwować można na rysunku 3.1. Widać na przykład, że zbiór *abalone* ma klas 10, natomiast zbiór *housing* klas 5 oraz że rozkład zmiennej odpowiedzi zbioru *machine* jest bardziej skośny niż analogiczny rozkład w zbiorze *stock*.

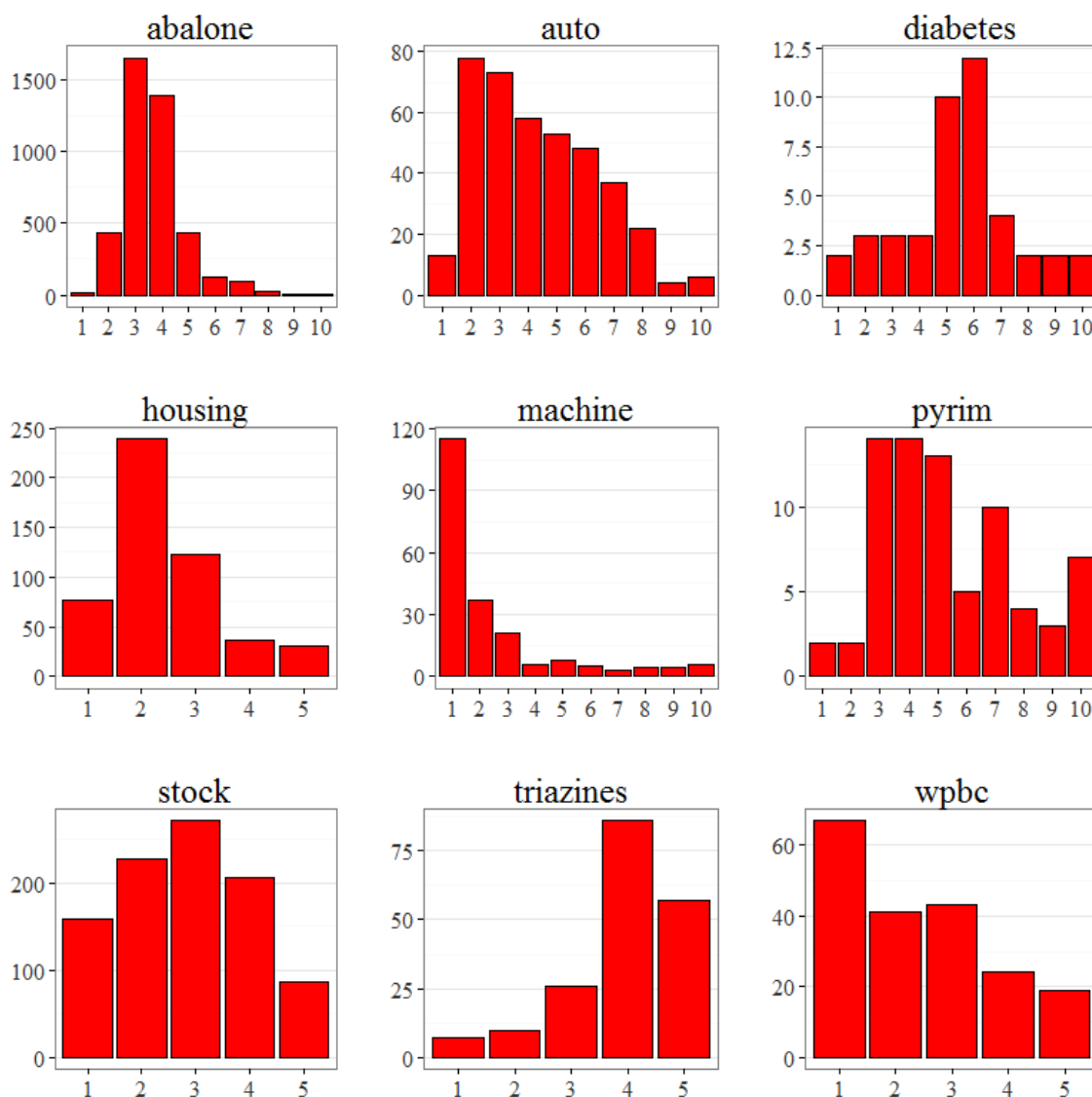
Każdy ze zbiorów został losowo podzielony na zbiór treningowy (do budowy modelu) i zbiór testowy (do oceny jakości modelu) w proporcji 7 : 3. Następnie, na każdym zbiorze treningowym zbudowany został każdy z pięciu modeli (model proporcjonalnych szans, model oparty o procesy gaussowskie, model Franka i Halla, sieci neuronowe oraz SVM) i przetestowany na zbiorze testowym. Dla każdego zbioru i każdej metody otrzymaliśmy współczynniki VUS, ACC i MAE (patrz równania (2.1), (2.2), (2.3)). Wszystkie wyniki zobaczyć można w tabeli 3.1. Dodatkowo, by móc porównać, czy metody regresji porządkowej rzeczywiście są potrzebne, zbudowany został również model klasyfikacji wieloklasowej. W tabeli 3.1 został on zaznaczony szarym kolorem, by móc go łatwo odróżnić i dzięki temu łatwiej odczytać tabelę.

### 3.2. Wstępna analiza wyników

Przyjrzyjmy się dokładniej tabeli 3.1. Pierwszą rzeczą, która rzuca się w oczy, jest bardzo słaba jakość współczynnika VUS. W większości przypadków jest on równy zero lub bardzo

		Procesy gaussow- skie	Model propor- cjonal- nych szans	Sieci neuronowe	Metoda Franka i Halla	SVM	Klasyfi- kacja wielokla- sowa
abalone	VUS [%]	0,00	0,00	–	0,00	0,00	0,00
	ACC [%]	56,38	55,18	56,86	55,18	56,30	57,58
	MAE	0,55	0,55	0,51	0,55	0,52	0,53
	BSC [%]	83,45	83,66	–	76,45	85,34	28,99
auto	VUS [%]	3,48	1,11	–	0,24	0,81	0,00
	ACC [%]	37,29	49,15	48,31	43,22	38,98	50,00
	MAE	1,08	0,63	0,66	0,73	0,90	0,64
	BSC [%]	94,16	94,12	–	88,77	87,96	33,80
diabetes	VUS [%]	0,00	0,00	–	0,00	0,00	0,00
	ACC [%]	23,08	23,08	23,08	15,38	23,08	30,77
	MAE	1,46	1,31	1,23	1,23	1,38	1,15
	BSC [%]	51,61	51,61	–	72,58	40,32	48,39
housing	VUS [%]	54,46	55,31	–	42,06	51,33	0,02
	ACC [%]	67,11	67,11	75,00	72,37	72,37	65,13
	MAE	0,36	0,35	0,26	0,28	0,29	0,40
	BSC [%]	91,86	92,25	–	86,92	91,98	51,15
machine	VUS [%]	0,00	0,00	–	0,00	0,00	0,00
	ACC [%]	57,14	68,25	66,67	66,67	60,32	69,84
	MAE	1,17	0,52	0,51	0,67	0,73	0,48
	BSC [%]	90,50	92,87	–	91,34	87,36	29,20
pyrim	VUS [%]	0,74	0,00	–	0,00	0,74	0,00
	ACC [%]	21,74	8,70	52,17	30,43	47,83	47,83
	MAE	1,30	1,65	0,91	0,91	0,96	0,87
	BSC [%]	81,82	74,09	–	80,00	82,73	61,82
stock	VUS [%]	53,97	73,12	–	56,53	96,01	0,01
	ACC [%]	58,95	72,63	83,16	81,40	91,58	86,67
	MAE	0,42	0,28	0,18	0,19	0,08	0,14
	BSC [%]	93,07	95,51	–	94,31	99,42	50,69
triazines	VUS [%]	1,72	0,00	–	0,61	1,83	0,00
	ACC [%]	51,79	39,29	39,29	53,57	51,79	44,64
	MAE	0,64	0,91	0,71	0,61	0,64	0,88
	BSC [%]	62,86	45,58	–	66,46	58,33	42,49
wpbc	VUS [%]	4,44	3,99	–	0,59	1,80	0,00
	ACC [%]	33,90	30,51	35,59	28,81	28,81	25,42
	MAE	1,12	1,25	0,86	1,02	0,92	1,29
	BSC [%]	67,59	61,29	–	62,67	57,53	40,48

Tabela 3.1: Tabela wyników. Kolumny odpowiadają kolejnym metodom modelowania, natomiast wiersze kolejnym zbiorom danych oraz wskaźnikom diagnostycznym. Na czerwono zaznaczony jest najlepszy wynik w każdym wierszu (nie uwzględnia on szarej kolumny, która odpowiada metodzie niezwiązanej z regresją porządkową)



Rysunek 3.1: Rozkłady odpowiedzi poszczególnych zbiorów danych

mały. Przypomnijmy, że reprezentuje on pewne prawdopodobieństwo, wobec czego już nawet wynik 50% nie byłby zadowalający. Sensowny wynik współczynnika VUS zobaczyć możemy tak naprawdę jedynie w przypadku zbiorów *housing* i *stock*. Warto więc zadać sobie pytanie, dlaczego tak się dzieje. Żeby zrozumieć to „dziwne” zachowanie współczynnika VUS, przeanalizujmy dwa poniższe przykłady.

Założmy, że chcemy przyporządkować klasy pewnym obserwacjom ze zbioru testowego. Niech zbiorem klas będzie  $\{1, \dots, 5\}$ . Dopasowaliśmy pewien model, który działa w ten sposób, że dla danej nowej obserwacji wylicza najpierw wartość pewnej funkcji rzeczywistej  $f(\cdot)$ , a następnie, korzystając z pewnych progów, przyporządkowuje jej jedną z pięciu klas, w zależności od tego, między jakimi dwoma progami znajdzie się wartość funkcji  $f(\cdot)$  dla tej obserwacji. Otrzymaną w ten sposób klasę oznaczmy przez  $\hat{y}$ . Funkcja  $f(\cdot)$  oraz wartości progów zostały nauczone podczas budowy modelu.

$y$	1	2	5	4	1	3	1
$\hat{y}$	1	2	1	4	1	3	1
$f$	1,3	2,2	1,1	4,4	1,2	3,5	1,4

Sortujemy po  $f$

$y$	5	1	1	1	2	3	4
$\hat{y}$	1	1	1	1	2	3	4
$f$	1,1	1,2	1,3	1,4	2,2	3,5	4,4

(a) Przypadek, w którym źle sklasyfikowana jest tylko jedna obserwacja. Zamiast klasy nr 5, została przyporządkowana klasa nr 1

$y$	1	2	5	4	1	3	1
$\hat{y}$	1	2	4	4	1	3	1
$f$	1,1	2,1	4,8	4,9	1,2	3,5	1,3

Sortujemy po  $f$

$y$	1	1	1	2	3	5	4
$\hat{y}$	1	1	1	2	3	4	4
$f$	1,1	1,2	1,3	2,1	3,5	4,8	4,9

(b) Przypadek, w którym źle sklasyfikowana jest tylko jedna obserwacja. Zamiast klasy nr 5, została przyporządkowana klasa nr 4

Rysunek 3.2: Tabele pokazują przykładowe zbiory testowe. W kolumnach przedstawione są kolejne obserwacje.  $y$  reprezentuje prawdziwą klasę, a  $\hat{y}$  klasę oszacowaną przez nas na podstawie wartości funkcji rzeczywistej  $f$ . Następnie, kolumny tabeli są sortowane według wartości funkcji  $f$

Przejdźmy do omówienia przykładów. Rozważmy najpierw przypadek z rysunku 3.2a. Wszystkie klasy – oprócz jednej – zostały prawidłowo przyporządkowane. Podkreślmy, że pomyliliśmy się tylko w jednym przypadku. Co prawda pomyłka jest poważna (gdyż przyporządkowaliśmy klasie nr 5 skrajnie różną wartość), ale wciąż jest to tylko jeden przypadek. Skorzystajmy ze wzoru (2.3), pamiętając jednocześnie, że VUS to procent dobrze uporządkowanych – w naszym przypadku – piątek (1, 2, 3, 4, 5) powstałych po posortowaniu wektora prawdziwych klas  $y$  według rosnących wartości  $f$ . Bardzo łatwo zauważyć, że współczynnik VUS wyniesie 0. Wynika to z faktu, że jedyna odpowiedź o numerze klasy 5 wśród zmiennej odpowiedzi

uplasowała się na samym początku, tym samym uniemożliwiając znalezienie choćby jednej poprawnie uszeregowanej piątki.

Ktoś mógłby powiedzieć, że pomylenie skrajnych wartości jest poważnym błędem, więc dobrze, że VUS wyszedł taki słaby. Przyjrzyjmy się zatem przykładowi z rysunku 3.2b. Tutaj sytuacja jest podobna – tym razem również została pomyłona tylko jedna wartość (obserwacja o rzeczywistym nr 5 dostała nr 4). Błąd ten jest jednak niewielki i – chciałoby się powiedzieć – dopuszczalny i nieszkodliwy. Jednak w tym przypadku również nie znajdziemy ani jednej dobrze posortowanej piątki, współczynnik VUS wyniesie więc 0.

Podsumowując współczynnik VUS, który ma bardzo ładną interpretację i znakomicie sprawdza się w przypadku klasyfikacji dwuetykietowej, w przypadku regresji porządkowej nie spełnia niestety swojej roli. Pozostaje nam zatem oparcie oceny klasyfikatorów jedynie na współczynnikach ACC i MAE. Nie jest to jednak satysfakcjonujące, gdyż VUS miał dawać nam pojęcie nie tyle o dokładnej poprawności klasyfikacji, ile o tym, czy obserwacje zostały poprawnie posortowane (patrz rozdział 2). Jest to aspekt bardzo istotny, gdyż tym właściwie różni się regresja porządkowa od innych rodzajów regresji czy klasyfikacji. Istnieje zatem silna potrzeba stworzenia współczynnika, który mógłby temu zadaniu sprostać. I właśnie w tym celu powstał współczynnik BSC (ang. Bubble Sorting Coefficient).

### 3.3. Współczynnik BSC

Zaproponowany przeze mnie współczynnik BSC ma niemal tę samą funkcjonalność, co wskaźnik VUS. Mianowicie, pokazuje nam, w jakim stopniu nasz model prawidłowo posortował dane testowe. Jednocześnie jest on odporny na takie odchylenia danych, które automatycznie powodują wyzerowanie współczynnika VUS, a które szczegółowo omawiane były w poprzednim podrozdziale. Zyskując tę odporność, tracimy jednak na jego teoretycznej interpretacji. Bowiem o ile VUS reprezentował pewne prawdopodobieństwo (patrz rozdział 2), o tyle BSC jest tylko jego pewnym przybliżeniem. Zanim wgłębimy się jednak w jego wady i zalety, zdefiniujmy najpierw, czym właściwie ten współczynnik jest.

---

#### Algorytm 3 Wyznaczanie współczynnika BSC

---

```
function liczba_przestawien(wektor)
  ile_zamian := 0
  for j in length(wektor) : 2 do
    for i in 1 : (j - 1) do
      if wektor[i] > wektor[i + 1] then
        zamień wektor[i] i wektor[i + 1]
        ile_zamian := ile_zamian + 1
      end if
    end for
  end for
end function
```

*w1* jest wektorem klas posortowanym ze względu na wartość funkcji  $f(\cdot)$   
*w2* jest wektorem klas posortowanym malejąco

```
return 1 - (liczba_przestawien(w1)/liczba_przestawien(w2))
```

---

Współczynnik BSC najprościej będzie zdefiniować przy pomocy algorytmu (patrz algorytm 3). Podobnie jak w przypadku VUS, sortujemy rzeczywiste klasy obserwacji według odpowiadających im i posortowanym rosnąco wartościom funkcji pomocniczej  $f(\cdot)$ . Następnie, stosując algorytm sortowania bąbelkowego, zliczamy ile przestawień sąsiednich wyrazów będzie potrzeba, by poprawnie uporządkować wejściowy wektor obserwacji. Porównujemy to do maksymalnej liczby przestawień (czyli do takiej, która jest potrzebna do rosnącego posortowania tego samego wektora, lecz uporządkowanego malejąco, a nie według wartości funkcji  $f(\cdot)$ ). W rezultacie otrzymujemy liczbę wahającą się od 0, reprezentującego najgorszy przypadek, do 1, reprezentującej przypadek najlepszy.

Przyjrzyjmy się teraz tabeli 3.2. Przedstawia ona 12 przykładowych wektorów klas oraz policzone dla nich współczynniki VUS i BSC. Łatwo zauważyć, że w większości przypadków są one równe. Różnią się za to znacznie tam, gdzie – wbrew intuicji – VUS jest równy zero np. w przykładach nr 5 i 8, które odpowiadają tym z omawianego przez nas wcześniej rysunku 3.2.

Nr	Posortowany według $f(\cdot)$ wektor prawdziwych klas								VUS [%]	BSC [%]
1	1	2	3	4	5	6	7		100,00	100,00
2	1	1	1	2	3	4	5		100,00	100,00
3	7	6	5	4	3	2	1		0,00	0,00
4	5	4	3	2	1	1	1		0,00	0,00
5	1	1	1	2	3	5	4		0,00	94,44
6	5	1	1	1	2	3	4		0,00	66,67
7	5	1	1	1	1	1	1		0,00	0,00
8	5	1	1	1	1	1	6		0,00	54,55
9	5	1	1	1	1	1	5		50,00	50,00
10	1	1	1	5	1	1	1		50,00	50,00
11	5	1	1	5	1	1	1		20,00	20,00
12	1	5	2	3	4	5	4		25,00	73,68

Tabela 3.2: Porównanie wskaźników VUS i BSC

Zalety współczynnika BSC są więc bardzo duże. Posiada on bowiem interpretację VUS, która jest niezwykle ważna w regresji porządkowej, a jednocześnie jest bardzo odporny na nietypowe sytuacje w danych. Szczególnie, że takie sytuacje (jak np. w przykładzie nr 5 z tabeli 3.2) pojawiają się bardzo często, gdy rozkład klas w danych jest nierównomierny.

Warto jednak podkreślić, że „dziedzicząc” po współczynniku VUS bardzo jasną intuicję, BSC przejmuje również niektóre jego wady. Przykładowo, gdy przyporządkujemy każdej z obserwacji klasę o jeden niższą, niż ma w rzeczywistości, wtedy zarówno VUS jak i BSC wyniosą 100%, gdyż wektor klas faktycznie będzie posortowany poprawnie. Jednak żadna klasa nie będzie przyporządkowana właściwie. Wadą jest również to, że i ten współczynnik stosować można jedynie do metod, które posługują się funkcją pomocniczą  $f(\cdot)$  przy wyznaczaniu klasy. Jest to niejako utrudnienie, gdyż na przykład sieci neuronowe takiej funkcji nie zwracają.

Trzeba jednak być świadomym, że w analizie wyników nie należy stosować tylko jednego współczynnika. Najbardziej skuteczne jest łączenie informacji. Warto spojrzeć zarówno

na ACC, VUS, MAE i wreszcie BSC i zastanowić się, na czym najbardziej nam zależy – na precyzji czy na kolejności dopasowania. Może czasem warto zrezygnować z doskonałej kolejności, by zyskać dużo większą precyzję? I odwrotnie. To już kwestia indywidualnego przypadku i charakteru naszych danych.

### 3.4. Dalsze wnioski

Wróćmy do wniosków płynących z eksperymentu i jego podsumowania zawartego w tabeli 3.1.

Na początku przyjrzyjmy się ostatniej kolumnie nazwanej *Klasyfikacja wieloklasowa* i zaznaczonej szarym kolorem. Jako jedyna z kolumn nie odpowiada ona żadnej z metod regresji porządkowej. Jest to prosty model wielomianowy dla danych nominalnych o więcej niż dwóch klasach. Krótko mówiąc, nie uwzględnia on uporządkowania naszej zmiennej odpowiedzi. Zauważmy, że współczynnik ACC jest dla metod regresji porządkowej i klasyfikacji wieloklasowej bardzo podobny – czasem trochę mniejszy, czasem większy, ale podobny. Nasuwa się zatem pytanie, po co komplikować modele i tworzyć różne metody przeznaczone specjalnie dla regresji porządkowej, skoro zwykła i znana klasyfikacja wieloklasowa daje taki sam efekt. Odpowiedź nasuwa się sama, gdy tylko przyjrzymy się współczynnikowi BSC. Otóż w większości przypadków jest on znacznie różny (a konkretnie znacznie mniejszy) od tego wyliczonego dla metod regresji porządkowej. Świadczy to o tym, że o ile stosując klasyfikację wieloklasową, popełniamy mniej więcej tyle samo błędów, to te błędy, które popełniamy są znacznie poważniejsze niż w pozostałych metodach (czyli przykładowo częściej mylimy skrajne wartości). Jednoznacznie potwierdza to skuteczność regresji porządkowej, która stworzona została właśnie po to, by być odporną na tego typu błędy.

Warto zwrócić również uwagę na niespodziewanie dobre zachowanie sieci neuronowych. Osiągają one najlepszy procent poprawnej klasyfikacji w 56% zbiorów, a w 22% niewiele odbiegają od najlepszego wyniku. Niestety nie możemy tutaj porównać współczynnika VUS ani BSC. Dlatego ryzykujemy, że uporządkowanie klas będzie słabej jakości. Trzeba jednak pamiętać, że metoda ta została specjalnie stworzona, by brać porządek klas pod uwagę, dlatego nie ma podstaw sądzić, że współczynniki te odbiegałyby bardzo od tych uzyskanych na tych samych danych dla innych metod.

Wreszcie, widać wyraźnie, że nie ma jednej najlepszej metody. Każda „wygrywa” na którymś ze zbiorów oraz każda ma swoje wady i zalety. Dlatego, przeprowadzając analizę na konkretnych danych, powinno się wziąć pod uwagę kilka istniejących sposobów modelowania, by następnie porównać je na zbiorze testowym i dopiero wtedy zdecydować, która z metod jest w naszym przypadku najlepsza. Zresztą takie podejście nie dotyczy tylko regresji porządkowej, ale wszystkich analiz statystycznych.





# Podsumowanie

Regresja porządkowa jest ważnym działem uczenia maszynowego. Od problemu klasycznej regresji różni ją to, że zmienna odpowiedzi jest nominalna, natomiast od problemu klasyfikacji wieloklasowej to, że zmienna odpowiedzi ma pewien naturalny porządek. W mojej pracy przedstawione i omówione zostały metody modelowania regresji porządkowej opisane w dostępnej literaturze. Dodatkowo, pokazane zostały wskaźniki diagnostyczne, mające na celu ocenę jakości klasyfikatorów. Wreszcie, wszystko to zostało zebrane, porównane i podsumowane na rzeczywistych danych.

Nową rzeczą, która została przedstawiona w pracy jest współczynnik diagnostyczny BSC (ang. *Bubble Sorting Coefficient*), który radzi sobie lepiej niż współczynnik VUS, zachowując jednocześnie podobną interpretację. Omówiony został dokładny algorytm jego obliczania oraz wady i zalety samego współczynnika.

Problem regresji porządkowej wydaje się mieć bardzo duże zastosowanie w praktyce (np. systemy rekomendacyjne czy badania socjologiczne). Jest jednak dostępnych bardzo mało opracowań teoretycznych tego tematu, nie wspominając o oprogramowaniu. Poza tym, wciąż otwartych jest wiele zagadnień, które warto rozwijać. Przede wszystkim trzeba tu wymienić regresję porządkową, w której klasom przyporządkowane są pewne wagi. Żeby zrozumieć, o co chodzi, odwołajmy się do przykładu ze wstępu mojej pracy, w którym chcieliśmy przewidywać opinię klienta o produkcie. Ważona regresja porządkowa zakładałaby tu, że bardziej zależy nam na poprawnym sklasyfikowaniu klientów, którym produkt potencjalnie może się spodobać niż na tych, którym może się nie spodobać (bo wtedy jest większa szansa osiągnięcia pewnych zysków) lub odwrotnie (bo wtedy nie stracimy pieniędzy na niepotrzebną reklamę). Ponadto, dalej powinno się rozwijać wskaźniki diagnostyczne, które – jak wykazała moja praca – nie są najlepsze.

Problem regresji porządkowej wciąż jest zatem problemem otwartym, który prawdopodobnie bardzo mocno się w najbliższych latach rozwinie.



# Literatura

- [1] Cheng J., Source Code for Nnrank Algorithm, [http://sysbio.rnet.missouri.edu/multicom\\_toolbox/nnrank%201.1.html](http://sysbio.rnet.missouri.edu/multicom_toolbox/nnrank%201.1.html)
- [2] Cheng J., Wang Z., Pollastri G., *A neural network approach to ordinal regression*, [w:] *IEEE: International Joint Conference on Neural Networks*, Hong Kong 2008, str. 1279–1284
- [3] Chu W., Benchmark of ordinal regression, <http://www.gatsby.ucl.ac.uk/~chuwei/ordinalregression.html>
- [4] Chu W., Ghahramani Z., *Gaussian Processes for Ordinal Regression*, [w:] „Journal of Machine Learning Research”, 2015, nr 6, str. 1019–1041
- [5] Chu W., Sathiyaraj Keerthi S., *Support Vector Ordinal Regression*, [w:] „Neural Computation”, 2007, nr 19, str. 792–815
- [6] Chu W., Source Code for Gaussian Processes for Ordinal Regression, <http://www.gatsby.ucl.ac.uk/~chuwei/README.gpor>
- [7] Chu W., Source Code for Support Vector Ordinal Regression, <http://www.gatsby.ucl.ac.uk/~chuwei/svor.htm>
- [8] Dobson A. J., *An Introduction to Generalized Linear Models*, wyd. 2, Londyn 2001
- [9] Ebdon M., *Gaussian Processes for Classification: A Quick Introduction*, [w:] *A Gentle Introduction to Gaussian Processes. Report in three parts*, 2008
- [10] Ebdon M., *Gaussian Processes for Regression: A Quick Introduction*, [w:] *A Gentle Introduction to Gaussian Processes. Report in three parts*, 2008
- [11] Frank E., Hall M., *A simple approach to ordinal classification*, [w:] *Machine Learning: ECML 2001. 12th European Conference on Machine Learning. Proceedings.*, Niemcy 2001, str. 145–156
- [12] Koronacki J., Ćwik J., *Statystyczne systemy uczące się*, wyd. 2, Warszawa 2008
- [13] Nakas C.T., Yiannoutsos C.T., *Ordered Multiple-class ROC Analysis with continuous measurements*, [w:] „Statistics in Medicine”, 2004, nr 23, str. 3437–3449
- [14] Rasmussen C., Williams C., *Gaussian Processes for Machine Learning*, 2006
- [15] Torgo L., Regression Data Sets, <http://www.dcc.fc.up.pt/~ltorgo/Regression/DataSets.html>
- [16] Waegman W., De Baets B., *A survey on ROC-based ordinal regression*, [w:] Fürnkranz J., Hüllermeier E., *Preference Learning*, 2010, str. 127–154



Marta Sommer  
Nr albumu 237503

Warszawa, 14 grudnia 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